RoboCupRescue 2006 - Robot League Team Resquake, Iran

Ehsan Aboosaeedan[1], Ali Jazayeri[2] ,Arash Kalantari[3], Ehsan Mihankhah[4] , Hesam Semsarilar[5]

[1] K.N.Toosi University Of Technology
Seyed Khandan Bridge, Shariati St.,
Tehran , Iran +98 21 8600075
ehsanaboo@hotmail.com

[2] K.N.Toosi University Of Technology
Seyed Khandan Bridge, Shariati St.,
Tehran , Iran +98 21 8600075
sajazayerim@yahoo.com

[3] K.N.Toosi University Of Technology
Daneshgah Bulv., Forth Square Of TehranPars,
Tehran , Iran +98 21 8600075
Arash1362@yahoo.com

[4] K.N.Toosi University Of Technology
Seyed Khandan Bridge, Shariati St.,
Tehran , Iran +98 21 8600075
ehsanmihankhah@yahoo.com

[5] K.N.Toosi University Of Technology
Daneshgah Bulv., Forth Square Of TehranPars,
Tehran , Iran +98 21 8600075
hesam2k@yahoo.com

Abstract. Through this paper we introduce Resquake's recent work on implementing an Autonomous Mobile Robot. The team-work on Rescue robot has ended up to different mechanical and electrical designs and also the user interface is progressively becoming more friendly and efficient for the operator while new features are being added to the system. Experiment results have made a better sense of operator's needs while driving the robot and thus the whole system has improved to create a better feeling of standing at the robot's position for the operator while collecting as much information as possible from the environment. In this paper, we discuss our latest locomotion system which is maneuverable on uneven terrains. Electrical design of motor drivers and microcontroller system and power circuits will also be described. User interface is also explained in detail. Our new approach to solve wireless problem will be mentioned as well. The project is actually consisted of two different parts: preparing a good Tele-Operated system and designing an autonomous system. As we started with RoboCup Rescue league in RoboCup 2004-Lisbon and continued in RoboCup 2005-Osaka we currently have reliable basis like Avril (Our Autonomous System), with the neural network solution for path planning and decision making, which is becoming ready for 2006 event and Silver (Our Tele-Operated system), with outstanding stair climbing ability and the innovative design that compromises the contradictory aspects of a small system with higher maneuverability and a high performance large system for surmounting undefined obstacles, which was introduced in Osaka and achieved the 2nd Place Best Design Award and is becoming even better for Bremen.

1. Team Members and Their Contributions

- Ehsan Aboosaeedan — Power Supply and Electrical Design, New Wireless System Designer
- Ali Jazayeri — Microcontroller And Embedded System, Electrical Design And Neural Network Control
- Arash Kalantari — Mechanical Design And Simulation
- Ehsan Mihankhah — PC-Based Control and Software Developer,\ Operator
- Dr. S. Ali A. Moosavian — Advisor
- Dr. HemidReza Taghirad — Advisor
- Hesam Semsarilar — Mechanical Design And Implementation

2. Operator Station Set-up and Break-Down (10 minutes)

Everything is packed in a toolbox (the Operator's Laptop and printer and other accessories such as joystick and mouse and a bunch of papers and writing tools and …), and each robot has its own package. So we need five people, Operator will not do anything, so we have the rest 4 members of the team in setup and Break-Down Time. Each person will setup the parts in the box he is carrying. We estimate a 3 minutes setup and a 5 minute Break-Down process. Hesam will carry the toolbox. He will plug the Laptop accessories (joystick, mouse, printer …) Ali will put the papers and writing tools in the place near the operator where was found handy during the practices before the competition. Ehsan (Aboosaeedan) and Arash will setup the robots.

Ehsan (Mihankhah) is the operator who will do nothing but watching the others and verifying the setup.
Break-Down procedure is similar to setup procedure.

