Control Architecture design for a Mobile Robot via the Internet

A. Sayouti[\$], F. Qrichi Aniba[\$], H. Medromi[\$]
[\$]Equipe Architecture des Systèmes, ENSEM, BP 8118, Oasis, Casablanca, Morocco.
sayouti@gmail.com, qrichi_f@yahoo.fr, hmedromi@yahoo.fr

***Abstract*— Controlling robotic devices from remote places, using an existing, wide spread network has significantly extend their application. Internet, considering its wide spread and accessibility, is an ideal media for data transmission between a robot and its operator, but the limitation in bandwidth, the notable delay, and the unreliability of the network introduces several problems. The solutions, proposed within the framework of telerobotics to face the limitations of Internet, are founded on the autonomy and the intelligence granted to the robot in order to interact with its environment and to collaborate with the remote user. This paper describes a system framework and a control architecture based on multi-agents systems. An illustration of our architecture is given in an application of control of an autonomous mobile robot developed by the team architecture of systems, to the ENSEM of Casablanca.**

Index Terms— Multi agents systems, Intelligent control, Internet, control architecture, distributed system architecture.

I. INTRODUCTION

Internet telerobotics is a new robotics field which attracted much attention among researchers in the last few years. The term internet telerobotics refers to a robot system remotely controlled and monitored through the Internet. The growing interest in this field is stimulated by the advancement of Internet, which provides access to various computing resources virtually from everywhere in the world. Increasing application of Internet as communication media for telerobotics system also comes from the fact that it uses standard communication protocol, and that the physical media is readily available for telerobotics application, eliminating the need for developing a dedicated, proprietary and expensive communication system. Internet telerobotics differs from conventional telerobotics in that they are available, when desired, to the general public via the Internet. It allows people to actively participate in remote exploration in various applications. Areas where internet telerobotics is predicted to be useful include entertainment, telemaintenance [1], telemanufacturing [2], telemedicine, teleoperation and mining. It has tremendous implications for education and training, allowing students and trainers to actively explore remote environments. It also has implications for research as well, where laboratories can share access to expensive resources.

During the Nineties, several projects appeared of robotic systems control, using Internet as communication network [3] with various objectives: the Mercury project to prove the feasibility by Goldberg and al. [4], the Australian telerobot for the interaction with the user by Taylor and al. [5], Rhino by Burgard and al [6], Xavier by Simmons [7], Puma-Paint by Stein [8], mobile robotics KhepOnTheWeb by Saucy and al [9], increased reality by Otmane Ariti [10], etc. This article is presented as follows: on the next section, we describe the system framework. In section 3, we propose our control architecture. Then we present an application of control via internet of a Lego mobile robot. Finally, some conclusions are presented in section 5.

II. THE PROPOSED SYSTEM FRAMEWORK

There are now numerous architectures available for implementing a web-based robotic system. One of the popular architecture used is the CGI (Communication Gateway Interface) mode. Through the HTML (Hyper Text Markup Language) form, a request can be passed from client to server to launch a progress to perform some predetermined actions in the server. This mode has some shortcomings, such as poor performance, short of interaction and stateless protocol. Another is based on Java technology and can avoid the limitations of CGI [11]. Because Java program is executable within a web page, it requires less bandwidth and enables an interface to be interactive instead of being static. But the Java applet suffers from the connection quantities along with the number of servers and clients. In addition, there are security restrictions associated with Java such as only allowing applets to connect to the host they were served from.

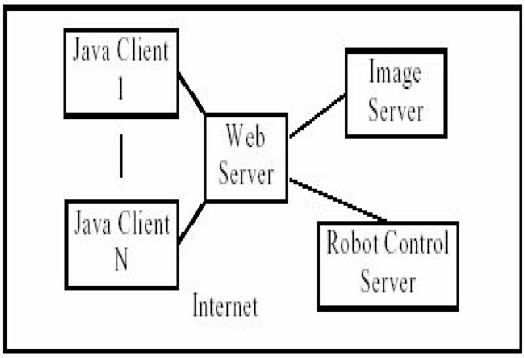

Fig 1. System Architecture

A more flexible and extendable approach is to use the central server architecture as shown in Figure 1. All the clients and servers are connected to a central web server, and only need to know the location of the web server and communicate with each other through the web server. With this architecture, we can either put all image service, robot control service and web service in one computer or put them in several computers and connect them with TCP sockets. With this architecture, it is also very easy to add more computers for robot control and image processing or for multi-robot control.

