APPLYING MULTI-AGENT SYSTEMS AND BIO-MIMETIC NAVIGATION ARCHITECTURES FOR COORDINATION AND SCHEDULING OF HUMANS AND ROBOTS

DAVID ISERN, Dept. d'Enginyeria Informàtica i Matemàtiques, Universitat Rovira i Virgili (URV), Spain. disern@etse.urv.es
ROBERTA ANNICCHIARICO, Fondazione IRCSS Santa Lucia, Roma, Italy. r.annicchiarico@hsantalucia.it
JAVIER VÁZQUEZ-SALCEDA, Institute of Information and Computing Sciences, Utrecht University. The Netherlands. javier@cs.uu.nl
CRISTINA URDIALES, Dep. Tecnología Electrónica, ETSI Telecomunicación, University of Malaga, Spain. cristina@dte.uma.es
CARLO CALTAGIRONE, Fondazione IRCSS Santa Lucia, Roma, Italy.

ABSTRACT

We present a system to coordinate navigation for human and mobile robots to accomplish a given task. Coordination is supported by software agents, which handle global knowledge to assign tasks. Then, an agent in each mobile plans a route in a deliberate way and tracks it reactively. Thus, coordinated navigation is hierarchically decomposed into progressively simpler behaviors that can be adapted to the nature of the mobile. The system has been successfully tested for situations in a real hospital environment.

KEYWORDS: Bio-mimetic Navigation, Mobile Robots, Human/Robot Coordination, distributed scheduling, multi-agent systems.

1. INTRODUCTION

Navigation can be defined as the act of reaching a goal while avoiding collisions. In Bio-mimetic Systems, navigation is typically solved in a hierarchical way: while deliberative planning is performed at high level using knowledge about the layout of the environment, low-level navigation takes care of unexpected events in the path in a reactive way [1].

Navigation is usually an important part of any task involving mobile robots. When a given task is too complex for a single robot to accomplish, cooperative systems provide a solution by combining the actions of several robots. Cooperative Robotics is based on the synergy concept: the combined action of all robots is more efficient than the sum of their isolated behavior. In case of navigation, cooperative robots need to take into account the state (relative positions, tasks currently performed, etc.) of each robot with respect to the others and to the goal(s). This requires not only a local knowledge about the environment but also a global picture of the whole cooperative system. Consequently, Navigation Coordination is a new level that should be above Deliberative Planning and Reactive Navigation in the navigation hierarchy. It is important to note that this highest level does not need to take into account how each robot handles its own motion but only needs to provide partial goals to each component of the group.

In robotics, the simplest coordination mechanism consists of making each robot act only on its own benefit. In this case robots do not work together explicitly, but their combined action is valid for the task at hand [2]. When this mechanism fails for complex tasks, it is necessary to move on to intentional cooperation [3][4]. However, in the problem of coordinating humans and robots, we need more powerful coordination mechanisms. Software Multi-Agent Systems (MAS) are a good paradigm to integrate human and artificial entities. On one hand they provide a natural way of integrating humans in a distributed system by means of Software Agents that a) have a profile of the human interests, abilities and responsibilities and b) act on behalf of the human, representing it in the virtual world. On the other hand Coordination is one of the most explored problems in MAS, usually applying solutions coming from Game Theory or from Organizational Theory.

In this paper we present a MAS Coordination Layer to allow efficient coordination not only of robots but also of humans. As an example of structured domain, we apply this architecture in a complex hospital environment, in this case the Fondazione IRCSS Santa Lucia, in Rome, where human staff must be coordinated efficiently with electric wheelchairs (to move patients) and robotic platforms (to move material).

2. THE HOSPITAL ENVIRONMENT

2.1 Patient schedules and activities

Each patient has one session of physical therapy to the morning (40 min), and some have another one in the afternoon (40 min). All patients affected by Parkinson's disease have one individual session of physical therapy in the morning and one session in the afternoon as a working group (they move all at the same time from the rooms to the gym). Some patients may have also sessions of logaoedics rehabilitation, cognitive rehabilitation, phoniatrics rehabilitation (during the lunch), hydro-kinesis-therapy (in a swimming pool situated 50 meters from the unit) and/or respiratory rehabilitation (outside the unit). Usually the patients have their physical therapy in the gymnasium, except when the clinical conditions do not make it possible; in this case the physiotherapist go into the patient's room to execute the therapy at bed.

