FUZZY-GAIN SCHEDULING CONTROL ARCHITECTURE FOR LARGE RANGE ROBOTS

E. Gambao, A. Barrientos, C. Balaguer[*], R. Saltaren, R. Aracil
DISAM
Polytechnic University of Madrid, Spain
robotica@disam.upm.es
fax : + 34 1 5642961

Keywords : Robot control, Hydraulic robots, Large range robots.

Abstract :

1.- INTRODUCTION

Typical robots applications needs medium size robots with 5 to 50 kg. payload capacity and 0.1 mm. of TIP precision. Classical feedback PID control schemes with feed-forward and other well-known control schemes are normally sufficient for these kind of robots application and is thoroughly used by industrial robots.

However robot control is one of must difficult testbeds for control theory. A multilink robot is a multivariable, parameters variant, non-lineal system where classical control st5rategies can be apply only if several simplification and assumption are accepted. Technical literature report an ample collection of robot control proposal, but his real implantation is limited because the complexity of algorithm make impossible his application out of papers.

Today robotics go-out of factory and new applications of robots are considered. Between those several needs large robots with a large workarea (5-20 m) and high payload capacity (100-500 kg.). Robot for space, mining , building construction, civil construction and others sectors are some of these applications. In these cases hydraulic actuators are frequently used because his high ratio robot weight/payload .

In these kind of large robots with hydraulics actuators, to assume that the pay-load, inertia momentum, actuators characteristics etc. are constant is impossible and PID control, based in constant lineal model is totally inefficient.

Therefore, to control these new kind of robots, robotics research are applying all the new control methods trying to achieve high performance with a reasonable time consuming algorithm, usable in a real time control system.

Adaptive control is one of the must attractive solution for robot control. Model variation and imprecision can be theoretically avoided by an on-line model identification and a on-line PID or other control strategy parameters tuning. Nevertheless again real time response of the system together the difficulty to specify rules to tune the control difficult strongly the implementation of this method in robots.

Gain-scheduling can be considered as an off-line version of adaptive control, where controller tuning for a finite set of different situations is made out of line. In real time, one of the parameters sets for controller is selected, according the instantaneous situation of the system. The use of a controller tuned for a situation different of the actual can be inefficient and in some case originate instability.

Fuzzy-logic philosophy can be an adequate strategy to define the rules to adjust the control parameters, because control rules are defined in a intuitive way, without strict algorithm formulation.

Combining both philosophies (Gain-scheduling and fuzzy-logic) a control system for 2 large size robots has been developed. These robots are part of the ESPRIT 6450 project ”Robot system for computer integrated construction”. Both are developed to perform blocks pick and place and others task in buildings constructions.

In this paper, after a short description of the ROCCO project objectives and robots characteristics, the hardware and software control architecture is described. Special attention is made to control technique selection. As this kind of application is highly non repetitive and non structures, man-machine interface is specially important , because no specialist operator must frequently interactuate with the robot control .

2.- ROBOT SYSTEM FOR COMPUTER INTEGRATED CONSTRUCTION. ESPRIT 6450 ROCCO PROJECT.

The main objective of ROCCO Esprit project is to develop an integrated system for the automation of the construction process, including :

- Assisted drawing of the complete building using CAD
- Working planning development, including facade partitioning in elementary parts and on site work lay-out planning.
- Generation of manufacturing commands for the fabrication of blocks, panels and others.
- Development of the logistics of the on-site resources supply
- Robot programs generation
- Development of the robotics system

The robotics system, includes two robots destined to manipulate and assembly construction blocks of maximum dimension of 100 cm x50 cm x 50 cm and near of 400 kg of weight. One of the robot (the little one) assembly blocks in house building. This robot is mounted in autonomous vehicle and has a working area of 4.5 m. with 5 DOF.

The second robot is dedicate to assembly industrial building, with typical eight of up to 8 m and standardised lay-out. The robot (fig 1) has a working area of 8.5 m and it is able to manipulate payloads up to 500 kg. The robot has 6 DOF driven by hydraulic actuators using servo-valves. His accuracy is about 🕸 5 cm. The robot is mounted in a mobile platform. that permits to move the robot along the construction site. Because his large reaching area and high payload capacity, this robots is most difficult to control than the first one.