Internet doesn't provide a guaranteed Quality of Service (QoS), it entitles a number of limitation and difficulties, such as bandwidth constraint, transmission delays, packet lost, connection rupture etc. The situation above influence the performance of Internet based telerobotics systems. The solutions, proposed to face the limitations of the communication channel, are based on the autonomy and the intelligence granted to the robot [12]. To design an autonomous robot implies to design a control architecture, with its elements, its definitions and/or its rules [13].

III. CONTROL ARCHITECTURE

The organization of a robotic system - or its control architecture - determines its capacities to achieve tasks and to react to events. The control architecture of an autonomous robot must have both decision-making and reactive capabilities: situations must be anticipated and the adequate actions decided by the robot accordingly, tasks must be instantiated and refined at execution time according to the actual context, and the robot must react in a timely fashion to events. This can be defined as a rational behavior, measured by the robot's effectiveness and robustness in carrying out tasks. To meet this global requirement, a robot control architecture should have the following properties:

- Programmability: a useful robot cannot be designed for a single environment or task, programmed in detail. It should be able to achieve multiple tasks described at an abstract level. The functions should be easily combined according to the task to be executed.
- Autonomy and adaptability: the robot should be able to carry out its actions and to refine or modify the task and its own behavior according to the current goal and execution context as perceived.
- Reactivity: the robot has to take into account events with time bounds compatible with the correct and efficient achievement of its goals (including its own safety).
- Consistent behavior: the reactions of the robot to events must be guided by the objectives of is task.
- Robustness: the control architecture should be able to exploit the redundancy of the processing functions. Robustness will require the control to be decentralized to some extent.
- Extensibility: integration of new functions and definition of new tasks should be easy.

Robot control architectures being at the core of the design of autonomous robots, many authors have addressed - and still do - this issue. The approaches differ in several manners, sometimes by the philosophical standpoint itself: some projects aim at imitating living beings, while others are more AI oriented. The first trend of research sought inspiration from biology and ethology and has mainly yielded stimuli-response based systems. The second trend tried to use symbolic representations and some reasoning capacities.

A. *Brief overviews of control architecture*

One of the first authors who expressed the need for a control architecture was R.A. Brooks [14]. In 1986, he presented an architecture for autonomous robots called "subsumption architecture". It was made up of various levels which fulfil separately precise function, processing data from sensors in order to control the actuators with a notion of priority. It is a reactive architecture in the sense that there is a direct link between the sensors and the actuators. This architecture has the advantage to be simple and thus easy to implement, nevertheless, the priorities given between the different actions to perform are fixed in time and offer a limited flexibility. Then other various architectures were developed based on different approaches, generally conditioned by the specificity of robotic application that the architecture had to control:

- the architecture 4-D/RCS developed by the Army Research Laboratory [15] has the main characteristic to be made up of multiple calculative nodes,
- the CLARAty [16] proposed by the Jet Propulsion Laboratory developed in collaboration with NASA where one of the interests of this representation is to work at the decisional level only on one model emanating from the functional level,
- the LAAS architecture (Laas Architecture for Autonomous System) [17] is made up of 3 levels: decisional, executive and functional,
- R.C. Arkin describes and uses a hybrid control architecture, called AuRA for Autonomous Robot Architecture [18], including a deliberative part and a reactive part,
- A. Dalgalarrondo [19] from the DGA/CTA presents a hybrid control architecture including four modules: perception, action, attention manager and behavior selector,
- the DAMN [20] architecture results from work undertaken at the Carnegie Mellon University in response to navigation problems. Multiple modules share simultaneously the robot control by sending votes which are combined according to a system of weight attribution,
- Kim Hong-Ryeol & Al patented a five hierarchical level architecture in 2005 [21]. All these architectures show the diversity of the approaches mainly due to the robotic applications or the topics of the designers.

