SIMULATING TELEROBOTICS BY CELLULAR TELEPHONY

Rodrigo Montúfar-Chaveznava, Crisóforo Paisano
Computer Science Department,INAOE, Luis Enrique Erro 1, Tonantzitla,Puebla, Mexico

José María Cañas
Robotic Labs (GSyC), ESCET, Universidad Rey Juan Carlos, C/Tulipán s/n, Móstoles, Madrid, Spain.

Keywords: Telerobotics, GSM, Simulation, Cellular telephony.

Abstract: In this work we present a simulation system to control a mobile robot using a cell phone. We exploit the cellular telephony technology to communicate and command a robot provided with a modem in a simulation environment. The system considers telerobotics can be performed using phone tones or sending text messages via SMS. Due the cellular telephony is expensive, the simulation system give us the possibility to experiment infinitely this novel way of telerobotics without expending a lot of economical resources. We design all components required for a basic cellular telephony system and we employ a robotics simulator for the Pioneer mobile robot, expecting to translate the system primitives to real systems in a near future. The idea comes up due that cellular telephony has an enormous covering, and taking advantage of such situation, telerobotics could be performed in places where wireless networking and power sources are not available at all such as the country.

1 INTRODUCTION

Telerobotics is related to robot autonomy. In this field, humans have to execute several tasks, such as planning, world perception and manipulation. This mean, there is somebody acting in real-time as part of the robot control system. However, we need to satisfy some requirements in the control system and they are not always available, especially when we are working with mobile robots.

Telerobotics is a response of the necessity of carry out operations where it is difficult to put a human being due to constraints such as cost, safety or time. Besides, telerobotics demands reliable navigation because the environment where robots operate is usually unknown, dynamic or unstructured and the erratic navigation can guide us to an unpredictable performance and system failures or accidents. Telerobotics also requires of the efficient generation of action commands because the success on the task depends on the robot behavior. Moreover, telerobotics claims for localized sensor information, particularly when activities such as map construction and registration are necessary.

Actually, telerobotics has extended to Internet and it is accessible to almost everybody, it provides user interfaces easy to use. However, networking is not available everywhere, and this fact limits the telerobotics expansion.

Considering that cellular telephony has a bigger coverage than Internet, the introduction of this kind of technology in robotics is considered an alternative to continue expanding the use of telerobotics to places where it is considered a solution or assistance.

In this work, we present a simulation system to carry out telerobotics using a cell phone. Actually, we have not found any reference about the use of cell telephony in robotics. During the development of the work, we have considered all aspects related to communication in cellular telephony and we have considered the use of a Pioneer robot simulator, the idea is to translate the primitives of the system to real Pioneer robots very soon.

2 CELLULAR TELEPHONY

In last years, we have been witness of the birth and development of wireless technologies. Cellular telephony is the one that has had the most important development. At the beginning, cellular telephony was conceived for voice communication; however,

Montúfar-Chaveznava R., Paisano C. and María Cañas J. (2005).
SIMULATING TELEROBOTICS BY CELLULAR TELEPHONY.
In *Proceedings of the Second International Conference on Informatics in Control, Automation and Robotics - Robotics and Automation*, pages 329-334
DOI: 10.5220/0001167703290334

nowadays it is able to provide a diversity of services, such as data, audio and video transmissions with some restrictions. At present, we identify five generations of development in cellular telephony (Ibrahim, 2002).

First Generation (1G). It was designed in 1970, but implemented until 1984. It was completely analog and employed for voice communication.
Second Generation (2G). 2G was designed in 1980, and introduced in 1991. It was digital instead of analog. The 2G system employs sophisticated coding protocols and actually is in use.
Second and half Generation (2.5G). It was designed in 1985 and introduced in 1999. 2.5G has a higher capacity than previous versions and employs packetized data.
Third Generation (3G). It was designed in 1990 and implemented in 2002. 3G has the ability to contain high voice and data convergence with wireless access to Internet; and consequently it is suitable for multimedia. 3G protocols produce high-speed data transmission; they are designed for specialized applications such as MP3 audio, video in movement, videoconference and high-speed access to Internet.
Fourth Generation (4G). The 4G design began in 2000 and we expect it will be totally implemented in 2010. 4G will be completely IP-oriented, with higher capacity, multimedia and transmissions of hundreds of megabits.