3. Communications

All Wireless communication is implemented over 5 GHz via IEEE802.11a wireless LAN cards and access points. No other frequency range is occupied. We are using Gigabyte PCMCIA LAN cards.
One of our new approaches is working on new video transmission hardware to replace Wireless LANs. This hardware will be working in 5 GHz as well. We will talk about it in more details later.
Right at the moment, video, audio and data transmission is done by sending packets over wireless network in C#.NET. The system worked well in Osaka (We did not have ANY problem with wireless communication in Osaka, Neither on Video transmission nor on data transmission). We can set the wireless system to work only on one channel, but we can change to any 802.11a channel (almost all specified channels to work in 54Mbps and 108Mbps) in case we are asked to. The signal strength is 48dB or less.
Here we will discuss the software code for video, audio and data transmission. Please note that all the programming is done in .NET environment and the programming language is C#.NET. The source code can be downloaded from our website (www.resquake.com).

Data Transmission:

Data transmission is done using windows sockets. Using System.Net.Sockets namespace in C#.NET we have all the classes and events we need to transmit data over the LAN. As the program should do many things other than listening to sockets and responding, we need to add threads listening to sockets while the CPU is free for doing other tasks.
So, we have one delegate that is passed as the thread function to keep watching windows sockets for beginning the communication. When the connection got established other threads should start working. JoystickThread and positionThread are two other threads that transmit data over the LAN. We of coarse need a delegate for the thread that receives data from the LASER scanner on the robot (This data will be used for map generation and localization in PositionThread).
Almost the same code is written in the robot side program for communication with the program on operator station PC or laptop.

Video Transmission

Video transmission is the most critical part of communication as we need large bandwidth and a reliable connection (Unfortunately in RoboCup we do not have it) to send video streams. The problem with video streams is that we have to buffer them before showing and this means delay. Another problem is that we need an uncut stream. What if the connection is lost for some reason? All these have leaded us to the point that we have to send frames instead of streams. If we can send more than 12 frames in a second, then we have a live (almost no delay) video and we should not worry about rebuilding the lost stream. We may lose some frames that results a less frame rate but we still have the video. So, what we are doing at the moment is sending frames through LAN with VideoCapX ActiveX control (for more information see www.fathsoft.com) The key properties and method names for sending and receiving frames over LAN are: ServerMode, SendFrame and ReceiveFrame. The C# source code could be found at the fathsoft website. The experience we had in Osaka was great. We had two live videos from two different points of view.

Audio Transmission

Do we have the same problem with audio transmission? The answer is NO. What we really need from the audio is to see whether there is a sound or not. So the delay is not much critical here. We have better have a high quality sound and just accept the delay of about a second or two. We have used Windows Media Encoder SDK for sending audio streams over the LAN. (For more details see MSDN in www.microsoft.com).

Nothing more seems to be left about the wireless communication system but the new hardware project that we are currently working on it and we will talk about it later. If there is the need of any further information feel free to contact us (ehsanmihankh@yahoo.com).

New Approach for Using Hardware for Communication

Weight, size and power supply have always been problematic issues for a mobile system. If we can reduce the stuff we are carrying with the robot, we are one step ahead.

We are working on a transceiver hardware that works in 5 GHz (actually the same components that are used in wireless LAN cards) to remove the laptop from the robot. Instead we do the process in microcontrollers and DSPs (ATMEGA128, AT91SAM7S256 and ARM 40800 microcontrollers) and the rest can be done in operators's PC or a separate PC or LAPTOP which is in wireless communication with the robot.

This way we are cutting down on power (by removing interface components) and also we can reduce the size and weight (by removing the Laptop and it's accessories from the robot)

4. Control Method and Human-Robot Interface

Resquake User Interface which is a simple user friendly environment for the operator to control the robot (motors and actuators) and send commands to it and of course gather the data coming from sensors and preview video. Some parts of user interface were discussed in communication section where we described wireless Data, Video and Audio transmission. Here we talk about the rest of the code. First thing is driving the joystick. We can do it by adding Directx namespaces to the program and using the proper functions. So the first step is adding Microsoft.DirectX and Micro-

soft.DirectX.DirectInput namespaces. Then the Joystick events should be captured in a separate thread and finally the changes should be sent to the robot through windows sockets.
Our previous work on graphical user interface could be found in 2005 and 2004 TDPs. The graphical user interface will be changed due to new sensors and actuators that are added to the system.
Some items such as printed report, video size and map mode will be included in 2006 program as well, but we are going to improve the graphical interface (A migration from VS.NET 2003 to VS.NET 2005 will happen for sure).

