Universitat de Lleida

Document downloaded from:

http://hdl.handle.net/10459.1/59345

The final publication is available at:

https://doi.org/10.1007/978-3-319-40159-1_14

Copyright

(c) Springer International Publishing Switzerland, 2016

A proposal of a multi-agent system implementation for the control of an Assistant Personal Robot

Dani Martínez, Eduard Clotet, Javier Moreno, Marcel Tresanchez and Jordi Palacín

Department of Computer Science and Industrial Engineering,
University of Lleida, 25001 Lleida, Spain

Abstract. This paper proposes a control system design for a mobile robot assistant based on a multi-agent architecture. The robotic platform used in this paper is a second generation Assistant Personal Robot (APR-02) with an own design. The control implementation is distributed among different agents in which each one is designed to fulfill a specific functionality such as localization, navigation, task managing, vision, hearing, communications, and environmental supervision. In addition, a set of shared memory instances are implemented to ensure the cohesion among all the agents while working together. The proposed methodology provides robustness and effectiveness by assigning each agent on a single CPU thread.

Keywords: Robot agent, Personal Robot, multi-agent.

1 Introduction

Since last decades robotics has advanced from performing limited and particular tasks to become more flexible being capable of realizing different applications with high levels of adaptability. Agents have the leadership of this evolution by means of providing autonomy, reactivity, collaboration, and initiative to robotics [1]. Moreover, from such new concepts new challenges have emerged which are very popular among researchers in robotics such as navigation, localization, and mapping. The simultaneous location and mapping method (SLAM) is a popular probabilistic technique that estimates relative robot displacements by environmental observation at the same time the robot is building a virtual map[2]. So, this method allows the robot the possibility of using the estimated poses and the virtual map to perform complex navigation procedures. In [3] localization and navigation algorithms are presented for autonomous robots that can patrol multiple floors of the building. Furthermore, spatial representation is also important in autonomous navigation to recognize the area in which the robot is [4]. A multi-agent framework for autonomous systems was proposed in [5] that provides a fluid operation of multiple control loops within an agent.

Assistant robots have also take advantage of the agent concept and have been enhanced to become so much more operational. This field in robotics is one of the most interesting due to its deep challenge and its final application that can help common people in their homes. The robot methodologies presented in [6] are applied on some of the first developed assistive robots for home care and museum guidance. Elder people and persons with reduced mobility are a special motivation on developing assistant robots. In [7] is presented a home assistive mobile robot capable of tracking the user, engage a remote supervision, and other robot services. Moreover, there are other applications which use similar platforms such as for ambient intelligence services as the robot described in [8].

This paper proposes a practical methodology to implement a control system for an assistive robot based on a multi-agent design. The main platform used in this paper is an Assistant Personal Robot (APR) as the one described in [9]. Several agents are implemented in order to engage different functionalities to the mobile robot. The motivation of implementing a multi-agent architecture in a mobile robot is to provide independence, versatility, and reliability among different robot sub-systems. The agents are described in this paper and are based on a modular architecture with a complete collaborative behavior. The proposed tests are based on measuring the overall performance of the whole system on the robot computational unit. Each agent is assigned to a single processing thread and their computational requirements can be quantified.

2 Agent methodology

The methodologies presented in this paper are implemented on an APR platform which has several onboard sensors and actuators: laser range sensor (LIDAR), a 3D camera, a motor control board, a main display, and the robot main computational unit. The APR computational unit implements several robot agents, each one assigned to perform a single dynamic function. Furthermore, such robot agent processes require a constant communication among them to engage a correct development of its tasks by means of using shared memory spaces. Fig. 1 shows a diagram of the proposed multi-agent system implemented in the APR. Most of the described robot agent processes have a direct connection to a hardware device. So, the communication between the robot and an external device is exclusively restricted to a single robot agent.

