2012 Second Brazilian Conference on Critical Embedded Systems

Multi-Agent Autonomous Patrolling System using ANN and FSM Control

Daniel Oliva Sales, Daniel Feitosa, Fernando Santos Osório, Denis Fernando Wolf

USP – University of São Paulo / ICMC – SSC – LRM (Mobile Robots Lab.) São Carlos, SP – Brazil

{dsales, feitosa, fosorio, denis} @ icmc.usp.br

Abstract

The multi-agent patrolling problem has recently received growing attention from the community due to the wide range of potential applications. This work presents an autonomous patrolling system composed by 4 intelligent robots that can freely move through an indoor environment and detect intruders. The robots use a localization/navigation system composed of an artificial neural network (ANN) trained to detect key features of the environment. These features are used to identify context changes, being used as input of a finite state machine (FSM), allowing a topological map localization and navigation of the robot in the environment. When an intruder is detected, a broadcast message with its position is sent, making all other robots execute a multi-agent version of a coordinated A algorithm in order to determine the best path to reach that position and to surround the target. Then, the robots autonomously navigate through this defined path until reach the goal. Experiments were performed in the player/stage environment in order to evaluate the multi-agent system. The localization/navigation system with intruder detection was evaluated in the real world with a Pioneer P3-AT mobile robot.*

1. Introduction

The application of different Artificial Intelligence techniques to Autonomous Mobile Robots has an important role in the scientific robotics community. Patrolling tasks have also been receiving growing attention because there is a wide variety of applications in real-world domains. This task is commonly performed by a group of agents [2].

This work presents an application of different AI techniques combined forming an embedded AI platform, designed for this multi-agent task: a patrolling system based on mobile robots capable of detect intruders and autonomously navigate through the environment, communicating between them when an agent is detected.

The autonomous navigation capability is one of the most desirable features in a mobile robot. In order to execute this task, three basic operations must be performed: localization, mapping and navigation [19].

An Intelligent Autonomous Robot capable of navigate into a structured environment composed by corridors, lobbies, and rooms must know its approximate localization, the environment map and the path to be followed (origin/destination). Navigation in this environment consists basically to follow a defined path, based on a previously defined map of the environment, and also considering its sensors data [19][16].

In order to perform a patrolling task, the robot must use its sensorial system not only to navigate, but also to detect intruders and anomalies. So, AI techniques embedded in the robot must have relatively low computational cost, working in soft real-time.

In this paper, a Topological Map Navigation/ Localization system was adopted. This approach was developed in a previous work [17] and its main advantage is to not require a well defined/detailed environment map; the environment is simply described as a graph with key-points and connections.

This navigation/localization system is based in a combination of an ANN [7] with FSM control [16]. The robots behavior and the path to be followed can be easily described as a sequence of states in a FSM: the path is a sequence of key-points to be visited, and the behavior is a response to the sensors when an agent is detected. The ANN is used to classify the data obtained from sensors, recognizing the key-points and detecting other agents. The detected agents can be intruders (enemies) or one of the team robots (allies).

Limited communication between robots is allowed. So, if an enemy agent (intruder) is detected, an alert message with its position is sent, and all robots must go to that position in order to surround it. A multi-agent version of a coordinated A-Star (A*) algorithm [6][13] is used in order to determine the best path between each robot and the enemy's position.

The main contribution of this paper is to show the feasibility of the previously developed topological navigation approach implementation in a multi-agent patrolling system, instead of using complex algorithms for pose estimation and mapping.

The next topics of this paper are organized as follows: section 2 contains a review of some related works; section 3 presents a system overview, with brief explanation of system components; section 4 shows the experimental results and section 5 the conclusion.

978-0-7695-4728-2/12 $26.00 © 2012 IEEE
DOI 10.1109/CBSEC.2012.14

2. Related Works

In Robotics, Finite State Machine (FSM) based approaches are often used as cited in [12], as for example, the "Situated Automata" and the "Reactive Deliberation Architecture" [15]. The use of a FSM provides a sense of sequence and context to the robot's behavior. This is especially useful when a simple reactive navigation system is not adequate; immediate reaction to the information provided by the sensorial system is not enough to guarantee the correct control when following a more complex path.

The use of FSM was adopted in multi-robots patrolling tasks, as for example, to create a behavioral control for multi-robot perimeter patrol [10]. In this work, the FSM was simple and pre-defined, and did not adopt any adaptive algorithm. The localization and navigation was based on pre-built maps created using a SICK laser range-finder and a SLAM method based on a particle filter. The system response actions to intruders were also limited.

