The Backup Restore Challenge

Tomas Lampo, *Student,* Javier Sierra, *Student,* Marynel Vázquez, *Student*

***Abstract*— This is sp15.gia.usb.ve's Team Description Paper for its participation in the SEK category of the 7^{th} IEEE Latin American Robotics Competition[2]. This contest's challenge consists on constructing and programming two autonomous robots that must cooperate efficiently to restore a spaceship main computer, in order to allow its crew to return to Earth safely. This paper describes the hardware and software decisions made by the team and the drawbacks encountered while designing the robots.**

I. INTRODUCTION

Every year since 2002, the Institute of Electrical and Electronics Engineers, IEEE [1], has held a Robotics Contest, known as the IEEE Latin American Robotics Competition, LARC, where students and researchers measure their knowledge by participating on the different categories this contest has to offer.

For this year's contest, three Computer Science students, under the guidance of robotics expert Carolina Chang, formed the sp15.gia.usb.ve team and chose to participate in the SEK category, using LEGO Kits to construct their robots.

In order to fulfill the challenge, two robots must be designed using only LEGO blocks and pieces, as well as LEGO sensors and Mindstorms Bricks. The rules[3] specify that these robots should work on a closed Arena, divided into two different areas: a "password room" and a "backup room".

One of the robots must go to the backup room to find the last backup tape, which is represented by a black wood cube stored among several blue wood cubes, representing old backup tapes. Tapes are positioned into cabinets, one above the other. This robot must obtain the last backup tape without dropping any of the old backup tapes.

The other robot must go to the password room to find the password card, needed to restore the main computer. This password positioned in the top of the steps in the password room. This robot must climb the stairs and obtain the password card.

Once the robots have obtained the last backup tape and the password card, they must place themselves in front if the main computer's input and introduce the last backup tape and the password card at the same time, restoring the main computer.

Once the challenge was carefully studied and understood, the team members had several ideas to solve the problem and proceeded to discuss them. These ideas led to the construction and programming of two autonomous robots, *Copa* and *Cabana*, that solve the problem by working together on gathering the wood cubes and introducing them in the main computer in an efficient way.

II. STRATEGY

The robots must move as fast as they can and communicate along the way to achieve their goals.

Copa, the robot on the backup room, notifies *Cabana*, the robot on the password room, that the challenge has begun. Then *Copa* visits each cabinet in order to look for the last backup tape. Once it has found it, the robot grabs the tape and returns to the main computer input, and then notifies *Cabana* that it is ready to introduce the tape into the main computer.

Cabana, while *Copa* is doing its task, proceeds to climb the stairs and sweep the top step, collecting the password card. Once it is done sweeping the step, it goes downstairs and returns to the main computer input, and then waits until *Copa* notifies it's time to introduce the password card.

Once *Copa* has notified *Cabana* that it's time to introduce the password card and the backup tape into the main computer, *Cabana* confirms it is ready, and then they introduce the password card and the backup tape at the same time.

III. HARDWARE DESIGN

A restriction highly taken into consideration was the robot's size. The rules specify that it must fit into a 25 centimeter cube and each must have, at most, 6 sensors and 6 actuators. So each model was carefully measured before being approved, and many other robot models where rejected by this criteria.

The team also discovered that the following aspects where necessary for successfully completing the task:

- Each robot's structure should allow it to easily move forward and backwards.
- Each robot must be able to grab the wood cube and introduce it into the main computer.
- Robots need a robust chassis, that would allow each robot to accomplish its task confortably.
- *Cabana* needs to be able to climb stairs and grab the password card, while *Copa* needs to be able to grab the last backup tape without dropping the other tapes, which led to different anatomies for the robots.
- Navigation is made easier if robots are *holonomic*.

Copa was designed and constructed in a way that allows it to elevate itself, so it can analyze the tapes with a light sensor and grasp the correct one. On the other hand, *Cabana* was designed and constructed in a way that allows it to climb the stairs and pick up the cube with a spinning mill inside it. This mill ejects the cube into a tray that is then used to deliver the cube into the main computer.

The team concluded that one NXT[4] brick was enough for *Cabana* to accomplish its task, while two bricks were needed for *Copa*. *Copa* would use one brick for navigation and the other one for cube recognition and retrieval.