There is a special group of patients, the day-hospital patients. They are patients who do not sleep inside the hospital and only come into the hospital to attend their rehabilitation program.

2.2 Doctors' schedules

During the week, doctors A, B and E examine patients in the rooms on the long corridor[1]: patients in rooms 321-327 on Monday; those in rooms 301-308 on Thursday. Tuesday Dr. B observes his patients in the gymnasium and Friday Dr. A will make the same with his patients. The sequence of the rooms to be visited must obviously hold account of the timetables of the session of rehabilitation of the patients; therefore the doctors will not simply enter each next room. Nurses must hold account of this timetable to schedule morning activities and attend the patients in the proper sequence. The same happens for the other half of the unit. During the week, doctors C, D and F examine patients in the rooms on the short corridor: patients in rooms 309-316 on Tuesday; those in rooms 317-320 and DH patients on Wednesday. Monday Dr. C observes his patients in the gymnasium and Friday Dr. D will make the same with his patients. This complex scheduling of doctors' visits avoids excessive interference with the nurses' daily work (a group of doctors systematically examining the patients would slow the daily activities down, as activities in the room are suspended during the doctors' visit and at least one nurse must be present).

In case of an emergency, the first choice must be towards the nearer doctor; in case of equal distance, the choice must be towards the doctor who usually treats the sick patient. In any case, immediately after the first aid, the "usual" doctor or the foreman must be informed.

3. AGENT COORDINATION LAYER

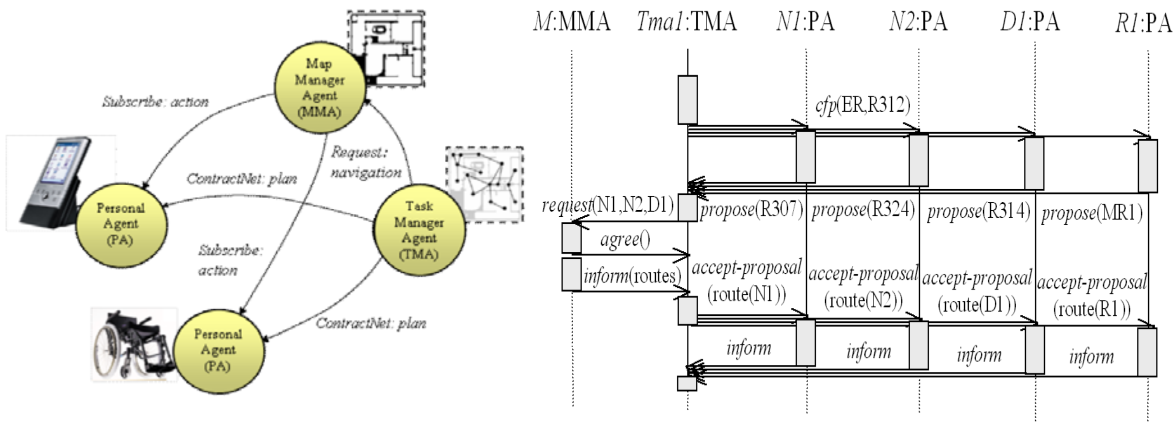
In our system there are different elements to coordinate: humans (doctors and nurses), robots (autonomous wheelchairs, robotic platforms) and services (emergencies, tasks). The first two types of elements are autonomous entities that must perform the services in a planned manner. All these elements have to be coordinated in an efficient way. As we mention in section 1, we propose the use of software agents' coordination techniques to solve this problem.

---

[1] In practice, the IRCSS unit must be divided into two halves (long Corridor and short Corridor).

3.1 Agents of the system

Each human and each robot has a *Personal Agent* (PA), which has information about the capabilities of the human/robot, its responsibilities, its current location and state (e.g., busy or free). The information is used by the *Task Manager Agent* (TMA) to schedule all the tasks to be performed, distribute them among humans and robots and coordinate them (e.g., a robotic platform delivering some material to a human in a given location) by giving them some partial goals (task, location, deadline). In the case of humans, the PA delivers the information to the person by means of a portable device (beeper, PDA, Mobile phone, table PC). In the case of robots, the PA builds the navigation plan that is then followed by the reactive layer of the robot. In order to maintain an updated map of the environment, the *Map Manager Agent* (MMA) receives information from the robots in the environment and keeps a metric and topologic map of the environment as proposed in [5], which is used by the TMA in its scheduling (e.g., assigning tasks to the entities closer to the goal) and the PA (to build the navigation plans).