The use of real-time machine learning method such Artificial Neural Networks (ANN) proved to be an interesting way to process the sensors data, identifying and classifying the FSM states (current and transitions), and determining the actions to be taken. In a recent work, [16] proposed an ANN integrated into a vision system, capable of recognizing different environment situations. These situations were used to generate a sequence of control actions, defining the FSM control. This work was based only in a vision system (no other sensors, as laser range-finders) capable of recognizing navigable paths on the ground. There is no intrusion detection (or response actions), or more sophisticated obstacle avoidance methods implemented on it.

ANNs are tolerant to noise and input data imprecision, and are also able to identify the states and transitions between the states [7]. ANNs are also very efficient in generalizing their knowledge and adjusting their outputs to many inputs, even when some inputs were not explicitly taught to the net (generalization capability). This way, ANNs are a very useful tool for path key-features classification/recognition and state detection.

Recent researches were also developed combining FSM and ANN in robotic problems (autonomous vehicles), since autonomous parking [8] until autonomous navigation in indoor and outdoor environments.

Several works presents strategies to perform the patrolling task. [2] presents an analysis of cyclic strategies versus partition-based strategies, concluding that cyclic strategies are better for most cases when there are no long "tunnels" separating regions.

Works as [18] and [9] are aimed in show the advantages of the application of Reinforcement Learning (RL) [20] in the patrolling task. However, as long as the RL use a Markov Decision Process (MDP) to do the patrol, the efficiency of this solution depends of the environment map.

In [11] a fault-tolerant modular approach is presented. This work shows that a behavior based strategy with predefined actions could construct a solid base to execute a complex task.

One of the most used algorithms for path planning in robotics is the A* algorithm. In [5] is presented an example of a successful robotic application of this heuristic to determine the best path between two points of the environment.

The proposed approach integrates the ANN learning with FSM features to recognize specific key-features of the environment (context) allowing the implementation of a robust FSM-based robot control architecture, similar to other approaches described in this section. The main topics of this work are: the integration of an ANN into the FSM control used in patrolling task, improving the work as presented by [10]; the Topological Map Localization and Navigation, based on the use of a laser ranger-finder, allowing a more robust navigation and improving the proposal presented by [16] (which uses a more simpler vision system); the possibility of defining different patrolling strategies as those proposed by [2] since different sequences of actions can be represented and coded into the FSMs; and the multi-agent coordinated response to this using an adapted A* algorithm to surround the target; the integration of these different AI techniques (ANN, FSM, Intrusion detection, A*, Multiagents coordination) composing a unique solution for autonomous mobile robots based patrolling tasks.

3. System Overview

The developed patrolling system is composed by a specific amount of robots, initially placed in different positions at the environment. Robot's control is fully decentralized; there is no communication with a central station. Moreover, the following assumptions must be held:

- each robot is autonomous
- each robot knows its initial position
- each robot can estimate its current position
- all robots know the environment topological map
- communication between robots is allowed

Robots behave according to current state detected combined with messages sent by the ally agents. Two main behaviors were implemented: patrolling and surrounding.

At the patrolling stage, the robots can patrol randomly or follow pre-determined cyclic paths until detect another agent. Self-localization occurs simultaneously to topological navigation. If an agent is detected by a robot, a broadcast message with detected agent's position is sent to robot team. If any ally reply with a position similar to detected, then patrolling behavior is kept; otherwise, surrounding behavior is activated, and robots must go to indicated position.

3.1 Topological Navigation/Localization System

As mentioned earlier, the robots have an embedded navigation/localization system based on the combination of ANN with FSM control. This kind of implementation for navigation tasks was specially chosen for this work due to its simplicity and low computational cost. Figure 1 shows the navigation system flowchart.

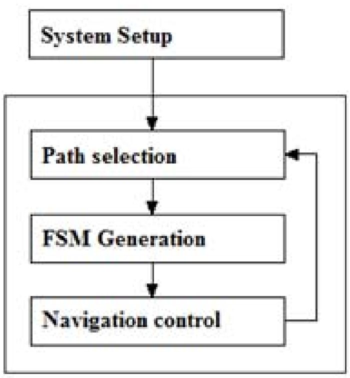

Figure 1. Navigation System Flowchart

System Setup

System setup is made offline and just once. The ANN implemented must be trained to accurately identify the keyfeatures of the environment such corridors, intersections and turns.

As the system is designed for a structured and well defined environment, supervised learning was used. A database with input/output pairs must be generated and classified by a specialist before training. The inputs must be data obtained from robot's sensors and outputs are the FSM states associated to each different key-feature.