Taking into account that the kinematic design of a robot highly influences the accuracy of odometry measurements for dead reckoning[5], a differential drive was chosen as the mobility configuration for *Copa* and *Cabana*. This decision was made because of the simple geometric equations that can be used to compute the position and odometry of the robots.

Having two of the three output ports of the NXT Brick taken to control the wheels of the robots, on the third one analog motors were connected in series. The main difference between analog motors and digital servo motors is that the latter cannot be connected to other motors, and are larger than the analog ones, since they include an encoder sensor.

These motors have been carefully synchronized by the team, allowing the robots to achieve their delicate task of picking up the blocks in an almost fail-proof way.

IV. Software Design

The robots were programmed using the tool "Not eXactly C (NXC)"[6]. Despite its lack of accurate documentation, this tool was chosen for its similarity to the C language, which is known for its speed and simplicity.

Navigation has proved to be specially difficult for *Cabana* in the given Arena, since it must climb the stairs in order to pick up the cube. Therefore, the team found itself in the necessity of designing an accurate odometry and sensor set that would allow the robot to move through the Arena, without losing its way.

The team also found out that one cannot completely rely on the sensors, for they have proven to be inaccurate. The Compass reacts differently depending on its proximity to the NXT brick or the motors. The Ultrasonic Sensor tends to give "ghost" readings, and sometimes fails to read a wall in front of it, if it is facing its thin side. Having experienced these drawbacks, the team decided to rely on their odometry and correct flaws with sensors.

V. Task Accomplishment

Several drawbacks were encountered while trying to accomplish the task with the robots. These drawbacks are:

A. Navigating the Arena

As was described before, the robots need to rely on odometry to navigate the Arena. So robots move forward and backwards using the odometry, and then check with the Compass Sensor if they have turned enough (or too much), and proceed to correct their movement, if necessary. By doing so, the team guarantees that the robots always move in straight lines, and turn in right angles. This makes the Robots navigate the Arena in an "Euclidian" way.

B. Climbing the Stairs

Cabana uses tracks to climb the stairs. These tracks were set up in an angle that allows the robot to climb the steps without needing additional leverage. However, climbing each step may lead the robot to make harsh motions, which may lead to the robot losing its heading. Therefore, the robot must align itself each time it climbs a step.

Additionally, once the robot is on the top of the stairs, it must sweep the top area, to make sure it picks up the cube with its mill.

C. Cube Grasping

Copa needed to be able to elevate itself to the level of the cube that needed to be grasped. To accomplish this, an elevating system was deviced. Once the robot has recognized the last backup tape, it moves to align the grasping system with the cabinet and then proceeds to rise itself to the correct height. Afterwards, the robot extends an "arm" to reach the cube and proceeds to retrieve it by rotating wheels.

D. Communication

The robots need to keep in touch while they are accomplishing their task.

From the start one of the robots must let the other one know the task has begun. This same method is used by the robots to deliver their cubes at the same time, once they are in front of the main computer input.

As has been mentioned before, *Copa* uses two NXT bricks to accomplish its task. These bricks need to comunicate with each other, so the robot knows when it is safe to move between cabinets and when it's time to recognize and grasp cubes.

Acknowledgment

The authors would like to thank everyone at the Artificial Intelligence Laboratory, GIA[7], for their support and advises.

They would also like to thank Movistar for funding the project.

References

[1] Institute of Electrical and Electronics Engineers, IEEE http://www.ieee.org/

[2] 6^{th} IEEE Latin American Robotics Contest for Students (LARC 2007) http://roboticforum.org.mx/contest.html

[3] Rules for the Latin American Robotics Contest, SEK Category 2008 ftp://jri2008:web@users.dca.ufrn.br/regras/SEK2008_en.pdf

[4] NXT® LEGO® Mindstorms® Kit http://mindstorms.lego.com/Overview/

[5] J. Borenstein, H.R.Everett, L.Feng, "Where am I?" Sensors and Methods for Mobile Robot Positioning, The University of Michigan, 1996. http://www-personal.umich.edu/ johannb/position.htm

[6] Not eXactly C Language http://bricxcc.sourceforge.net/nbc/

[7] Artificial Intelligence Laboratory, GIA Universidad Simón Bolívar Caracas, Venezuela http://www.gia.usb.ve/