

# The web-based programming interface for the Mitsubishi Movemaster robot

**K. Foit\***

Institute of Engineering Processes Automation and Integrated Manufacturing Systems, Silesian University of Technology, ul. Konarskiego 18a, 44-100 Gliwice, Poland

\* Corresponding author: E-mail address: krzysztof.foit@polsl.pl

Received 27.01.2008; published in revised form 01.04.2008

## Manufacturing and processing

### ABSTRACT

**Purpose:** of this paper. The aim of this paper is to present a prototype of web-based programming interface for the Mitsubishi Movemaster RV-M1 robot.

**Design/methodology/approach:** In the previous papers [11-14] the off-line, remote programming system for the mentioned robot has been presented. It has been used as a base for developing a new branch: web-based programming interface. The web techniques have been selected due to possibility of use existing code fragments for elaborating new applications and modularity of this solution.

**Findings:** As a result, a prototype of the system has been developed.

**Research limitations/implications:** Because the presented system is in the early development stage, there is a lack of some useful functions. Future work will include elaboration of the robot's visualisation module and implementation of a trajectory translator intended to co-operate with CAD software.

**Practical implications:** The elaborated system has been previously intended for educational purposes, but it may be adapted for other devices, like small PLC's or other robots.

**Originality/value:** Remote supervision of machines during a manufacturing process is an actual issue. Most of automation systems manufacturers produce software for their PLC's and robots. Mitsubishi Movemaster RV-M1 is an old model and there is very few programs dedicated to this machine. On the other hand the programming and development of applications for this robot are very easy. The aim of the presented project is to develop a flexible, remote-programming environment.

**Keywords:** Automation engineering processes; Robotics; Mechatronics; Technological devices and equipment; Numerical techniques

## 1. Introduction

Every contemporary factory uses industrial networks for a data exchange between machines. The network also connects different levels of a company, giving the possibility of analysing and change of production parameters from an office room. Programs for controllers are often prepared and tested on computers, using a virtual environment, then are uploaded to a real device [1-9].

In the papers [11-14] the off-line programming and simulation package has been presented. The software package has been

written in a high level, object oriented programming language and has consisted of three applications:

- The server module – SERWER,
- The programming and simulation module – ROBO,
- The Internet camera and robot control module – KAMROB.

The complete system has been designed to operate with the Mitsubishi Movemaster RV-M1 robot by using a typical PC computer and serial connection between the robot's control unit and the computer. The package has been intended to run on the Microsoft Windows platform with the use of a network connection.

The Mitsubishi Movemaster RV-M1 robot has a five-axis manipulator with 1,2 kg lifting capacity. The controller of the

robot is equipped with an array of inputs and outputs. Moreover, it has a possibility of connecting it to a computer using Centronics or RS232 port. In spite of small size, the robot is a typical industrial machine. Programming of the robot can be carried out by using a teach box or a personal computer. In case of using a computer, program is prepared in Movemaster Commands language, which has very simple syntax and allows writing of the code in any text editor. It is also possible to use the batch mode when command has been executed immediately after sending it to a robot's controller [15]. Described properties of the Movemaster Commands language allow a user to quickly create its own communication software for co-operation with the robot.

The software described in [11-14] is intended for the Windows platform only. The main disadvantage is that it requires installation and it cannot be easily ported to the other operating system platform. In the papers [10-14] an idea of the platform-independent system has been presented, but only as the general concept. This paper presents more details of this idea and describes a prototype of the software.

## 2. The existing system and the concept of the new one

As it was mentioned earlier, the existing system consists of three applications: the server module, the programming and simulation module and the Internet camera and robot control module. The example of a hardware configuration, used with this software package is shown in Figure 1.

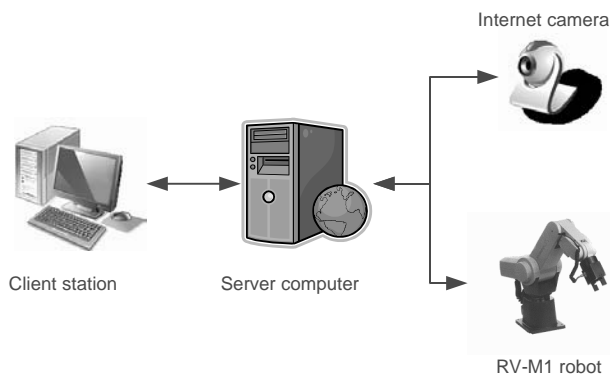


Fig. 1. The example of a hardware configuration used with ROBO software package

As it can be seen, it is a typical configuration with a server on the one side and a client on the other one. The ROBO application is installed on the client side and the SERWER and KAMROB should be installed on the server computer. There can be one server for both applications or two machines – one for handling the robot's controller and the other for the Internet camera control.

