Robosapien Robot used to Model Humanoid Interaction to Perform tasks in Dangerous Manufacturing Environments

R. Stopforth¹ and G. Bright²

¹,²Mechatronics and Robotics Research Group (MR2G), University of KwaZulu-Natal, Durban, South Africa

E-mail: ¹ stopforth@ukzn.ac.za and ² brightg@ukzn.ac.za

Abstract. Humans are involved with accidents in manufacturing environments. A possibility to prevent humans from these scenarios is, to introduce humanoid robots within these industrial areas. This paper investigates the control scenario and environments required at a small scale level, with the use of the Robosapien robot. The Robosapien robot is modified to control it with a task of removing a cylinder and inserting it into a hole. Analysis is performed on the performance of the Robosapien robot and relating it with that of a humanoid robot. A discussion with suggestions is concluded with the efficiency and profitability that would need to be considered, for having a humanoid robot within the manufacturing environment.

1. Introduction
Studies in advanced manufacturing environments have shown that 20 % (of 103 reported cases) were maintenance and repair personnel that have been injured in accidents [1]. Dangerous environments could be considered as areas with hazardous chemicals, gases, voltage and currents, and machinery that have moving parts. Service, maintenance and repair robots have been limited with the tasks that they perform and were designed to perform set specific tasks. The maintenance robots are many times only located and used for a specific section of a manufacturing environment.

Humanoid robots such as the Honda Asimo [2] robot have been developed to walk in a stabilized manner over different obstacles. General Motors have developed for NASA the Robonaut [3], which was designed to perform operations and maintenance that were too dangerous for astronauts.

The question that arises is whether humanoid robots will be efficient in the manufacturing environment, considering the complexity of their design and development, therefore having a high purchase and maintenance price themselves. As to whether it would be more beneficial for a company to have one humanoid robot to perform multiple tasks, or a number of cheaper service robots able to perform each only one task, will depend on a company’s needs and requirements.

Before investment in humanoid robots for manufacturing companies are pursued, it is required to evaluate the performance at a small scale and to determine the technologies that will be incorporated into the manufacturing environment so that the humanoid robot will be able to perform desired tasks.

This paper views some of the requirements that will need to be considered by investigating the performance of a small-scale humanoid robot. The Robosapien robot is used with a personal digital
assistance device in order to control it remotely from a host computer. The task that was assigned to the robot was to go to a destination, pick up an object and drop it off at another destination. This was a simple task that was used to determine if the humanoid was capable of performing a user defined function of replacing a part. As parts within a manufacturing environment have become modular for quick repairs and a decreased downtime of the plant, or as for recently research reconfigurable manufacturing systems (RMS) [4], the task assigned to the Robosapien robot is ideal, as it represents the removal of a faulty module.

2. Communication and Software Interaction
The HP RX3715 Personal Digital Assistant (PDA) or pocket PC was used to relay commands from a computer to the Robosapien robot. The purpose for the use of the PDA as a relay station was that it allowed for Wireless network communication for the PC to interact with the designed software and having the ability to communicate with the Robosapien robot via infrared transmission. Further descriptions of the configuration is described later on in this section.

There were two main types of pocket PC software used for the Robosapien interaction. They were Nevo and dotPocket. There is a lot of remote control software available that enables one to control a device that has infrared capabilities. The remote control software that has been chosen to be used in this research and analysis is Nevo. There are others similar to Nevo such as Ultramote and VitoRemote. They all have the basic capabilities of controlling infrared devices.

Nevo offers complete audio visual, home and digital media control and is only available for users of mobile devices. It allows the user to seamlessly connect, control and interact with digital media and electronic devices in a network. Some interesting features that are available with Nevo are:

- The ability to control automation devices.
- It has an intuitive touch screen visual interface.
- It allows the user to customise the graphical interface.

The Nevo software was programmed to map with all of the existing functions of the Robosapien robot and additional tasks that enable the robot to perform a specific routine. This function is very useful as it allows the user to control the Robosapien and multiple devices as well.

Programs can also be written and executed on the PDA to control the robot. The software used needs to be specifically for embedded devices because the operating system of the PC is different from that of the PDA and some programs that are written for a PC do not work on the PDA.