2.1 The Cell Phone

Cell phones are considered very sophisticated radios (Yucatan, 2001). We can compare the cell phone with the short-wave radio to figure out such sophistication. A short-wave radio allows communicating two persons using the same frequency, and only a person can talk in turn. Whereas a cell phone is dual, it uses a frequency to talk and other to listen. A short-wave radio has 40 channels. A cell phone can use 1664 channels. Cell phones operate in regions named *cells*, which can be alternated following the phone displacement. The cells provide a high scope to phones, considering a short-wave radio can only transmit until five miles. The great idea concerning cell phones is that a region can be divided in small cells extending the communication frequency to the entire region.

2.2 Telephone Tones

The Dual-Tone Multi-Frequency (DTMF), also known as *touch tone* dialing, is employed to transmit telephone signals over the voice frequency band.

Previous to DTMF, the telephone systems employed a sequence of clicks to dial numbers (pulse dialing). The clicks were the beginning or end of a connection on the line phone, and they were equivalent to turn on or off a switch.

DTMF was developed in Bell laboratories to use long distance numbers and establish wireless connections via microwaves or satellite. Encoders and decoders were included in the terminals to translate standard pulses in DTMF tones and vice versa. The *touch-tone* system introduced a standard keyboard. The DTFM designers considered the possible use of phone lines for computer access, then they included some additional keys such as # and *.

The DTMF keyboard is a 4×4 matrix, each row corresponds to a specific low frequency, and each column to a specific high frequency. When a key is pushed, a sinusoidal tone is produced containing high and low frequencies. The tones are evaluated to determine the pressed key. Frequencies were designed using a ratio of 21/19, which is barely lower than an entire tone, to avoid harmonics or natural frequencies that could appear when two tones are transmitted. Table 1 presents the keyboard correspondence to DTMF values (Fogelklou, 1999).

Table 1: DTMF Frequencies

1	2	3	A	697 Hz
4	5	6	B	770 Hz
7	8	9	C	852 Hz
*	0	#	D	941 Hz
1209 Hz	1336 Hz	1477 Hz	1633 Hz	

2.3 Global System for Mobile (GSM)

GSM has some characteristics, which makes it different in the mobile communications universe (Rahnema, 1993). It was created in 80's as result of the cooperation between many European countries. GSM shares common elements with other mobile communication technologies as voice and data digital transmission, or the cellular topology. A GSM network has three basic elements: the terminal or cell phone, the base station and the network or node subsystem. Additionally, there are centers of operation established to monitor the network status.

The terminal has the subscriber identity module card (SIM), employed to identify a terminal when it connects to the network. The SIM gives mobility to the user, allowing the access to the network services, independently of the terminal or location. There is a unique number to identify every terminal, the international mobile equipment identity (IMEI), which is independent of the SIM.

The base station controls the connection between terminal and network. It is known as the *cell*, because it covers a certain geographic region. The base station subsystem (BSS) has a base transceiver station (BTS), and a base station controller (BSC). Every BSS has one or more BTSs. The BTS houses the equipment for transmission and reception and negotiate the protocols with the terminal. There are more BTS in urban zones than in the country, and some times broadcast equipment is employed to guarantee the service. The stations use digital techniques to allow many users connect to the network; this negotiation is called *multiplexing*.

The BSC manage the resources of one or more BTSs. Some functions of the BSC are the handoff, the establishment of the channels to be employed and the frequency switching. The BSC establishes the connection between the cell phone and the mobile service-switching center (MSC), which is the core of the GSM system.

The MSC is the network core, where a cell phone is connected to networks such as the public switched telephone network or the integrated services digital network. The node where the MSC is located has equipment to control some functions such as security, messenger service or service collecting.

2.4 Short Message Service (SMS)

GSM provides the transmission and reception of short text messages (SMS), where two classes are specified: Point to point (SMS/PP) and Cell broadcast (SMS/CB).

SMS/PP allows sending a message form a GSM phone to another. SMS/CB allows sending one or more messages at the same time to all GSM phones located in certain zone. SMS/CB can have at most 93 characters, but it can link up to 15 messages to produce a macro message. Each SMS/CB message is assigned to a category where the information and the language employed are classified. In this way, it is possible to read selectively or discard the messages.

SMS is a protocol with no-connection. In fact, during transmission, a connection between transmission and receptor is not produced.