The ROBO application is used for off-line programming and simulation issues. It is also possible to generate a program using a

DXF file, which can be created in a CAD program (for example in AutoCAD). A path is coded by a line entity drawn in 3-D. Each point represents a position of the robot's wrist. The status of the gripper (open or closed) is represented by a point entity. The application is using a dictionary file to translate DXF commands into proper Movemaster Commands directives.

The main window of the program consists of two areas (Figure 2): a control panel and a simulation window. The simulation routine uses the OpenGL libraries to achieve the exact imitation of a real environment. Moreover, there is possible to add some type of equipment to the virtual environment: besides an object of manipulation, tables and shelves can be added to the robot's scene. The application can also handle more than one virtual robot, but this feature requires at least two copies of the ROBO program connected to the same server.

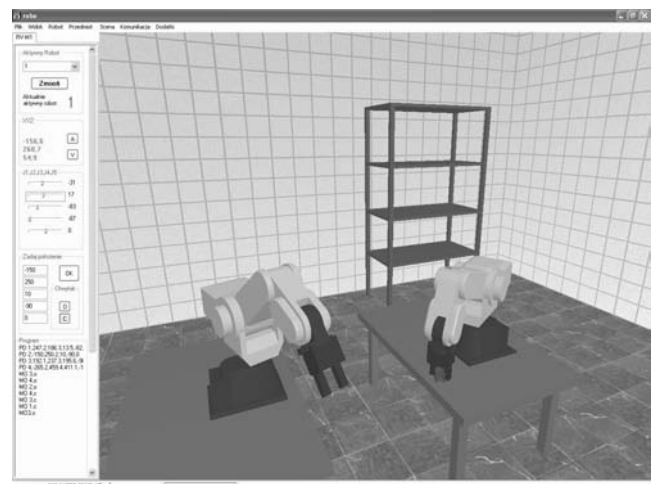


Fig. 2. The main window of the ROBO application with two robots in simulation area

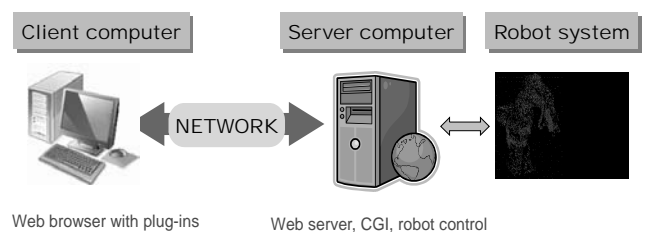


Fig. 3. The concept of the web-based programming system

All advantages of the system are described in details in the papers [10-14]. It should be remarked that this package has been developed in collaboration with a student and is intended for educational purposes only. The main drawback of this system is that it requires installation on the client and server side and it works only under Windows operating system. In the papers [10,14] an idea of a platform-independent system has been shown. There has been one solution based on a web server (Figure 3), and the other

based on the Java environment (Figure 4). In the further part of this paper, a selected method will be described.

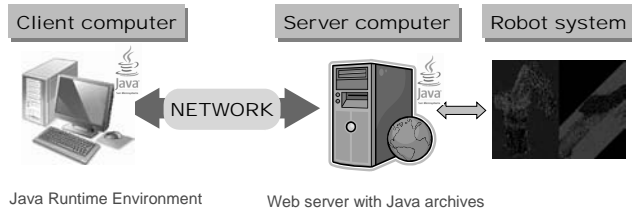


Fig. 4. The concept of the Java-based programming system

## 2.1. The concept of the new system

For the further development the method based on a web server has been selected. It has some benefits in comparison with the Java-based system:

- there is no need to translate the source code of the previously developed system,
- the new applications can be created on the base of the existing source code in the case that the server will be run on the Windows operating system,
- there is no need to develop a graphical user interface for any new application – the interface is being provided by the web resources,
- a high level of modularity can be achieved, because applications can be developed as independent modules used “on demand”,
- the user interface can be easily modified and developed from scratch to the advanced one,
- there is possibility of using secure connections certificates and distributed services,
- there can be alternate user interfaces intended to use on a different operating system and browser platforms.

It has been assumed, that the first version of the web-based programming system has not been equipped with 3-D simulation module. It is planned to enable this feature in further versions using the Adobe Flash or the VRML/X3D plug-in. The Internet camera functionality has been also disabled. These features have not been important at the first stage of development. On the other hand, special software may be required to develop them.

## 3. The first version of web-based programming interface

The first version of web-based programming interface for the Mitsubishi Movemaster RV-M1 robot has been based on a web server running on Windows operating system and CGI programs. CGI stands for Common Gateway Interface and it is a protocol for interfacing external applications with a server. This means that a program installed on the server side could be started-up by the web server service, after receiving information from a remote web browser. In this manner, any application or script can be called.