The dotPocket software allows the user to control the pocket PC from a host computer. The screen of the pocket PC appears on the monitor of the host computer (as seen in figure 1) and the user is able to have full control of the pocket PC. A wireless connection is requires for a link to be formed between the PDA and the host computer. This tool is very useful as it makes it possible to control the pocket PC while it is in a different location within a set proximity.
A pocket PC can also be used to control the robot using Infrared. In this instance, the PDA acts as a remote but it can also be programmed to do a specific task, not limiting it to the pre-programmed functions. The PDA that is used supports Infrared and this feature was used to communicate with the Robosapien. The PDA supports slow Infrared which has a maximum transmission rate of 1.15 Mbps. With tests performed it indicated that the infrared could be used for short range transmission of 0.2 m.

The use of infrared requires low power therefore it was beneficial for use in the PDA. The use of the PDA was required it to be on the robot (Figure 2). This means that the PDA will be as mobile as the robot and will continually be in line of sight of the Infrared receiver that is situated on the robot. Therefore, commands that are sent to the robot from the PDA cannot be accidentally sent to other devices instead.

Although Infrared operates on the line of sight technique and it is usually troublesome to use in some controls, this was not a problem between the PDA and the robot as the PDA was attached to the back of the robot.

Figure 1. Picture of the PDA screen that is displayed on the host computer
The wireless LAN (or WiFi) was used to communicate between the host computer and the PDA. The host computer has 802.11g wireless capabilities but is also compatible to 802.11b standards. The PDA has built in WiFi, and thus a wireless network can be set up between the PDA and the host computer. This wireless network enables the host computer to connect to the PDA and even share the same internet connection. This was very useful and was used to view the PDA screen on the host computer and give commands to be send to the Robosapien robot. An overview of the communication system between the Robosapien robot and the host computer is shown in figure 3.

**Figure 2.** Robosapien with the PDA on its back

**Figure 3.** Flow diagram showing communication methods between the PC and Robot using the PDA
3. Communication and Software Interaction

The task that the Robosapien robot is to perform, involves picking up pipes from a table and inserting them in a subassembly, which could represent any part that is serviced or replaced in a manufacturing environment. The tasks are accomplished by two Robosapien robots. There are three pipes to be picked and inserted into a block with three holes. The lead robot will insert the first pipe in the first hole of the block, and then the slave robot will insert the second pipe in the second hole of the block. In order to model this task, it is important to determine the work envelope and all robot limitations relating to the task.

The robot’s work envelope was computed with relation to the model. Figure 4 shows a rough sketch of robosapien with numbered joints indicated by asterisks.

![Figure 4](image)

**Figure 4.** Rough 2D sketch of the Robosapien robot and its joints

3.1 Joints

Robosapien has rotational joints only; there are no direct translations that can be performed by the robot. The joints rotational angles numbered one to four in figure 4, were obtained by executing commands on the remote control and can be represented as follows in figure 5:

![Figure 5](image)

**Figure 5.** Joints

3.2 Object description
The pipes 30mm in diameter and 80 mm in length will be on top of a table 140 mm in height. The pipe is described by a coordinate system bearing a fixed relationship to it (see figure 6). Joint 3 is taken as a reference coordinate system. The center line of the pipe must be described by a vector 100i+210j, from the origin of the fixed coordinate system. The transforms are described in terms of direction cosines from the coordinate system axes.

![Figure 6. Pipe definition](image)

A graphical representation of the pipe and its coordinate axis is as shown in figure 6. A specification of the position and orientation of the coordinate system is sufficient to reconstruct the object in any other position and orientation.

### 3.3 Task description

Transformations are used to describe a task. The task consists of the two robosapien robots picking up pipes from a table and inserting them into a subassembly. Figure 7 illustrates the concept of one robosapien robot ready to perform the task.

![Figure 7. Task](image)

By defining a series of the arm and end effector positions with subscript pn, the task can be described as a sequence of arm moves and actions. Referring to these positions, the task positions can be described as:

| MOVE    | p1   | Approach the pipe |
|---------|------|-------------------|
| MOVE    | p2   | Move over the pipe|
| GRASP   |      | Grasp the pipe    |
| MOVE    | p3   | Lift it vertically|
MOVE p4 Approach hole at an angle
MOVE p5 Stop on contact with the hole
MOVE p6 Stand the pipe up
MOVE p7 Insert the pipe
RELEASE Let go of the pipe
MOVE p8 Move away

The above description is very rigid, yet similar actions are performed with the removing and insertion of modules or parts within a manufacturing environment.