5. Map generation/printing

Localization is done using optical encoders and map generation is done taking advantage of the location and the data coming from a LASER scanner. Data from optical encoders are processed in the microcontroller and the position is sent on request to the Laptop on the robot. LASER scanner generates an array of data containing the distance of the objects around the robot. This data will be serially sent to the Laptop on the robot. The map is generated using these two sets of data. Here some information about the C# code for generating the map is provided. We have tried to remove encoders from localization process because of the inevitable additive error of the mechanism and implement SLAM with our LASER scanner.
The Robot should gain information from the environment through the serial port and process it in special threads.
Data is read from the serial port in the Laptop on the robot and is sent to operator station. Here we should process and make the map. There is the delegate that is passed as the function of the PositionThread that uses this set of data and generates the map.A timer is refreshing the preview of the generated map for the operator each 2 seconds.
And finally the map is printed using System.Drawing.Printing namespace and Print-Document object of C#.NET.

6. Sensors for Navigation and Localization

Here is the picture of the rotary encoders (Fig.1):

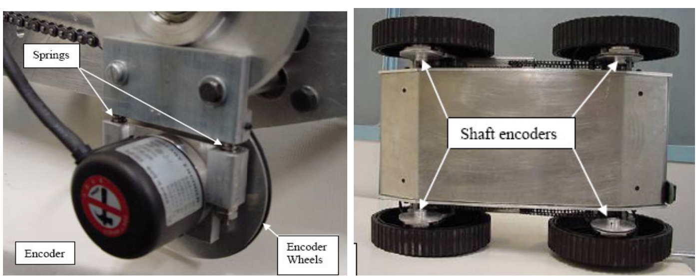

Fig.1 – Rotary Encoders

Four rotary encoder manufactured by Autonics are used in Avril The exact part number is E40S-1000-3-1 and a complete description is available at:
http://ecp-co.com/pdf/rotary_encoder/e40s.pdf

Here is the picture of the LASER range finder which can find distances between the robot and nearest objects within 270° of angle. Figure.2 also shows what a laser range finder produces.

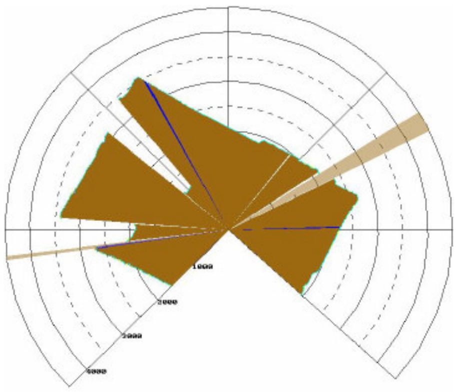

Fig.2 – URG-X002 Laser Range Finder manufactured by HOKUYO LTD

When we are using encoders for localization, LASER scanner helps us generating the map but when we do not use optical encoders (mainly because of the additive errors produced by the them, that is caused by the changes in surfaces in real environment) LASER scanner is used both for localization (by comparing the changes of two consequent sets of data) and map generation (adding new pixels to the map).

7. Sensors for Victim Identification

7.1 Thermal Sensor

Our thermal sensor is actually a pyrometer (Fig.3).

Fig.3 – Raytek Pyrometer

Sensor sends data serially to the laptop on the robot. This data is captured using a setarate thread and is sent to the operator using windows sockets. This data is available on screen for the operator and is declared in the victim sheet as well.

7.2 –Motion Detection:

The video sent to the operator will be analyzed using VideoCapX control and if any motion would be detected the operator will be prompted.

7.3 – Sound Detection:

As the operator has live video and audio (it has been thoroughly discussed in communication part), the sound can be a means of victim identification as well.

8. Robot Locomotion

Up to now four different locomotion systems have been designed and manufactured. The mechanical structures of the former three (Fig.4) have been fully described in 2004 and 2005 Resquke TDP. The latest product, Silver (Fig.5), was designed and fabricated within the following procedure:
First, the general task to be performed by such a robot was defined, and variant kinematical mechanisms to form the basic structure of the robot were evaluated. Choosing an appropriate mechanism, geometric dimensions and mass properties were detailed to develop a dynamic model for the system. Next, the strength of each component was analyzed to finalize its shape. To complete the design procedure, Patran/Nastran was used to apply the finite element method for strength analysis of complicated parts. Also, ADAMS was used to model the mechanisms, where 3D sketch of each component of the robot was generated by means of Solidworks, and several sets of equations governing the dimensions of system were solved using Matlab. Finally, the components were fabricated and assembled together with controlling hardware. A brief description of the so-called procedures is illustrated in this section.