After logging-in, the main page of the interface is displayed in a web browser. A screen area is divided between a menu frame (on the left side) and an operation frame (Figure 5).

The Control Panel menu contains three commands:

- Nest – sets the manipulator to the origin position (zero),
- Run – executing the program,
- Clear program – erases the program from the robot’s memory,
- Clear positions – erases the positions from the robot’s memory.

The *Program* and *Positions* menus are intended for manipulating with the robot’s program and positions stored in the robot’s controller memory. The program or positions can be sent to or received from the controller, may be also saved to a file or received from. These pages also contain the command for erasing all the positions or the program from the controller’s memory.

MENU	CONTROL PANEL
Control Panel	NEST Executes the NEST command. This moves the manipulator to the origin position
Program	RUN Executes the program stored in the robot's controller memory
Positions	CLEAR PROGRAM Clears the program stored in the robot's controller memory
	CLEAR POSITIONS Clears the positions stored in the robot's controller memory

Fig. 5. The Control Panel page from the web-based programming interface

The request of a command execution is carried to the server and managed by a proper service then an external program is called. In this case, there are two applications: the first one elaborates incoming data and the second realises the communication between the server station and the robot’s controller. A user gets confirmation for every executed command including a short report about status.

## 4. Conclusions

The aim of this research work is to elaborate a platform-independent system for programming the Mitsubishi Movemaster RV-M1 robot. Presented, web-based structure should be regarded as a project in the early development stage. In the future the project will be supplemented with possibility of the robot’s visualisation and the translation of a trajectory drawn in a CAD program.

The system is developed for educational purposes so far, but it may be adapted for other devices, like small PLC’s or other robots.

## References

- [1] T. Otrębski, Remote monitoring and control using GPRS service, Control Engineering Poland 8/36 (2006) 62-64 (in Polish).

- [2] Presher, The network control of machines – in the spotlight, *Design News* 9/16 (2006) 48-51 (in Polish).
- [3] S.F. Chan, R. Kwan, Post-processing methodologies for off-line robot programming within computer integrated manufacture, *Journal of Materials Processing Technology* 139 (2003) 8-14.
- [4] Yasuda, Distributed autonomous control of modular robot systems using parallel programming, *Journal of Materials Processing Technology* 141 (2003) 357-364.
- [5] G. Kost, R. Zdanowicz, Modeling of manufacturing systems and robot motions, *Journal of Materials Processing Technology* 164-165 (2005) 1369-1378.
- [6] J. Alvares, J. C. Espindola Ferreira, WebTurning: Teleoperation of a CNC turning center through the Internet, *Journal of Materials Processing Technology* 179 (2006) 251-259.
- [7] T.S. Mujber, T. Szecsi, M.S.J. Hashmi, Virtual reality applications in manufacturing process simulation, *Journal of Materials Processing Technology* 155-156 (2004) 1834-1838.
- [8] Q. Peng, F.R. Hall, P.M. Lister, Application and evaluation of VR-based CAPP system, *Journal of Materials Processing Technology* 107 (2000) 153-159.
- [9] W.B. Lee, C.F. Cheung, J.G. Li, Applications of virtual manufacturing in materials processing, *Journal of Materials Processing Technology* 113 (2001) 416-423.
- [10] K. Foit, J. Świder, The project of a platform-independent, off-line programming system for industrial robots, *Proceedings of the 7<sup>th</sup> International Scientific Conference “Computer Integrated Manufacturing – Intelligent Manufacturing Systems CIM’2005”*, Gliwice – Wisła, 2005, 62-65.
- [11] K. Foit, J. Świder, D. Mastrowski, The project of an off-line, remote programming system for Mitsubishi Movemaster industrial robot, *Proceedings of the XIII International Scientific and Technical Conference “Machine-Building and Technosphere of the XXI”*, Sevastopol, Ukraine, Paper Collection Vol. 4, 2006, 252-255.
- [12] D. Mastrowski, The system for remote programming, supervision and simulation of robots, cooperating with Mitsubishi Movemaster RV-M1 Robot. MSc thesis, Gliwice 2006 (in Polish).
- [13] J. Świder, K. Foit, G. Wszolek, D. Mastrowski, The off-line programming and simulation software for the Mitsubishi Movemaster RV-M1 robot, *Journal of Achievements in Materials and Manufacturing Engineering* 20 (2007) 499-502.
- [14] J. Świder, K. Foit, G. Wszolek, D. Mastrowski, The system for simulation and offline, remote programming of the Mitsubishi Movemaster RV-M1 robot, *Journal of Achievements in Materials and Manufacturing Engineering* 25/1 (2007) 7-14.
- [15] Mitsubishi Movemaster RV-M1 User’s Manual.