3.1.1. Structure of the Task
The structure of the task can be firstly defined by defining the structure of the arm. The arm is defined by the product of three transformations such that the positions in the task description are replaced by

\[ MOVE_{pn} = MOVE Z T_6 E \]

where:
- \( Z \) represents the position of the arm with respect to the base coordinate system;
- \( T_6 \) represents the end of the arm with respect to joint 3;
- \( E \) represents the end effector at the end of the arm.

With such a description, the calibration of the arm to the work station is represented by \( Z \). The structure of the task is now represented in terms of the following transforms:

- \( P \) position of the pipe in base coordinate;
- \( H \) the position of the block with three holes;
- \( ^iHR \) the position of the \( i^{th} \) hole in the block with respect to the H coordinate system;
- \( ^pPG \) the position of the gripper holding the pipe with respect to the pipe;
- \( ^pPA \) the gripper approaching the pipe;
- \( ^pPD \) the gripper departing with the pipe;
- \( ^HRPHA \) the pipe approaching the \( i^{th} \) hole;
- \( ^HRPCH \) the pipe in contact with the hole;
- \( ^HRPAL \) the pipe at the beginning of insertion;
- \( ^HRPN \) the pipe inserted.

The task can now be represented as a series of transform equations solvable for \( T_6 \), the control arm input, as shown in figure 8.
Figure 8. Task Position Transform Graph

The above represents the essential structure of the task, and each transformation represents a separate piece of information. In order to define the transforms a combination of definition and teaching-by-doing will be used. Some of the transforms are logically defined symbolically, such as HR_i, which may be obtained from engineering drawings.

It is important that transformations representing pure translation or rotation are defined as such, for this information can be used to speed up and simplify matrix multiplication during task execution. Other transforms are logically defined by moving the arm from position to position, obtaining T_6, and solving for required transforms. The coordinate frames corresponding to P, H, and Z are shown in figure 9. While the interrelationship of these three coordinate frames is fixed by the physical object, the location of the base coordinate system is arbitrary, and can be specified in terms of any of the three frames.

Figure 9. Task frames P, H and Z

The base frame will be specified in terms of Z. While the arm coordinate frame is located in the shoulder, it is appropriate for the base coordinate frame to be at joint 3 or at T_6 p_z = -45. The arm cannot reach joint 3, so it will be set behind the origin of base coordinate such that T_6 p_x = 0 and T_6 p_y = -135.
So

\[
Z = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & -135 \\
0 & 0 & 1 & -45 \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
= I
\]  

(1)

where \( I \) is the identity transform, and thus

\[
Z = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 135 \\
0 & 0 & 1 & 45 \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
\]  

(2)

The end effector transformation is shown in figure 10.

**Figure 10.** End effector transformation

The end effector is defined by a transform with respect to the end of the arm. Using the convention that the \( z \) axis of the end effector is in its direction of approach to a task and the \( y \) axis describes orientation, the end effector shown in figure 10 is described as:

\[
^7E = \begin{bmatrix}
1 & 0 & 0 & 100 \\
0 & 1 & 0 & 75 \\
0 & 0 & 1 & 75 \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
\]  

(3)

While the pipe have been described in figure 7, a part with three holes is now looked at, \( H \). At top elevation its features are defined by means of a transform array \( HR \) (see figure 11).
Figure 11. Block with three holes

\[ HR_1 = \begin{bmatrix} 1 & 0 & 0 & 130 \\ 0 & 1 & 0 & 50 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \]  
(4)

\[ HR_2 = \begin{bmatrix} 1 & 0 & 0 & 80 \\ 0 & 1 & 0 & 50 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \]  
(5)

\[ HR_3 = \begin{bmatrix} 1 & 0 & 0 & 30 \\ 0 & 1 & 0 & 50 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \]  
(6)

The most important transform is the pipe inserted in a hole (see figure 12). The z axis of the pipe must agree with the axis of the hole. The direction of x and y is arbitrary, as the pipe has cylindrical symmetry.