FSM Generation

The environment can be represented by a topological map, a graph with key-points and adjacencies. This way, it is easy to establish a route between two points and all paths can be seen as a sequence of states (intermediate key-points that must be visited sequentially in order to reach the goal).

A corridor, for example, is defined as a continuous state in which the sensorial data is typically similar along its extension until the robot reaches a key-point (extremity, intersection, turning point) and the ANN recognizes it. So, the localization algorithm is based on a topological map (graph), and it isn't necessary to estimate very precisely the position of the mobile robot in the environment.

A module of the system is responsible to convert any complex path into a sequence of states. This description of sequence of states and associated actions to be executed by the robot (generated FSM) are stored in memory, allowing the robot to follow a specific complex pre-defined path.

Navigation task

A hybrid control was implemented. It combines the deliberative control obtained from the FSM approach with local reactive control behaviors. The deliberative control allows the robots to estimate its approximated position in the environment and to navigate topologically, in a "global navigation" point-of-view. Reactive control is used for "local navigation", inside each state (corridor, intersection, etc.). Each different state has its own actions, defined by an appropriated reactive control behavior.

The sensor processed data is used to know if the robot is still inside the current state or if a context change is needed. A state change occurs when the detected data is compatible with the next expected state.

Localization task

As the path is followed step by step, and each step is a specific part of the environment (related to a key-point in the topological map), it is easy to the robot to maintain the estimate of its own approximate position. The current state of the FSM gives this information, because the robot knows its initial position, the previous and the next key-points.

3.2 Path Planning

Path planning is performed in both behaviors. In patrolling behavior, each robot's route is previously specified (manually or by an algorithm that manages the robots' coverage area). In this work we do not discuss about optimal automatic patrolling path planning generation. Once defined the patrolling path (in our case it was defined manually), the robot FSM is automatically generated (considering the provided topological map of the environment), allowing the robot navigation control.

In surrounding behavior, automatic path planning is made with execution of A* algorithm. Each robot calculates the shortest path from its current approximated position to enemy's position taking into account the allies' positions (informed by them when the enemy was detected). The coordinated A* algorithm uses the topological map (graph) to generate the robots' path, adding an extra cost to graph edges which are already used by the other robots in planned paths, according to the algorithm described in [13].

After determining a path, the correspondent state machine is generated and stored to be followed in navigation step.

4. Experiments and Results

Two main experiments were performed: first, the navigation/localization system was evaluated individually (in simulated and real environments). Initially, the Player-Stage Simulator [4] was used to implement the proposed patrolling system, and it was also used to interface and control the real robot, a Pioneer P3-AT. Then, the full multi-agent patrolling system was evaluated in the simulated environment of Player-stage.

The main sensor used (for both simulated and real environments) was a 180º range Sick LMS200 Lidar (laser range-finder). This sensor was used to obtain the data provided to the ANN, in order to detect the key-features of the environment (e.g. corridors, intersections, turns).

Environment

The environment where experiments were carried out consists on an amount of connected corridors and 90º turns. This way, there are straights, turns and intersections to be processed by the system. Simulated environment is presented in Figure 2 and the real environment in Fig. 3.

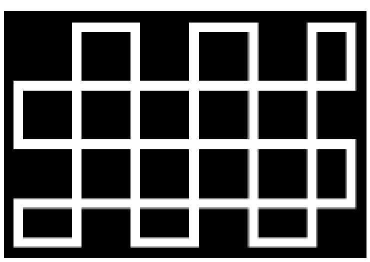

Figure 2 – Simulated environment map (player-stage)

Figure 3 – Real environment

FSM States

Five main states were defined: agent detection, straight, left turn, intersection and right turn. These five situations are illustrated in Figure 4.

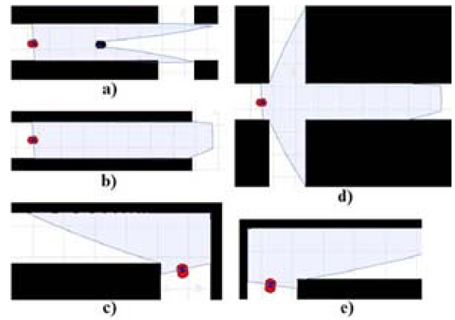

Figure 4 – FSM states. a) agent detection, b) straight, c) left turn, d) intersection, e) right turn

Implemented ANN

The ANN was implemented using Stuttgart Neural Network Simulator (SNNS), so the trained ANN was converted to C language, using SNNS2C tool, and integrated within the player-stage robot control programs.