(a) **(b)**

Figure 1. a) Architecture proposed and communication protocols used.
b) Inter-agent messages exchanged to handle an emergency

3.2 Communication and knowledge exchanged

In order to have a flexible interaction between the aforementioned agents, communication is the key element to be defined. Therefore, we have to create the *ontologies* defining all the terms agents use to communicate, and the *protocols* to be followed.

Ontology. An ontology storing definitions of the most important concepts involved in the system was designed using Protégé-2000 ontology editor [6]. We have two main ontologies: a) one defining all actions and predicates allowed between agents, (e.g., *give a plan*, *arrange a meeting* between PAs, etc.); b) to define the main concepts to map in our system (e.g., *doctor*, *nurse*, *robot*, *plan*, doctor's *personal data*, *wheelchair features*, *room*, *floor*, *hospital*, etc.). As we did not find in the repositories[2] an ontology covering all these concepts, we could only use several classes (with subclasses and slots) we found in the definition of both ontologies.

---

[2] As the design of a ontology can be a tedious task, there are some repositories available where researchers can obtain implemented ontologies in many domains such as healthcare, geographical information, etc. Some examples of ontology repositories are: **DAML** (*http://www.daml.org/ontologies*), **AgentCities** (*http://www.agentcities.net/property-list-one.jsp?propType=ontology*), **Web Ontology** (*http://students.washington.edu/kolbe/info320/projects/webontology/*) and **Protégé-2000** (*http://protege.stanford.edu/ontologies/ontologies.html*).

Communication protocols. These protocols define the rules guiding the data exchange between agents. We follow FIPA[3] specifications [7] to design our communication protocols and a tool to develop MAS as JADE[4] that implements these protocols. Fig. 1.a shows the agents proposed and both their data and FIPA protocol used. In our approach, the TMA monitors the main tasks of the system and manages all events of the system such as emergencies, scheduled tests, etc. When an event arises, it negotiates a plan with the concerning PAs using a Contract Net protocol. The TMA coordinates the event and sends the request to the MMA in order to calculate all navigation plans; moreover, it identifies possible conficts between these plans. Each navigation plan is finally sent to the concerning PA. In Fig. 1.b are shown the sequence of messages transmitted to manage an emergency. The architecture proposed is simple but flexible in the sense that TMA and MMA cover a specific domain, for instance a floor in a hospital, or an area, etc., and we can deploy a network of these agents that coordinate their tasks.

4. BIO-MIMETIC NAVIGATION

The navigation hierarchy proposed in [1] presented a hierarchical decomposition of high level navigation behaviors into progressively less complex layers, where each layer includes all the ones below. Top layers in the hierarchy are deliberative, while bottom ones are reactive. It was stated that both robotic and animal navigation mechanisms follow this decomposition, but while humans operate in a bottom-up way, robots usually work in a top-down one. While vertebrates are intelligent enough to cover the whole navigation hierarchy, the main problem with robots is that the top level, survey navigation, involves navigating through unexplored areas, if necessary. In order to work efficiently, deliberative navigation usually deal with topological models of the environment, which are compact and easy to process. However, unexplored areas do not appear in topological maps and, hence, can not be included in top planning problems. Alternatively, navigation through unknown environments can be reactively handled, but in such case the robot does not achieve survey navigation but goes back to lower levels. In [8] a new environment model where the topological representation extracted from the metrical one was used to solve this problem and, thus, achieve survey navigation in robots. Using the navigation hierarchy, a common navigation framework is set for humans and robots. However, this hierarchy includes no upper level for coordinated navigation. As aforementioned, in this paper we propose such a layer. While coordination is achieved by agents as described above, in this section we present the different layers supporting navigation.