Sending a SMS/PP message from a GSM phone to another is considered a two steps operation: the message transmission from the cell phone to a special network entity, the short message service center (SMSC), and from here to the cell phone receptor. The first operation is called SMS mobile originated (SMS-MO), and the second one is called SMS mobile terminated (SMS-MT).

SMS-MT allows the reception of text messages containing up to 160 characters in the screen of the GSM cell phone. SMS-MO allows the transmission of messages containing up to 160 characters to another GSM terminal, fax, modem or e-mail Internet address.

The success of SMS depends, in one hand, on the simplicity and facility to use it and, in the other hand, on the sensation of presence in the other side of the phone. Both factors have produced such success, in spite of the costs and limitations in communication.

3 ROBOT SIMULATORS

Due the novelty that robotics was sometime ago; actually we find many definitions for robot. Some definitions, obtained form different sources are: (1) Machinery controlled electronically, capable to move and execute automatically different actions, according an established program; (2) a machine, which apparently mimics behaviors and actions of people; (3) a machine, which acts automatically as a response to its environment; (4) a handful of motors controlled by computer software; or (5) a robot is a computer with muscles. These definitions are very different from the original meaning of robot (*robota*): force or work.

The introduction of microprocessors in 70's makes possible the progress in robot technology. The modern computers provide the *brain* to the muscles of mechanic robots. Nowadays, robotics is the symbiosis between electronics and mechanics

An acceptable robot for research is not usually cheap. It is necessary to have resources to get one or a colony and for maintaining. A robot is not always accessible to anyone, and when programming, sometimes the results can be unexpected. Then, robot simulators are a solution to overcome these problems.

At present, the robot Pioneer (Fig. 1) is one of the most employed, and we can have access to some Pioneer robot simulators to tests ideas, theories, programs and algorithms, without necessity of the robotic body. Some of the most common simulators for the Pioneer are:

Saphira (Konolige, 1997). It is considered a mobile robot control architecture. It was developed for research purposes and programming in the robot Flakey at SRI in 1994. Saphira is divided in two modules. The low-level routines are organized and implemented as individual software: *Aria*. Saphira is designed and supported by ActivMedia Robotics (ActivMedia, 2005). The system is based on a set of C++ classes. The structure of classes makes easy to expand and develop new programs adding new sensors when it is required. Saphira and Aria are considered two different architectures, one constructed over the other. Aria consists of a set of communication routines programmed to control the robot from the server in a computer. The system architecture is designed to define easily robot applications using programs. Saphira is an open architecture, available to everyone who wishes to write its own control system without worries about complexities in hardware control. Communication is an advantage of micro-task properties and reflex of the architecture status.

Stage/Player/Gazebo (Gerkey, 2003)**.** Player was developed in the Research Robotics Laboratories at the University of Southern California. Player is a free software; it is a multithread server of robot devices, which provides a simple and complete control of sensors and actions to execute in the robot. When Player is running, the control program client is connected to Player through a TCP socket, and the communication is performed sending and receiving messages translated as commands or status information. Player is designed as an independent language and platform. The program client can run in any machine with network connection. The client can be written in any language to open and control TCP sockets. The client characteristics are available in C, C++, TCL, LISP, Java and Python. Player is designed to virtually support any quantity of clients.

Java Mobile Robots – JMR (Gallardo, 2003)**.** JMR is a client-server architecture, which can support many robots. The robots are identified by an independent name. JMR can run different processes in any robot, and connect to the Saphira simulator to communicate to a real robot. The client controls devices such as sonars and cameras; it also controls speed and reads the distance from the robot to any obstacle; it is also possible to know the robot status, and we can dictate the distance to travel and the degrees to rotate. The server receives and simulates the commands sent by the client. In the server we can load the worlds where the robot will navigate, and read the images captured by the robot cameras, such images can be sent to the client and displayed.

Figure 1: The holonomic robot Pioneer 2DX

4 THE SYSTEM SIMULATOR

The system (Fig. 2) consists of the mobile robot simulator (*SimRobot*), corresponding to a modified version of JMR, and the basic GSM cellular telephony simulator (*SimGSM*). SimGSM has three modules: the GSM cell phone (*M1*), the GSM modem (*M2*) and the SMS/Phone center (*M3*).