B. *Proposed control architecture*

We propose a hybrid control architecture, called EAAS for EAS Architecture for Autonomous system, including a deliberative part (Actions Selection Agent) and a reactive part. It is made up of two parts, each using distinct method to solve problems (Fig. 2). The deliberative part which uses methods of artificial intelligence contains a path planner, a navigator and a pilot. The reactive part is based on direct link between the sensors (Perception Agent) and the effectors (Action Agent). Fundamental capacities of our architecture encompass autonomy, intelligence, modularity, encapsulation, scability and parallel execution. To fulfil these requirements, we decided to use a multi agents formalism that fits naturally our needs [22]. The communication between agents is realized by messages. Object oriented language is therefore absolutely suited for programming agents (we chose java). We use threads to obtain parallelism (each agent is represented by a thread in the overall process).

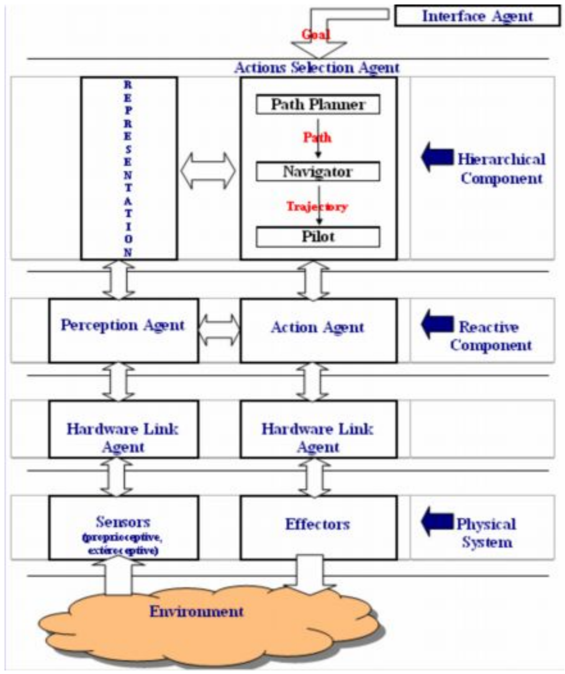

Fig 2. EAAS Architecture

EAAS architecture consists in five agents: interface agent, actions selection agent, perception agent, action agent and hardware link agent. The interface agent is the high level of our control architecture. It must generate a succession of goal, or missions for the actions selection agent, according to the general mission of the robot. It is the "ultimate" robot autonomy concept: the robot generates itself its own attitudes and its own actions by using its own decisions. The perception agent manages the processing of incoming data (the sensor measurements) and create representations of the environment. The actions selection agent must choose the robot behavior according to all information available and necessary to this choice: the fixed goal, representations and the robot localization. The action agent consists of a set of behaviors controlling the robot effectors. The hardware link agent is an interface between the software architecture and real robot. Changing the real robot require the use of a specific agent but no change in the overall architecture.

IV. EXPERIMENT

The aim of this project is to allow users to perform remote control experiments on a mobile robot through the Internet. In particular, the robot is made of Lego Mindstorms bricks. As shown in Figure 3, our Lego mobile robot was controlled to follow a given trajectory and to push a ball to the goal.

The tools chosen to develop this platform are java and classical internet browser. The choice of java is based on its well-known portability and web technology suitability. The communication between the client and the server is based on socket using TCP protocol.

Fig 3. User Interface

The user can directly control the Lego robot by clicking the start button on the control panel. The image display applet shows the visual feedback in a continuous jpeg image. The forum service allows users to send messages to each other, private or broadcast in order to interchange their ideas over the remote control subject. The remote user is invited to test the connection using the statistical or the dynamical way, before or during taking the control by clicking on the buttons labelled statistic or dynamic test respectively. This user interface allows students to undergo a distance learning with the opportunity to test their ability on line.

V.CONCLUSION

In this paper we have proposed an architecture for a telemanipulation system using internet as communication support and presented a control architecture based on multi agents systems to be able to manage the lack of quality of services in the context of remote control.

In a close future, Internet remote control will become a must in industrial contexts if we are able to propose reliable solutions, mainly in terms of network safety.

REFERENCES

[1] A. Bicchi, A. Coppelli, F. Quarto , L. Rizzo, F. Turchi and A. Balestino. Breaking the lab's walls telelaboratories at the university of pisa. In Proceedings of the 2001 IEEE International Conference on Robotics and Automation, pages 1903 – 1908, Seoul, Korea, May 2001.

[2] P. Le parc, J. Vareille and L. Marce. E-productique ou contrôle et supervision distante de systèmes mécaniques sur l'internet. Journal européen des systèmes automatisés (JESA), v.38, n. 5, p. 525-558, 2004.