The agents are designed to work independently, they only have to publish their results and gather information needed while performing their tasks. Such exchange of information among agents is managed by using shared memory instances subject to continuous querying and publishing operations. Although agents are independent, the global effectiveness of the robot system depends on the cooperation among all of them. The agents implemented in the APR platform are: task manager agent, SLAM agent, navigation agent, face recognition agent, voice recognition agent, sensor acquirement agent, and communications agent.

2.1 Assistant Personal Robot

The Assistant Personal Robot is a robotic system focused on implementing assistive services for reduced mobility people. Furthermore, this robotic design (fig. 2) allows its use into many other different applications such as a guide robot, a measurement platform, for surveillance services, and more. The design of the robot shape is focused to imitate a person maximizing the human comfort level provided by the robot presence and its social interactions. The motion of the robot is performed by three omnidirectional wheels providing high mobility in any situation. Such wheels are connected to three DC motors which are guided by an ARM-based motor control board. Moreover, this electronic board also control other motors such as the ones placed in the arms and in the head which handles the main screen. In this paper the APR-02 is used whose main computational unit is a domestic high-performance computer with reduced dimensions (ITX motherboard) running Windows. The whole system is powered by three 12V 12Ah DC batteries placed at the robot base. Additionally, the platform has a several sensors attached in order to obtain profitable environmental information. A Hokuyo UTM-30LX laser range sensor placed on the base provides the information used by the SLAM procedure as well as for the obstacle avoiding procedure. Furthermore, a Creative Senz 3D Camera is placed at the top of the platform which obtains RGB-D images for the object/face recognition process. This camera also has a dual built-in microphone which is used to perform a voice recognition process to decode simple vocal orders.

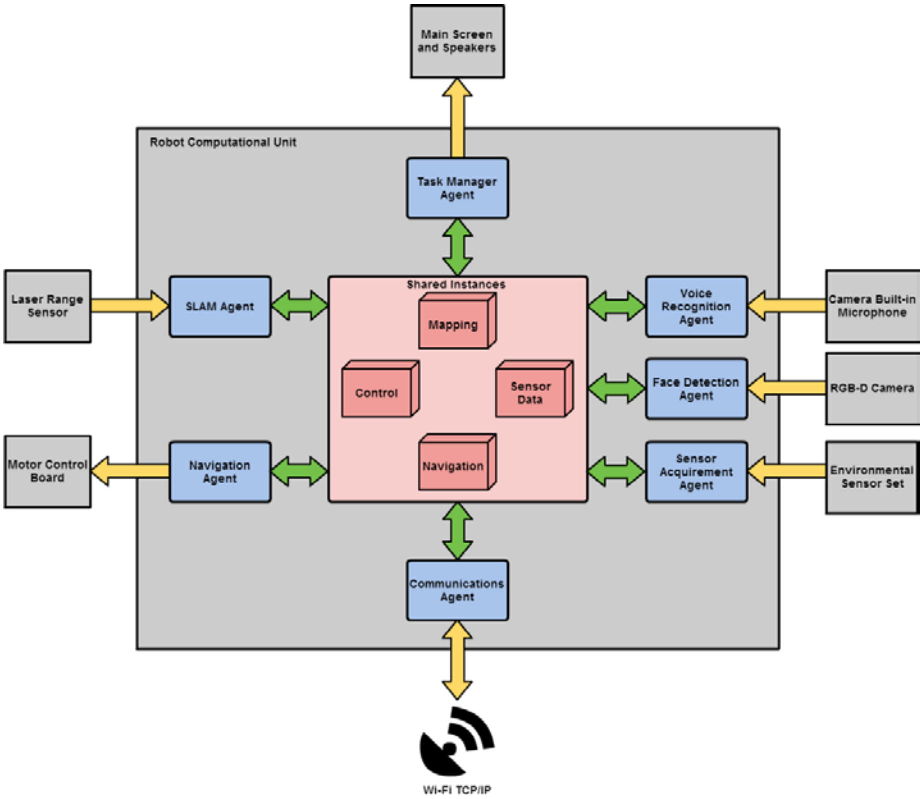

Fig. 1. Diagram of the proposed APR multi-agent architecture.