Figure 12. Pipe inserted in Hole

The gripper approaches and grips the pipe from its initial position by rotating position \( T_6 \) by \( \theta_2 \) to the right about the y axis. The transform results:

\[ ^pPA = ^pPG = \begin{bmatrix} \cos \theta_2 & 0 & \sin \theta_2 & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \theta_2 & 0 & \cos \theta_2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \]  
(7)

The departure position of the gripper holding the pipe with respect to the pipe will be a result of rotating position \( T_6 \) by \( -\theta_2 \) about the y axis from the approach transform. The transform results:
\[^{\text{PD}} = R(y, -\theta_{22})R(y, \theta_{2}) \]
\[
\begin{bmatrix}
  f & 0 & -g & 0 \\
  0 & 1 & 0 & 0 \\
  g & 0 & f & 0 \\
  0 & 0 & 0 & 1
\end{bmatrix}
\]

(8)

To insert the pipe the arm is again rotated by \( \theta_{11} \) upwards about the y axis of Z, and is again rotated by \( \theta_{22} \) to the right about the y axis of position \( T_{6} \). The transform with the pipe in the hole is given by
\[^{\text{HRPN}} = \text{Rot}(y, \theta_{22})\text{Rot}(y, \theta_{11})^{\text{PD}} \]
\[
\begin{bmatrix}
  m & 0 & -n & 0 \\
  0 & 1 & 0 & 0 \\
  n & 0 & m & 0 \\
  0 & 0 & 0 & 1
\end{bmatrix}
\]

(9)

The remaining transforms can now be determined by using the arm. The end effector is placed on the pipe at its pick up position and the following transform equation is true:
\[ Z_{T_{6}}E = P \text{PG} \]

(10)

which defines \( P \):
\[ P = Z_{T_{6}}E \text{PG}^{-1} \]

(11)

It will be assumed that the approach position is the same as when the pipe is contact with the grippers during pick, so:
\[ Z_{T_{6}}E = P \text{PA} \]

(12)

A departure point relative to \( P \) is now defined by rotating the pipe in the gripper to the departure position \( PD \), resulting:
\[ Z_{T_{6}}E = P \text{PD} \text{PG} \]

(13)

which defines \( PD \)
\[ PD = P^{-1}Z_{T_{6}}E \text{PG}^{-1} \]

(14)

The position of the block \( H \) is defined by:
\[ Z_{T_{6}}E = H \text{HR}_{1} \text{PN PG} \]

(15)

and solving for \( H \)
\[ H = Z_{T_{6}}E (\text{HR}_{1} \text{PN PG})^{-1} \]

(16)
The pipe at the beginning of insertion is defined by:

\[ Z_{T_6}E = H_{HR_1}PALPG \]  

and

\[ PAL = (H_{HR_1})^{-1}Z_{T_6}E_{PG}^{-1} \]  

The pipe on first contact with the PCH and an approach point PHA to the contact point are defined by

\[ PCH = (H_{HR_1})^{-1}Z_{T_6}E_{PG}^{-1} \]  

\[ PHA = (H_{HR_1})^{-1}Z_{T_6}E_{PG}^{-1} \]  

which completes the definition of all transforms.

For insertion of the second and third pipe the same program should be executed, the only thing that is going to change is the position of the hole from \( HR_1 \) to \( HR_2 \) and \( HR_3 \) respectively.

4. Results and Discussion

The Robosapien robot was programmed to perform a specific task of picking up a cylinder and inserting it into a hole. Due to the lack of motion (minimal degrees of freedom (DOF) with the different joints), the task were seldom performed accurately, even if it was possible to control Robosapien from the host computer to guide its motion.

The right end effector could only open to approximately 45 mm in diameter. So the object that had to be picked have to be less than 45 mm. The left end effector can only open to about 25 mm in diameter, so the object that had to be picked by the left hand had to be less than 25 mm.

The Robosapien robot can only pick objects from the side due to the limited DOF and can therefore not pick up objects from its front and from its back. The grippers could not close at the exact pick up point due to the limitation of friction on the actuators. The robot is limited to pick up objects that are very light in weight and which are not fragile, because it releases the objects and does not put them down gently. The weight of the object affects the stability or the balance of the robot.

Even the above mentioned limitations of the Robosapien robot, it could be assumed that these limitations would not occur with well designed and developed humanoid robots. To reflect back to the question that was indicated in the introduction: Would it be feasible for a company to consider a humanoid robot and what other modifications and costs might be associated with the new technology?