4.1. Top level: coordination

The top level of the hierarchy requires both global knowledge about the environment and also about the position of all available PAs. Whenever a task x is available, the TMA decides how many PAs need to be involved and which places $P_x=(P_1, P_2, \ldots P_N)$ need to be visited and then broadcasts a request. Then, PAs either available at the moment or in a short time period send their positions and time to availability to the TMA. When the TMA has received all this information, it shortens the list of potential PAs involved in x $PA_x=(PA_1, PA_2, \ldots PA_n)$ regarding all defined constraints. Then, in order to decide who does what, the following algorithm is used:

1. Calculate an efficient route R to visit all PA_x for each PA using a fast Traveling Salesperson Problem (TSP) solver.
2. Calculate all distances involved to cover each route $R(PA_i)$, $(d_{1PAi}, d_{2PAi}, \ldots d_{N\text{-}1PAi})$, in the work environment.

[3] Foundation for Physical and Intelligent Agents (FIPA) is a non-profit organisation aimed at producing standards for the interoperation of heterogeneous software agents.
[4] JADE or the Java Agent DEvelopment Framework (http://sharon.cselt.it/projects/jade/) is a software framework through a middle-ware that complies with the FIPA specifications.

3. Find the PA presenting the route whose first stretch is shorter, $min(d_{1PAi})$, PA_k and check which place P_j is first visited in such a route.
4. Remove P_j from P_x for all PAs except PA_k and return to step 1 until all places in P_x have been visited by at least a PA.

This algorithm can be easily adapted to different constraints, like requiring two agents at a place at once, but in the current version it simply grants that all places in P_x are visited by a PA. Some PAs can visit several places while other might visit only one. As a result, the TMA sends to each PA in PA_x a list of the places it must visit to globally achieve a low task execution time.

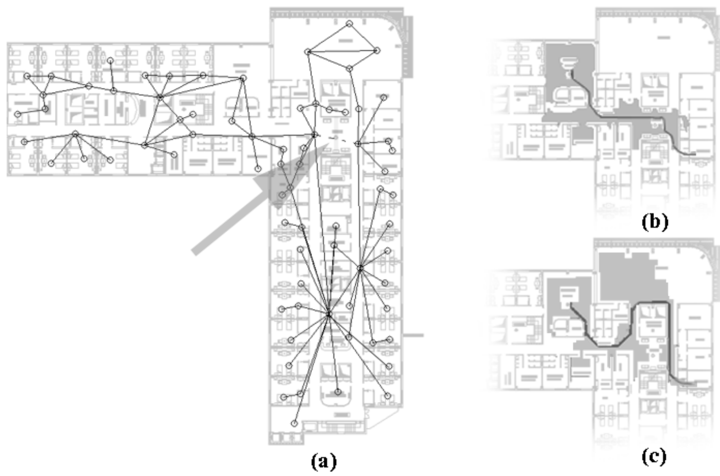


Figure 2. a) IRCSS unit map with the topological map overimposed, being updated of a door now closed. b) Previous path planned for doctor B. c) MMA plans a new path.

.

4.2.Medium and bottom level: planning and tracking

When a PA receives a list of places to visit, deliberative planning to move between each two places is performed at topological level, using the structure proposed in [5].In this structure each topological node is linked to a region of a metric map (Fig.2.a) so that geometrical information is preserved. Thus, a path can be calculated in a fast way and then propagated into a path region at metric level (Figs.2.b-c). Such a region can be thinned into a line *L*. Then, the points of inflection of such a line are extracted $(L_1, L_2, \ldots L_i)$. These points are the locations where the mobile should change its heading to follow path *L*. Rather than exhaustively tracking *L,* which might become non valid in a dynamic environment at any moment, $(L_1, L_2, \ldots L_i)$ become partial goals to the mobile, which should try to reach them in a reactive way using the purely reactive module proposed in [5]. If a significant change in the environment is detected by any mobile, the deliberative layer is retriggered and new partial goals are set for mobiles if necessary (Fig.2.c). To keep track of changes in a fast and efficient way, the model of the environment is centralized so that mobiles need to ask for deliberative plans to the MMA, which is also in charge of updating the model with all data received from PAs.