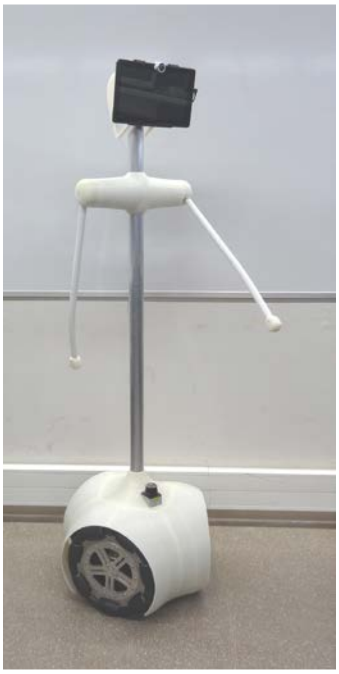

Fig. 2. Prototype of the Assistant Personal Robot: APR-01.

2.2 Task manager agent

The main process of the proposed system is the task manager agent. This agent manages the lifecycle and the main decisions of the rest of the agents by means of a task queue system. The robot routines are predefined by atomic tasks which can be also used by other routines such as “going to somewhere”, “find persons”, “look at face”, “show advice”, “request assistance”, “answer yes”, “say hello”, “perform something with arms”, “follow someone”, “check ambient conditions” and so on. Such actions are assigned to the task queue in order to execute them in an established order; however, they can be interrupted depending on the task priority. In order to ensure the fulfillment of the tasks, the agent uses the shared instances to check and control the different requirements of the task. Moreover, this agent also manages the output communication features of the robot by means of displaying images on the main screen and playing simple sounds through the embedded speakers.

2.3 SLAM agent

The implementation of the Simultaneous Localization And Mapping (SLAM) procedure is based on processing occupancy grid data [2] by applying a template matching method. The laser range sensor is continuously obtaining scans which are processed by this procedure returning new updated location of the robot coordinates and orientation. In order to reduce the computational cost of this process, the mapping part of this procedure is only applied in unknown or previously unexplored areas. In this way, the robot saves computational resources for other robot agent processes executed in the control system. The SLAM agent has an embedded obstacle avoiding procedure that detects unexpected static and dynamic obstacles. Once an obstacle has been detected, the SLAM agent requests the navigation agent an attempt to recalculate the robot path.

2.4 Navigation agent

The autonomous path planning process is defined in the navigation agent that is directly connected with the motor control board. This agent computes the path to reach the actual robot destination and emits motion orders. The path planning method implemented in the agent is based on a graph shortest path algorithm [6] applied with a discretized version of the virtual area map. So, this agent works simultaneously with the SLAM agent which is permanently obtaining new relative poses of the robot inside the virtual area map. The information used by such agents is available at the navigation and mapping shared instances. Figure 3 shows an example virtual map built by the APR with its actual destination marked. Additionally, the robot agent has some simple exploratory routines to create a new virtual map for first-use in unknown environments based on wall-following and remote-controlling.

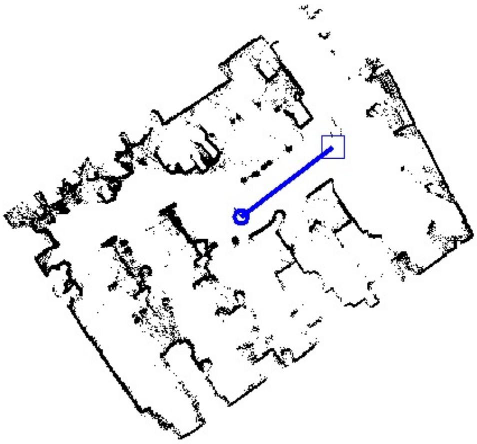

Fig. 3. An example of a virtual map used by the navigation agent of the APR.