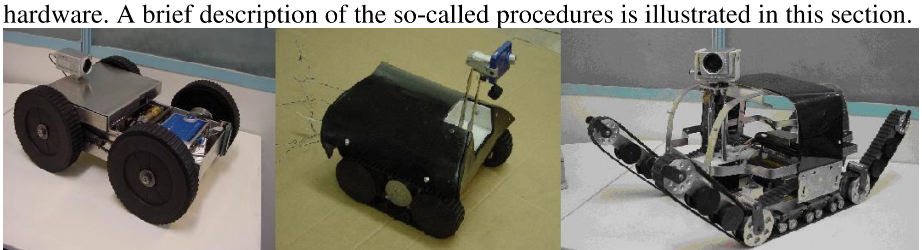

Fig.4 – Avril / Sangposht / Snail

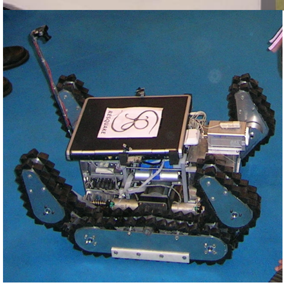

Fig.5 – Silver

8.1. Mechanism selection

The main part of the locomotion system is consisted of a pair of tracks which are used because of their ability to move on uneven terrains and their inevitable traction. As well as the type of locomotion, the size of a mobile rescue robot is also an important issue. In a destructed indoor field there may exist some obstacles that can not be passed by any system such as when the walls or the ceiling collapses. At this situation, the robot should search for a bypass or a way between the obstacles rather than climbing over them, which definitely depends on its size. A relatively small robot can easily pass a narrow aisle and continue its search. It should be noted that stair way is an inseparable part of an indoor environment. A rescue robot should have the ability to climb and move down stairways, whether destructed or not, to cover the whole area. In order to compromise between the two contradictory aspects of providing a small robot with maneuverability of a larger locomotion system, an exceptional mechanism has been developed. This mechanism, which includes a base with two expandable tracks (arms), enables the robot to resize depending on the situation that encounters.

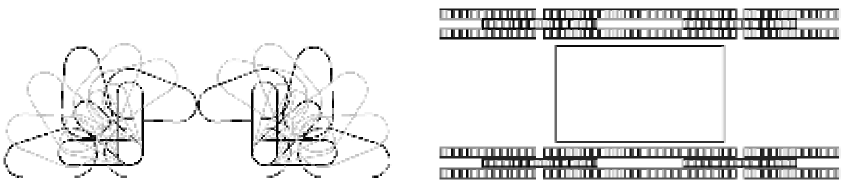

Fig.6 - Final mechanism chosen for the tracks

A planetary gear chain is substituted to simply transmit the power of the main joint of each arm to its second joint. In this case, the rotation of the two parts for each arm will not be independent. Thus, two desirable positions of the arms are considered, and the gear ratio is obtained such that the arms move based on a path between these two positions (Fig.6). It should be noted that while the main part of the arm rotates $\pi/2$ rad, the second part should rotate more than π rad. The gear chain with such performance should be a planetary gearbox. The main part of the first arm plays the role of the arm in the planetary chain, which is directly powered by a motor. The sun gear should be attached to the body of the robot and the planet gear should be a attached to the second part of the arm. A pair of medium gears is placed between the sun and the planet so that the diameter of gears does not exceed a reasonable length (the diameter of the main wheels of the tracks) (Fig.7) (Fig.8).