[3] K. Goldberg and R. Siegwart, Beyond Webcams : an introduction to online robots. The MIT Press, 2001.

[4] K. Goldberg, S. Gentner, C. Sutter and J. Wiegley. The mercury project: a feasibility study for internet robotics. IEEE Robotics & Automantion Magazine, pages 35--40, March 2000.

[5] K. Taylor and B. Dalton. *Issues in internet telerobotics*. FSR'97 International Conference on Field and Service Robots, 8-10 Décembre 1997.

[6] D. Schulz, W. Burgard and A. B. Cremers. *Predictive simulation of autonomous robots for tele-operation systems using the world wide web*.In IEEE/RSJ International Conference on Intelligent Robots and System, Victoria, B.C., Canada, October 1998.

[7] R. Simmons. *Xavier : An autonomous mobile robot on the web*. In In International Workshop On Intelligent Robots and Systems (IROS), Victoria, Canada, 1998.

[8] M.R. Stein. *Painting on the world wide web : the pumapaint project*. In Proceeding of the IEEE IROS'98 Workshop on Robots on the Web, pages Victoria, Canada, October 1998.

[9] P. Saucy and F. Mondada. *Khepontheweb: Open access to a mobile robot on the internet*. IEEE robotics and automation magazine, pages 41-47, March 2000.

[10] S. Otmane, M. Mallen, A. Kheddar and F. Chavand. *Active virtual guide as an apparatus for augmented reality based telemanipulation system on the internet*. In IEEE Computer Society "33rd Annual Simulation Symposium ANSS 2000", pages 185-191, Wyndham City Center Hotel, Washington, D.C., USA, April 2000.

[11] A. Sayouti, F. Qrichi Aniba., H. Medromi,, A. Lakhouili. "Applying the Interaction-Oriented Approach to Remote Control of Robotic Systems". Conferência Ibérica de Sistemas e Tecnologias de Informação (CISTI). Porto. Portugal. 2007.

[12] A. Sayouti, F. Qrichi Aniba., H. Medromi,, . "Remote Control over Internet of Robotic Systems Based on Multi Agents Systems". Second IEEE International Conference on Research Challenges in Information Science (RCIS), Marrakech, Morocco, 2008.

[13] Cyril Novales, Gilles Mourioux, Gerard Poisson. A multi-level architecture controlling robots from autonomy to teleoperation. First National Workshop on Control Architectures of Robots. Montpellier. April 6,7 2006.

[14] R.A. Brooks.A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, vol. 2, n°. 1, pp. 14–23, 1986.

[15] Albus J. S. 4D/RCS A Reference Model Architecture for Intelligent Unmanned Ground Vehicules, Proceedings of the SPIE 16th Annual International Symposium on Aerospace/Defense Sensing, Simulation and Controls, Orlando, FL, April 1 – 5, 2002.

[16] R. Volpe and al. CLARAty: Coupled Layer Architecture for Robotic Autonomy." *JPL Technical Report D-19975*, Dec 2000.

[17] R. Alami, R. Chatila, S.Fleury , M.Ghallab and Ingrand F. An architecture for autonomy. *The International Journal of Robotics Research*, Special Issue on Integrated Architectures for Robot Control and Programming, vol. 17, no 4, pp. 315-337, 1998.

[18] R.C. Arkin. Behavior-based Robotics. MIT Press, 1998.

[19] A. Dalgalarrondo. Intégration de la fonction perception dans une architecture de contrôle de robot mobile autonome. Thèse de doctorat, Université Paris-Sud, Orsay, 2001.

[20] J. Rosenblatt. DAMN: A Distributed Architecture for Mobile Navigation. In proceedings of the 1995 AAAI Spring Symposium on Lessons Learned from Implemented Software Architectures for Physical Agents, AAAI Press, Menlo Park, CA, 1995.

[21] K. Hong-Ryeol, K. Dae-Won, P. Hong-Seong, K. Hong Seok and L. Hogil. Robot control software framework in open distributer process architecture. Korean and World patent, WO2005KR01391 20050512, IPC1-7: G06F19/00, May 2005.

[22] J. Ferber: Les systèmes multi-agents, vers une intelligence collective,.Paris, InterEditions, 1995.