2.5 Face and voice recognition agents

The face and voice recognition agents are planned to simplify the interaction with the APR. The face recognition agent obtains images from the RGB-Depth camera and will detect and identify the existing faces on the images by using a pre-loaded face template database [10]. Figure 4 shows an example of a segmented image taken from the RGB-D camera in which the depth has been limited to 75cm. Once the person is identified, the personal profile is loaded for the assistive routines. The voice recognition agent detects single word instructions by computing the voice signal in time-domain and identifying the vocals [11]. At this point, the vocalized instructions are notified to the task manager agent in order to execute the formulated order of instructions.

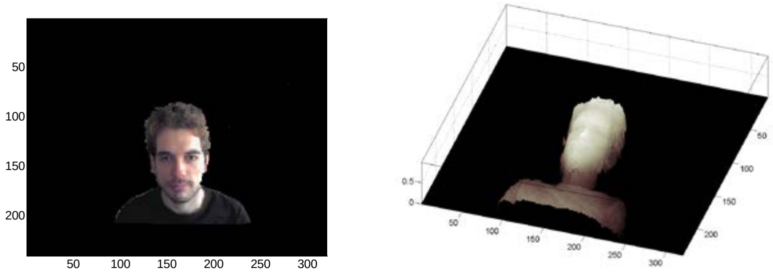

Fig. 4. Example RGB (left) and Depth (right) images.

2.6 Sensor acquirement agent

The environmental conditions can be monitored by the APR by means of a set of sensors attached to the platform. The sensor acquirement agent keeps executing constantly and gets the data from the connected sensors such as a thermometer, an odor sensor, an anemometer, and more. Such data is published at the sensor data shared instance for further use from other robot agents. Additionally, the agent can create a log register from the gathered sensory data. Figure 5 shows the measurements from different sensors while the robot was patrolling indoors.

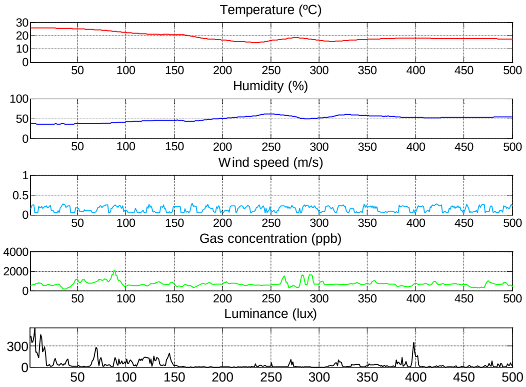

Fig. 5. An example of different acquired sensor data from the APR.

2.7 Communications agent

An agent dedicated to external communications is included in order to engage a wireless connection with a remote terminal for external supervision. This system allows a user to visualize in real-time the robot location, the area map, the sensor lectures, the actual task, and images from the RGB-D camera.

3 Results

This section presents the results on performance and feasibility of the methodology proposed in this paper. The APR platform used for the tests is the APR-02 which central processing unit is an Intel Core i7-6700 3.40GHz with 16GB of DDR4 memory and a solid state drive hard disc. This microprocessor has 8 execution threads from 4 physical cores in which its computational load is managed and measured by the operating system. In this paper the agent processes are assigned as a single execution tread in order to isolate them and avoid disturbances on other agents which may require high computational requirements. In this paper, each agent is executed in an individual Java virtual machine. All agents are working even when the robot is stopped. Table 1 shows the agent assignment by execution threats and the average computational load measured by the operating system.

Table 1. Agent assignment table.