The collected data to generate the database for ANN training was taken after manually control the robot through the map in different angles and positions. A log of laser detections was saved with related state manually associated (supervised learning).

The training algorithm used was Resilient Back-propagation (Rprop) [14]. This algorithm has been achieving good results for feed-forward networks for many applications comparing to other training algorithms in convergence and training time.

Several ANN architectures were tested. The number of neurons at the hidden layer and the number of hidden layers were adjusted until achieve good training results. The best network is a feed-forward Multi-Layer Perceptron (MLP) network described in Table 1.

Table 1 – ANN Learning

ANN Topology	180-20-5
Training Patterns	2269 Examples
Testing Patterns	566 Examples
Average Training Error	0 (MSE)
Average Testing Error	0 (MSE)
Learning Algorithm	RProp
Validation Method	5-fold cross-validation

All activation functions were defined as "Act_Logistic" function of SNNS that applies sigmoid logistic. The laser was set to cover 180º range with 1º of angle distance between each laser beam, so each input neuron corresponds to one laser beam only. The five neurons of ANN output are associated to each possible detected state.

Navigation System Evaluation

The navigation/localization system was tested in a Pioneer P3-AT robot equipped with a 180º range Sick Lidar as main sensor. The environment used in tests was a corridor with adjacent doors and intersections as shown on Figure 3.

Robot's task was to autonomously navigate through this corridor until reach a specific room and go inside it. This task was successfully performed by the robot, demonstrating the robustness and feasibility of the method.

A video showing some interesting parts of this experiment is available at: http://lix.in/-a99471

The multi-agent patrolling system was evaluated in player/stage environment. The simulation map used was shown in Figure 2. The evaluated system was designed with four robots (red, yellow, blue, green) in the team and one enemy robot (black) at the environment. Each robot was positioned at an opposite place and then autonomous navigation was started with a pre-specified path for each robot.

The intruder was intentionally positioned in the map in such a way that one of the patroller robots was able to find it. It was expected for this robot to send a broadcast message indicating the enemy's position. The other robots should have changed to surrounding behavior and redesigned their paths, going into enemy's direction. This task was successfully completed in all experiments.

In order to better observe the robots behavior using the ANN-FSM algorithm and the A* path planning at an intruder detection, a video with the demonstration of one full execution is available at: http://lix.in/-a80dc6.

Figure 7 shows a final screenshot of one execution, with enemy agent (black robot) surrounded by the four robots that form the intrusion detection team.

The A* algorithm was implemented using the available environment map. Robots planned the best path to enemy's position taking into account the allies' positions. Then, the navigation from the current position to the destination position of each robot was done using the ANN-FSM method.

Communication and Coordination

Communication between robots in Agent Detection State was an important feature to behavior change and path planning. Knowing the allies' positions (through this implemented communication) is fundamental to execute the coordinated A* algorithm. All messages must be sent in broadcast mode, and it must be considered that all robots are always in communication range. Communication range must be equal or bigger than the full patrolled area.

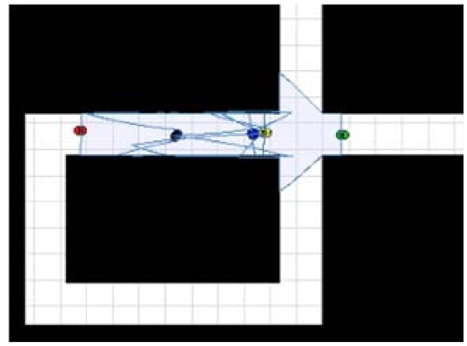

Figure 7 – Successful execution of algorithm, with enemy (black robot) surrounded by the robot team

Localization System Evaluation

Experiments were also made with robots performing random paths in patrolling behavior. A random decision should be taken in Intersection State, making the robot randomly choose between go straight or turn to any side.

Localization task was still accurate in this condition because the current and past positions were always known, due to FSM approach advantage: it is capable to deal with temporal (sequences of states) and contextual information.

Results

The implemented method obtained very good results, with 100% accuracy on classifying the sensorial data and moving through the environment in all experiments carried out.

5. Conclusion and Future Works

The association of ANN with FSM control in a Topological approach showed to be a very useful approach for mobile robots navigation. This navigation method showed to be very interesting to patrolling tasks. The AI techniques and algorithms adopted also showed to be appropriate to real-time embedded systems.

The system can be trained to recognize many other different situations and to use and combine other different types of sensors. This way, there is a potential and promising possibility of applying this system in different types of tasks, for both indoor and outdoor applications.