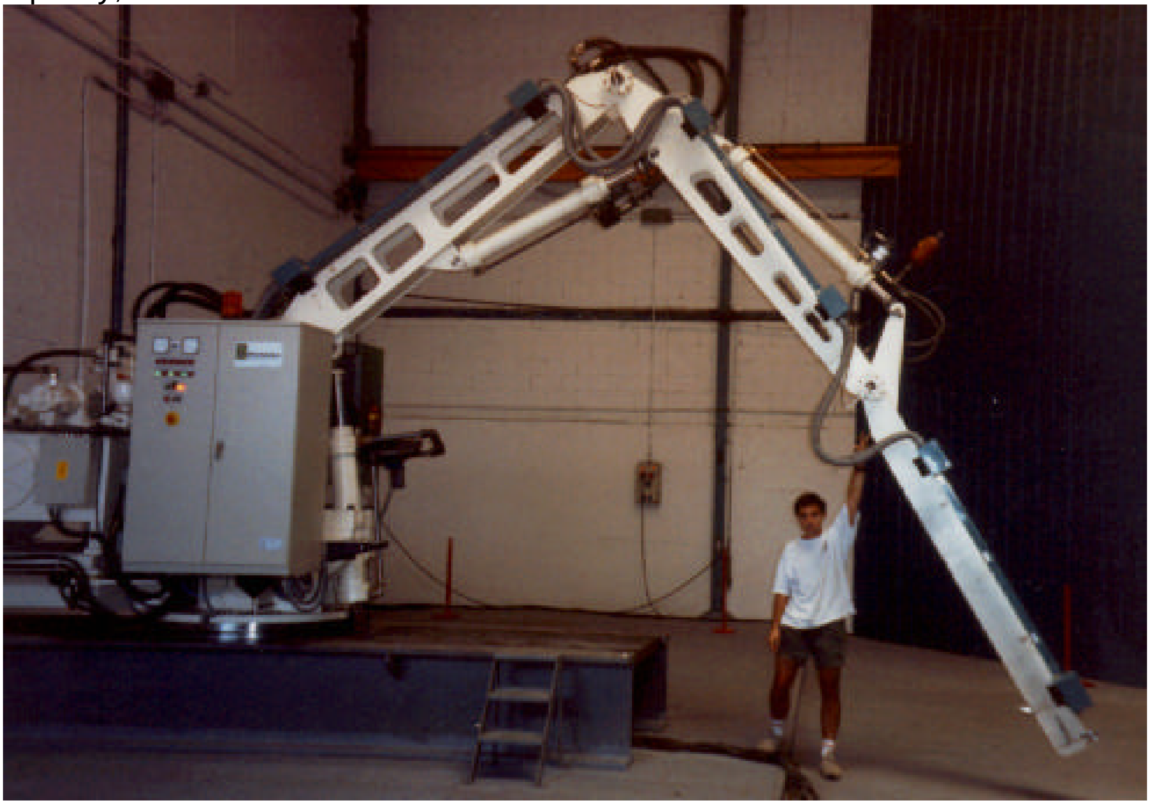

Figure 1. ROCCO ROBOT

3.- COMPARATIVE ANALYSIS OF ROBOT CONTROL TECHNIQUES

3.-1.- Control of Robot Manipulators

The purpose of robot arm control is to maintain the dynamic, tracking error or/and static responses in accordance with some prespecified performance criterion. In general, the control problem consists of obtaining dynamic models of the manipulator, and using these models to determine control laws or strategies to achieve the desired system response and performance.

Current industrial approaches to robot arm control system treat each joint of the robot arm as a simple joint servomechanism. This models the varying dynamics of the manipulator inadequately because it neglects the motion and configuration effects of the whole arm mechanism. These changes in the parameters of the controlled system are significant enough to render conventional feedback control strategies ineffective. The result is reduced servo response speed and damping, limiting the precision and speed of the TCP (Tool Center Point). Any significant performance improvement in this and other areas of the robot arm control require the consideration of more efficient dynamic models and sophisticated control techniques [Fu-87],[Slotine-91],[Lee-850],[Lee-82].

Considering the robot arm control as a path-trajectory tracking problem, motion control can be classified into three major categories [Fu-87]:

- Joint motion controls
- Resolved motion control (cartesian space control)
- Adaptive controls

It is possible to find a lot of references about the above exposed approaches. Not all this techniques are suitable for the ROCCO robots. Joint and resolved motion controls sometimes are inadequate because they require accurate modeling of the arm dynamics and neglect the changes of the load in a task cycle. These changes in the payload of the controlled system often are significant enough, like for the ROCCO robots, to render the above feedback control strategies ineffective. Any significant gain in performance for tracking the desired time-based trajectory as closely as possible, over a wide range of manipulator motion and payloads, require the consideration of adaptive control techniques [Vähä,84],[Lee-84],[Benckert-94].