Agent	**Average CPU thread load**							
	#0	**#1**	**#2**	**#3**	**#4**	**#5**	**#6**	**#7**
Task manager	8%	0%	0%	0%	0%	0%	0%	0%
SLAM	0%	86%	0%	0%	0%	0%	0%	0%
Navigation	0%	0%	13%	0%	0%	0%	0%	0%
Voice recognition	0%	0%	0%	7%	0%	0%	0%	0%
Face recognition	0%	0%	0%	0%	14%	0%	0%	0%
Sensor acquirement	0%	0%	0%	0%	0%	4%	0%	0%
Communications	0%	0%	0%	0%	0%	0%	7%	0%

Table 1 shows that each agent runs on a unique CPU thread and that the most demanding task at this moment is the SLAM procedure because the implementation of the face recognition system does not processes all images acquired by the APR. Further tests performing simple routines show that the mobile robot offers robustness and coordination among all the implemented agents. The robot keeps executing all the agents at full operability without any delay or complications caused by other agents. The state of the robot and some important information is visualized from a remote application installed to a personal computer.

4 Conclusions

This paper presents a multi-agent architecture designed to control an Assistant Personal Robot. Different robot agents are defined as independent processes for an effective operability. In this paper the agents are executed on Java virtual machines in unique CPU threads but can also be implemented on any other platform that provides modular programing such as ROS nodes. The first tests with the APR have demonstrated that the proposed multi-agent architecture operating in individual threads has improved the robustness and the applicability of the control system. Furthermore, the modular implementation of the robot features simplifies the development and debugging of tasks while avoiding irregular behaviors or inconsistent processes due to instantaneous CPU usage by an individual agent. Future work will be focused on the full development of all assistant robot routines. Field experimentation will be also performed in order to obtain reliable results from different APR services.

Acknowledgements

This work is partially founded by Indra, the University of Lleida, and the RecerCaixa 2013 grant.

References

1. Wooldridge, M., Jennings, N.: Intelligent Agents: Theory and Practice. The Knowledge Engineering Review, vol. 10, no. 2, pp. 115-152. Cambridge University Press, Cambridge (1995).
2. Thrun, S.: Robotic mapping: A survey. Exploring artificial intelligence in the new millennium, pp. 1-35. (2002).
3. Biswas, J., Veloso, M. M.: Localization and navigation of the CoBots over long-term deployments. The International Journal of Robotics Research, vol. 32, no. 14, pp. 1679-1694. (2013).
4. Zender, H., Mozos, O. M., Jensfelt, P., Kruijff, G. J., Burgard, W.: Conceptual spatial representations for indoor mobile robots. Robotics and Autonomous Systems. vol. 56, no. 6, pp. 493-502. (2008).
5. Py, F., Rajan, K., McGann, C.: A systematic agent framework for situated autonomous systems. 9th International Conference on Autonomous Agents and Multiagent Systems, vol. 2, pp. 583-590. (2010).
6. Graf, B., Hans, M., Schraft, R.D.: Mobile robot assistants. Robotics & Automation Magazine, vol. 11, no. 2, pp. 67-77. (2004).
7. Gross, H. M., Schroeter, C., Mueller, S., Volkhardt, M., Einhorn, E., Bley, A., Merten, M.: Progress in developing a socially assistive mobile home robot companion for the elderly with mild cognitive impairment. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2430-2437. (2011).
8. Martinez, D., Teixidó, M., Font, D., Moreno, J., Tresanchez, M., Marco, S., Palacín, J.: Ambient intelligence application based on environmental measurements performed with an assistant mobile robot. Sensors, vol. 14, no. 4, pp. 6045-6055. (2014).
9. Clotet, E., Martínez, D., Moreno, J., Tresanchez, M., Palacín J.: Development of a high mobility Assistant Personal Robot for home operation, Advances in Intelligent Systems and Computing, vol. 376, pp. 65-73. (2015).
10. Viola, P., Jones, M. J.: Robust real-time face detection. International journal of computer vision, vol. 57, no. 2, pp. 137-154. (2004)
11. Vilarrubi, M., Tresanchez, M., Martinez, D., Moreno, J., Garriga, J. A., Claria, F., Palacin, J.: Time domain detection of pure vowel sounds for simplified computer interaction. In: 17th International Conference on Information Fusion (FUSION), pp. 1-6. (2014).