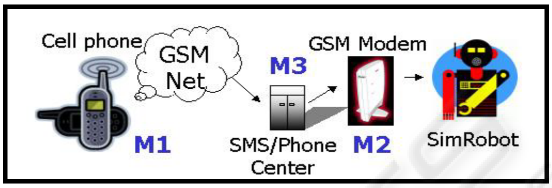

Figure 2: The SimGSM system

M1 is employed to communicate and control the robot via the telephone network. The robot control can be performed sending/receiving text messages or tones. *M2* is employed for communication and it is attached to a specific robot in the JMR environment. *M3* receives and re-sends text messages and establishes the connection between modules in the telephone network.

4.1 Communication by tones

First, the connection is stablished dialing, from *M1*, the number attached to *M2*. *M3* determines the connection. Once the connection is established, we can send commands (sequences of tones) from *M1* to *SimRobot* through *M2*, which is provided with a command verifier (*VeriCommand*) to validate the orders. If the command is correct, it is re-sent to *SimRobot* where the robot executes the action.

The commands are defined as a sequence of tones:

**	Start/end command.
#***n*	Robot moves forward *n* centimeters.
#*#*n*	Robot turns *n* degrees clockwise.

For example, to move the robot 20 cm forward, we send the sequence: ** #**20 **

The robot status is always checked previous to execute any command. The possible states are:

Collision: The robot has collided.
Moving: The robot is moving.
Stopped: The robot and motors are on, but it is stopped.
No motors: The robot is on, but motors are off.
Disconnected: The robot is off.

If the status is *No motors* or *Disconnected*, then the commands cannot be executed. If the status is *Collision*, then it is only possible to execute turns. If the status is *Stopped* then any command can be executed.

M2 can send tones to *M1* depending the status of the system, the tones are:

1 The connection is accepted.
2 The connection is denied.
3 The command has been executed successfully.
4 The robot has collided.
5 The robot is in an infinite loop.
6 The robot is executing a command.

4.2 Communication by SMS

The commands are written as a text message, following the rules dictated for tone communication. We employ the messages screen in *M1*. Once the message is written, we send it providing the phone number attached to *M2*. The message is processed in the SMS center and re-sent to *M2*. When *M2* receives the message, it is verified. If the command in not valid, then *M2* constructs an error message, which is sent to *M1*. If the command is validated, then it is passed to *SimRobot* and the robot executes the order.

5 MODELING THE SYSTEM

We employ UML as a tool to develop the system. UML determines a set of notations and diagrams to model object-oriented systems and describe the essential semantic respect the meaning of symbols and diagrams (Popkin, 1998). UML is used to model many kind of software and hardware systems, and the real world organization.

We consider four stages to model *SimGSM*: (1) Analysis, (2) Design, (3) Implementation and (4) Testing.

The stages include a set of UML diagrams. Analysis includes the diagrams of (1) Situation and (2) Classes. Design comprises (1) Classes (detail), (2) Sequence, (3) Interaction, (4) Status and (5) Activities. Implementation has only the Members diagram. Finally, the black box method was employed for Testing.

6 THE OPERATION

SimGSM has two interfaces (modules): The cell phone (Fig. 3) and the SMS/Phone center (Fig. 4). The cell phone (*M1*) emulates the basic functions of a real GSM cell phone and it is employed to send commands and receive the status of the robot in *SimRobot*. The SMS/Phone Center (*M3*) is in charge of verify, validate, monitor and establish communications and data transmission between modules.

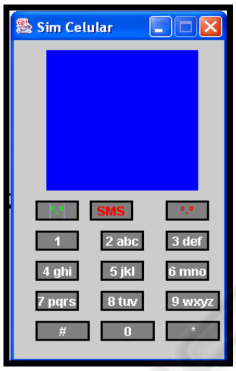

Figure 3: The cell phone interface

Figure 4: The SMS/Phone Center interface

The *SimRobot* interface is a modified version of JMR, where we add the GSM modem (*M2*), which is attached to the robot, and the *VeriCommand* module (Fig. 5). The additional modules are employed to monitor and verify commands, communication and data transmission.

The robot control simulation was performed employing the interfaces. The robot and the world are initially loaded in *SimRobot*. Next, we send several commands following the rules dictated in section 4.1 and we receive the corresponding (text or tone) message for every command sent. In *SimRobot* we could observe the robot behavior, the execution of commands and the robot status. During test we observe an acceptable performance. We test the two possibilities of control, considering all reasonable possibilities because we expect to translate this system to real world, using commercial cell phones, the GSM network and a Pioneer mobile robot.