Humanoid robots for manufacturing environments will be pre-programmed to control the manipulators. Feedback vision system will also be onboard of the humanoid robot for guidance and navigation to certain areas, but more importantly to identify objects and units that are to be maintained and serviced. These units could be unique for each manufacturing environment, as each company could prefer to use specific makes and models which could have different configurations. Unless the manufacture of the humanoid incorporates all models and makes within the onboard database, each company will need to train the humanoid robot to identify the units and the modules that will be required to be replaced. The training could consist of uploading of photos of different angles the units onto the humanoid robot. The above description will allow for generic humanoid robots to be built and be adapted to different manufacturing environments.
As per the research performed on the Robosapien robot, there was obviously an onboard communication system and a wireless network communication system that was required. The onboard communication system would be the responsibility of the mechatronics integration performed by the manufactures of the humanoid robot. The wireless network communication system would be the company’s responsibility and access points will be required to be installed at short distances from each other, so that their ranges overlap and allow coverage throughout the manufacturing environment. Similar protocols as the Robotics Communication Protocol (RCP) [5] could be used to transfer instructions and commands between the access points, machines, robots and humanoid robot. Cellular networks could be an option of communication, but there is a prone for the loss of signal due to metallic structures and building material.

A vision system will also need to be placed within the manufacturing environment, to allow the server to use image processing techniques to identify the humanoid robot’s location. Global Positioning System (GPS) would not be a possible location method, as the satellite signal will not be received through the metal and building material. The accuracy of GPS is also decreased if the satellite signals are bounced about the building structure.

The standard localization method used within the plant will also need to be installed for other robotic platforms and machinery. The localization system incorporated with the wireless system and any other sensory system located within the manufacturing environment will be required to be controlled by the same server, or at least a network of servers that would be able to communicate with each other different types of information. The server would be able to, in turn, send instructions to the humanoid robot of different tasks that have to be executed, similar to those performed form the host computer to the Robosapien robot. The instruction set would be dependent on the level of autonomy and intelligence of the humanoid robot to perform tasks. The instruction set could be as complex as: LOCATE unit XYZ, MOVE to unit XYZ, OPEN panel, DETECT module xyz, REMOVE module xyz, THROW removed module xyz into trash, INSERT new module xyz into unit, CLOSE panel, AWAIT for new instruction. Alternatively, for more advanced humanoid systems, the instruction set could be as simple as: LOCATE unit XYZ, REPLACE module xyz, AWAIT for new instruction.

5. Conclusion

A humanoid robot will only be useful in a Smart Manufacturing and Industrial environment. Companies that have already been established as a Smart Environments would benefit from the humanoid robot, as the only cost to the company would be that of the humanoid robot. Unless a company has budgeted to change into a Smart Environment for other robotic and sensory systems, the company will have to consider carefully whether it is beneficial and efficient to introduce a humanoid robot to their manufacturing environment. The reason for this statement is that the company will have to install a sensory system and a wireless network within the manufacturing environment before the humanoid robot can be operated. The cost of the systems that have to be installed to make the plant a Smart Environment, could be unexpected and high.

Saving human lives and the prevention of people involved in accidents are of high importance within a manufacturing environment. Due to this, the question is therefore raised: How much money is too much money to replace people with the involvement of dangerous environments and accidents? The answer to this problem is that it might be cheaper for the company to purchase multiple cheaper maintenance robots that is only able to perform a single task than a humanoid robot. In this way humans are saved from being less involved in manufacturing accidents.

As manufacturing environments evolve and improve, companies will be forced to start implementing Smart Environments. It will take time before these Smart Environments are established to introduce humanoid robots. Once these Smart Environments are established, it will definitely be beneficial for
companies to purchase humanoid robots that will assist with dangerous tasks and maintenance within the manufacturing environment.

[1] Aderhold J, Davydov V Yu, Fedler F, Klausing H, Mistele D, Rotter T, Semchinova O, Stemmer J and Graul J 2001 J. Cryst. Growth **222** 701

6. References

[1] Järvinen, J. & Karwowski, W. 1995 Analysis of self-reported accidents attributed to advanced manufacturing systems, *International Journal of Human Factors in Manufacturing*, Wiley Publishers **5** Issue 3, 251 – 266.

[2] Asimo by Honda, The World’s most Advanced Humanoid Robot, http://asimo.honda.com [accessed 18/01/2013].

[3] NASA – Robonaut 2, http://www.nasa.gov/mission_pages/station/main/robonaut.html [accessed 18/01/2013].

[4] Padayachee, J., Bright, G. & Masekamela, I. 2009 Modular Reconfigurable Machine Tools: Design, Control and Evaluation, *South African Journal of Industrial Engineering* **20**, No 2, 127 – 143

[5] Stopforth, R., Bright, G., Davrajh, S., Walker, A. 2011 Improved Communication between Manufacturing robots, *South African Journal of Industrial Engineers* **22**, Issue 1, 99 - 107