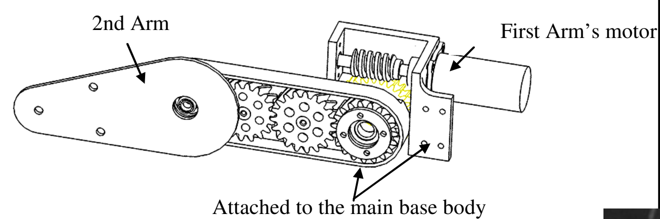

Fig.7 - Final designed arrangement for the arms

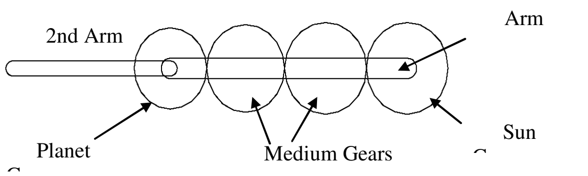

Fig.8 - Planetary gear-chain

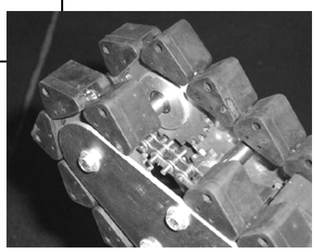

Fig.9 - Latex pieces
Fixed on the chain

8.2. Tracks

The traction of the locomotion system strongly depends on the friction between the track peaces and the surface on which the robot moves. That is why the material and the shape of the track pieces are of great importance. On the other hand, the tracks should also bear a reasonable tension. Thus, the tracks are designed to be assembled using two components. A basis of chain-sprocket provides the system with sufficient tensile strength and tooth shaped pieces made of latex filled the gap between the chain and surface to create required friction. Fig.9 shows how latex pieces are attached to the chains.

8.3. Suspension system

Two major advantages are obtained by including a suspension mechanism.

- Increasing the flexibility and stability of the system on bumpy surfaces;
- Damping any shock to the system caused by collisions.

The suspension system was designed by separating the two sides from the main body and then attaching them by a revolute joint (Fig.10). A pair of linear springs limits the angle of rotation and makes the system remain at a single position when no extra forces are applied. It should be mentioned that using dampers was not needed, because the friction of the plain bearings used as the so-called joints was enough to limit any extra shaking of the springs.

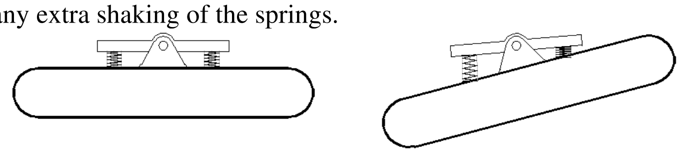

Fig.10
Basic structure of the suspension system

Finishing the design of locomotion mechanisms, now it is time to determine the dimensions. Some of the components for building the mechanisms are available as standard parts, so we should select other dimensions to match their counterparts. Besides, the overall size of the robot and the formulas on the gear chains must be considered in our calculations. Since there are numerous equations governing these factors, an optimized solution is not reachable by manual calculations. Thus, Matlab is used to extract desired values from a set of equations.

8.4. Weight Optimization and Stress analysis

The aim of such analysis is a minimal weight design, to reduce on-board energy consumption. The results lead us to redesign critical parts such that undesired deformation and stresses are prevented. Using classic methods, the stresses in all gears and bearings can be calculated and properly designed. However, this is not applicable to some parts, e.g. the motor base structure or the gear-set basis, where equations became really complex and unsolvable. In such cases, the finite element method using Patran/Nastran is applied(11).

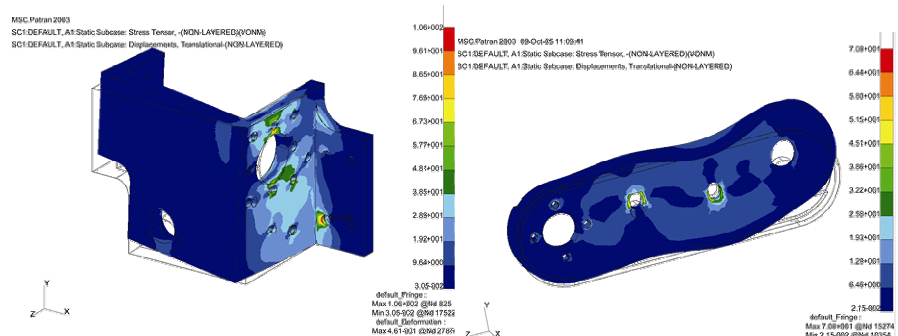

Fig.11. Exaggerated deflection of complicated parts under load

Following a complete stress analysis of all different parts, and redesigning of those required, the last step is to fabricate the designed parts and assemble those together with the available standard elements to achieve the robot as shown in Fig. 12. Finally silver works as a tele-operative mobile robot with the following capabilities:

1. Surmounting uneven terrains with a relatively high stability;
2. Climbing stairs and slopes up to 35^o;
3. High reliability of mechanical structure and hardware.