7 CONCLUSIONS AND FUTURE WORK

We have present a telerobotics system simulator as a first approximation to control mobile robots using a cell phone through the GSM network: Telerobotics by cell phone.

The system was modeled employing UML to consider all possibilities that can emerge during operation. UML modeling includes analysis, design, implementation and testing models, and every model has different UML diagrams.

The system was developed in Java, including JMR and JMF. It is a client-server architecture. We expect some modifications and additions in future to the system: (1)A module for speech recognition to command the robot via voice, (2) a module for speech creation to provide oral status information to user, and (3) the robot control via Internet considering that at present, cellular telephony technology provides such service.

We also expect to evolve the system to real world, using cell phones, the GSM network and a mobile robot as Pioneer as soon as possible.

Finally, due latency, the control of the robot using SMS messages is not recommended, but it is just an alternative and possibility.

REFERENCES

ActivMedia Robotics. 2005. http://www.activrobots.com.

Fogelklou, C. 1999. DTMF Generation Using the SX Microcontroller. *Scenix Application Note.* No. 6.

Gallardo-Lopez, D. et al. 2003. *Java Mobile Robots (JMR) Manual del Programador*. DCCIA, U. de Alicante.

Gerkey, B. P. et al. 2003. The Player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems. In *Proc. of the International Conf. on Advanced Robotics 2003*. Coimbra, Portugal, pp. 317-323.

Ibrahim, J. 2002. 4G Features. *Bechtel Telcom. Technical Journal*. Vol. 1, No. 1, pp. 11-14.

Konolige, K. et al. 1997. The Saphira Architecture: A Design for Autonomy. *Journal of Experimental & Theoretical AI.* Vol. 9, No. 1, pp. 215-235.

Popkin, Software and Systems. 1998. *Modeling Systems with UML*. A Popkin System White Paper.

Rahnema, M. 1993. Overview of the GSM System and Protocol Architectura. *IEEE Communications Magazine*. No. April 1993.

Yucatán, D. de. 2001. Teléfonos celulares ¿bendición o pesadilla. *Reportaje Especial del Diario de Yucatán*. http://www.yucatan.com.mx/especiales/celular/

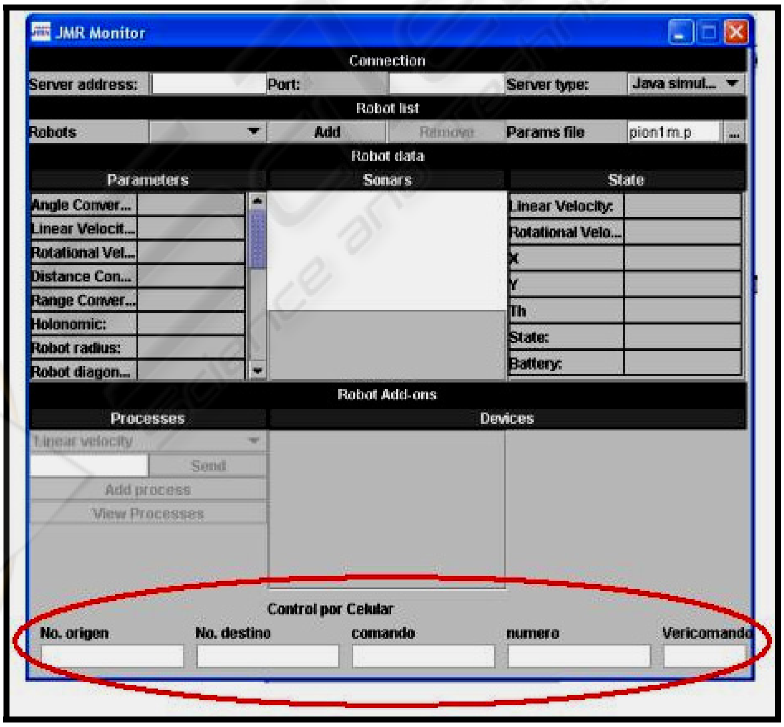

Figure 5: *SimRobot*: A modified version of JMR. The additional modules corresponding to *GSM modem* and *VeriCommand* are indicated at the bottom of image