5. RESULTS

This section presents a simple example of the proposed system. In this example, an emergency is detected in room 312 (R312). The TMA agent requests data to the staff's and robots' PA about availability. There are only two nurses available at R307 (N1) and R324 (N2) and a doctor (D1) at R314. There is a robot (R1) available at Medical Room 1 (MR1). TMA agent also knows that the emergency trolley and some equipment are at MR1. The defibrillator and the rest of the equipment are at Nurse Room 2 (NR2). Medical records are also at NR2 at night. The TMA requests the updated topographic map from the MMA and then executes three runs of the TSP to visit all required places. The final route for D1 is NR2 (180) and R312 (113), for N1 is MR1 (86) and R312 (608) and for N2 is R312 (634), where the traveled distances to each location are included. The resulting paths are presented in Fig. 3.a-c. The first agent to arrive to a room is N1 and the room is MR1, where R1 is. Since R1 is meant to transport equipment, it simply waits for N1 to arrive and then moves to NR2 to gather more equipment. Thus, its route is NR2 (519) and R312 (89) (Fig. 3.d). It is important to note that, in this case, D1 is expected to arrive to NR2 much earlier than R1, who has to wait for N1 and track a longer path and, consequently, has an equivalent traveling time of 605 against 180 for D1. Thus, it might be preferable to redirect N2 to NR2 (522) and send D1 directly to R312. Once the plan is constructed, the TMA sends it to the PA's. Each PA asks for a navigation plan to the MMA. In this example no PA finds a change in the environment to be notified to the MMA.

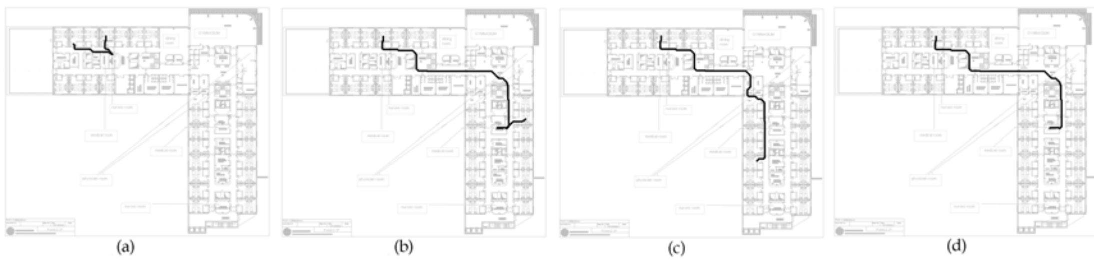


Figure 3: Paths for emergency in R312 for: a) D1; b) N1; c) N2; d) R1

6. CONCLUSIONS

In this paper we present a simple but effective agent coordination layer to coordinate humans and robots in complex environments. The system suggests to the entities not only the tasks to be done but also the best path to go from one place to the other, keeping updated metric and topographic maps of the environment.

As part of our future work, we will explore the duplication of the main agents (MMA, TMA) to increase the safety, soundness and decentralization of the system. For instance, cloning the TMAs and coordinating complex tasks between them. If connection to one of the TMA's is lost, others will take its role and the system will rapidly recover from the problem.

6. REFERENCES

[1] Franz, M.O. and Mallot, H.A., "Biomimetic robot navigation", Robotics and autonomous systems, 30, pp. 133-153, 2000

[2] Fukuda, T., Nakagawa, S., Kawauchi Y., and Buss, M., "Self organizing robots based on cell structures - CEBOT", Proc. of IEEE International Workshop on Intelligent Robots and Systems (IROS'88), pp. 145-150, 1988.

[3] Sycara, K., Decker, K., Pannu, A., Williamson, M. and Zeng, D., "Distributed intelligent agents", IEEE Expert, 11(6), pp. 36-46, 1996.

[4] Martin, D., Cheyer, A. and Moran, D., "The Open Agent Architecture: A framework for building distributed software systems", Applied Artificial Intelligence, 13 (1-2), 1999.

[5] C. Urdiales, E.J. Pérez, F. Sandoval and J. Vázquez-Salceda, "A hybrid architecture for autonomous navigation using a CBR reactive layer", Proc. of 2003 IEEE/WIC Int. Conference on Intelligent Agent Technology (IAT 2003), pp 225-232, 2003.

[6] N. F. Noy, M. Sintek, S. Decker, M. Crubezy, R. W. Fergerson, & M. A. Musen. "Creating Semantic Web Contents with Protege-2000". IEEE Intelligent Systems 16(2):60-71, 2001.

[7] FIPA Specification repository. http://www.fipa.org/repository/index.html

[8] E. Pérez, A. Poncela, C. Urdiales, A. Bandera y F. Sandoval, "Survey navigation for a mobile robot by using a hierarchical cognitive map", Cognitive Processing 3(3-4), pp.99-106, 2002