6. Acknowledgements

The authors acknowledge FAPESP and CNPq for their support to INCT-SEC (National Institute of Science and Technology - Critical Embedded Systems - Brazil), processes 573963/2008-9 and 08/57870-9 and financial support to authors (master´s grant).

7. References

[1] Agmon, N.; Kraus, S.; Kaminka, G.A. Multi-robot perimeter patrol in adversarial settings. In Proceedings of 2008 IEEE International Conference on Robotics and Automation, pages 2339–2345, may 2008.

[2] Chevaleyre, Y. Theoretical analysis of the multi-agent patrolling problem. In Proceedings of 2004 IEEE/WIC/ACM International Conference on Intelligent Agent Technology, pages 302–308, sep. 2004.

[3] Elmaliach, Y; Agmon, N; Kaminka, G.A. Multi-robot area patrol under frequency constraints. In Proceedings of 2007 IEEE International Conference on Robotics and Automation, pages 385–390, april 2007.

[4] Gerkey, B.; Vaughan, R.T; Howard, A. The Player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems. In: Proceedings of the 11th International Conference on Advanced Robotics (ICAR 2003), pages 317-323, Coimbra, Portugal, June 2003.

[5] Gürel, U.; Parlaktuna, O; Kayır, H.E. Agent-Based Route Planning for a Mobile Robot. 5th International Advanced Technologies Symposium (IATS'09), Karabuk, Turkey, May 13-15, 2009.

[6] Hart, P.E; Nilsson, N.J; Raphael, B. A formal basis for the heuristic determination of minimum cost paths. IEEE Transactions on Systems Science and Cybernetics SSC4 (2), pages 100–107, 1968.

[7] Haykin, S. Neural Networks: A Comprehensive Foundation. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1998.

[8] Heinen, M. R.; Osório, F. S.; Heinen, F. J.; Kelber, C. SEVA3D: Using Artificial Neural Networks to Autonomous Vehicle Parking Control. In IEEE WCCI-IJCNN. 2006

[9] Hu, Z.; Zhao, D. Reinforcement learning for multi-agent patrol policy. In Proceedings of the 9th IEEE International Conference on Cognitive Informatics, pages 530–535, july 2010.

[10] Marino, A; Parker, L; Antonelli, G; Caccavale, F. Behavioral Control for Multi-Robot Perimeter Patrol: A Finite State Automata approach. In: Proceedings of IEEE International Conference on Robotics and Automation (ICRA 09'), 2009.

[11] Marino, A; Parker, L; Antonelli, G; Caccavale, F; Chiaverini, S. A fault-tolerant modular control approach to multi-robot perimeter patrol. In Proceedings of 2009 IEEE International Conference on Robotics and Biomimetics, pages 735–740, dec. 2009.

[12] Medeiros, A. A survey of control architectures for autonomous mobile robots. Journal of the Brazilian Computer Society - JBCS. vol.4, n.3. 1998.

[13] Osório, F; Pessin, G; Ferreira, S; Nonnenmacher, V. Artificial Intelligence for Games: Special Agents with License to Kill... and Reasoning!. In Proceedings of the VI SBGames - Brazilian Symposium on Computer Games and Digital Entertainment, São Leopoldo, Brazil. 2007.

[14] Riedmiller, M; Braun, H. A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP Algorithm. In: IEEE International Conference on Neural Networks, pages 586—591, 1993.

[15] Sahota, M.K. Reactive Deliberation: An Architecture for Real-Time Intelligent Control in Dynamic Environments. In Proceedings of the AAAI-94, p. 1303–1308. 1994

[16] Sales, D; Shinzato, P; Pessin, G; Wolf, D; Osório, F. Vision-based Autonomous Navigation System using ANN and FSM Control. In: Proceedings of the IEEE Latin American Robotics Symposium (LARS), São Bernardo do Campo, Brazil, 2010.

[17] Sales, D; Osório, F; Wolf, D. Topological Autonomous Navigation for Mobile Robots in Indoor Environments using ANN and FSM. In: Proceedings of the I Brazilian Conference on Critical Embedded Systems (CBSEC), São Carlos, Brazil, 2011.

[18] Santana, H; Ramalho, G; Corruble, V; Ratitch, B. Multi-agent patrolling with reinforcement learning. In Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems, pages 1122–1129, 2004.

[19] Siegwart, R; Nourbakhsh, I. Introduction to Autonomous Mobile Robots. MIT Press. 2004.

[20] Sutton, R.S; Barto, A.G. Reinforcement Learning: An Introduction. MIT Press, Cambridge, MA, 1998.