Other specifications of the robot are listed in Table 1.

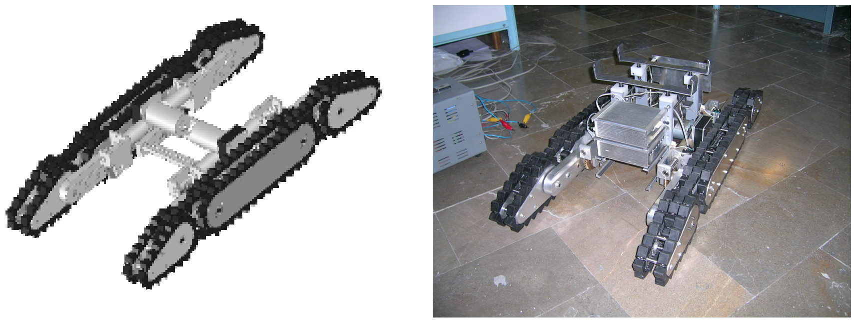

Fig 12.Complete locomotion structure

Overall weight	25 Kg
Length with expanded arms	80 cm
Length with closed arms	41 cm
Minimum height	26 cm
Width	40 cm
Maximum velocity	19 cm/sec
Arms maximum angular velocity	4 rpm
Number of ball bearings	82
Number of plain bearings	32
Total number of parts	688

Table 1. Main specifications of Silver

9. Other Mechanisms

Speed Control:

By controlling the current applied to the motors, we can retain the same speed when the robot is surmounting different obstacles. This actually happens because we are changing the torque by changing the applied current. This Control will be done using an Atmel AVR microcontroller (like ATMEGA16) and the feedback from rotary encoders.

10. Team Training for Operation (Human Factors)

Resquake Operator Interface has the least possible objects in the easiest and most accessible places and may need something about 10 minutes to describe what an operator should do to control the system. It is also very easy to setup Hardware parts, and we do our best to finalize the system in the way that they only would need to be turned on and would not need any time taking setup process.

We have tested the robot several times on various obstacles. The experience of participating in two previous RoboCup events helps us a great deal to see where we have to practice more and what kind of unpredictable problems we may face during a mission. It is a fact that the more the operator practices in real arena, the better performance he will give.

11. Possibility for Practical Application to Real Disaster Site

Resquake has done his best to make robust system practical for real disasters. There are some new ideas for working in such conditions, but of course the system is not completed yet and we do not claim that it is now ready to perform in a real disaster. The automated system has a pretty long way to reach the goal of an autonomous mobile robot that performs in real (RED) mission. The tele-opereted system is becoming close to the goal, but of course the communication system for the real mission should be something more reliable than 802.11 based wireless LANs. The arena is not always the ideal dry RoboCup arena in normal temperature in a roofed building and these are all the challenges that should be taken to account for the real disaster.

12. System Cost

These are the approximate costs of the latest robot (all in US Dollars)

Part Name		Qty.	Price
Mechanical	Motors	6	1000
	Gear box	6	120
	Structure Mechanical Part	1	1500
	Other Mechanical costs		100
Electrical	Localization System		2000
	Other Electronic boards		200
	Electronic Part Interfaces		2000
	All other Sensors		2500
Computer	Wireless LAN and Access Point	2	500
	Capture Card	2	400
	Laptop	1	1500
Others			300
Sum			12120

References

1. Shigley, Joseph Edward: Mechanical Engineering Design, 6th edn. , McGraw-Hill
2. Martin George Henry: Kinematics and Dynamics of Machines
3. McComb, Gordon: The Robot Builders Bonanza, McGraw-Hill, 2000
4. Oberg, Erik: Machinery Handbook, 26th edn. , Industrial Press, 2000
5. MSDN Library, Microsoft Corporations

Video

http://www.resquake.com/Qulification_Clip.WMV