3.2.- Hydraulic Large Range Robot Control

The position control of this kind of robots is a tough problem due to the extreme nonlinearities of the kinematic and hydraulic actuators. The nonlinearity depends both on the actual joint position and the velocity. An additional source of difficulty for the control design is the highly variable payload of the proposed robots as well as the possible elasticities of the mechanics. Because of the highly nonlinear behavior of hydraulic actuators and robot mechanics for the closed-loop position control of the robot joints sophisticated model-based concepts are required [Kuntze-91]. These non-linearities can be grouped in:

- Nonlinearities of the mechanics, e.g. changing of the load, nonlinear transfer functions of the kinematic chain.

- Nonlinearities of the hydraulic system, e.g. hysteresis of the control piston of the valve, nonlinear flow, changing of the oil viscosity with the temperature, etc.

Furthermore, the manipulator present other important difficulties due static deflections of the kinematic and the hydraulic actuators.

In spite of the lack of similar developed robots there are some references all over the world [Blume-89],[Kuntze-91],[Vähä-84],[Benckert-94]. There are several proposed control concepts and techniques such as:

Conventional Position Control

It is very similar to those ones of the most industrial robots based on classical control techniques. It is characterized by a cascaded signal structure in which the velocity control loop is superimposed by the position control loop. According to the sensor locations (axis, piston, etc.) different dynamics will be obtained. This type of control concept was implemented by [Kuntzw-91] using piston stroke as the controlled value. The results were acceptable but with a difficulty to compensate the dynamic deflection.

Force Feedback and Linearization

It is based on the use of sensor-based force feedback concept which implies the measurement of the pressures in both chambers of the hydraulic cylinders by means of highly linear transducers. The static nonlinearities caused by the hydraulic actuator and the robot kinematics can be approximately compensated by a static prefilter which comprises the inverted nonlinear analytical relations [Blume-89].

Feedback linearization

This method is useful when the model of the system is well known and when the model is highly non-linear. Input-controlled variables and input-output linearization techniques are used. The non-linearities are compensated in dynamical form. Nevertheless, the robustness of this type of control is not demonstrated yet, specially in the sense of the internal dynamics of the system [Fu-87].

Sliding Control

When the non-linearities of the system are not well known or unmodeled dynamics appears, the sliding control is one of the best techniques. The adequate selection of the sliding surface according the general sliding conditions, leads to the possibility of finite time control algorithm. Additionally, the selection of the boundary layers make the switching control in place of dither control more effective [Slotine-83]. On the other hand, the extremely difficulties in obtained adequate sliding surface of the systems with changing in time parameters, make difficult its practical implementation.

Robust control

When the system's model is not well known, the robust control, which guarantee the system's stability and its performances, can be used. The main idea of this technique is the development of not the best controller (optimum) but the controller which satisfactory works for a big range of changing parameters of the system. The design of the robust controllers is based on the minimization of the transfer function norm which includes both the changing parameters effects and the required performances. The most common used norms are H ▸ [Doyle-92].

Model Based Position Control

Several references are based on the use of model-based position control in order to improve the dynamical accuracy. Some of them are quite sophisticated like the *Predictive Functional Control* developed by the IITB (Fraunhofer-Institut fhr Informations und Datenverarbeitung) and the french research institute ADERSA [Kuntze-91]. Others use a standard stochastic linear state model to develop the control.

Adaptive Control

Due to the highly variation of the system parameters, several developments are based on the use of an adaptive or self-tuning control law [V ähä-84],[Lee-84],[Lee-85],[Benckert-94]. The online or offline parameter estimation is applied to change tuning of the controller. This method permits to get a good response of the system in any work condition (position, load, velocity, etc) by means of an appropriate selection of the controller parameters.

Structural static flexion needs special sensor to be detected and controlled. Several authors propose the use of an external sensor that provides the TCP position [Kuntze-91],[Vähä-84],[Benckert-94]. A much higher control performance can be expected with this sensor information.

After the review of the most useful robot control techniques and the similar existing systems the conclusions, focused on the ROCCO control implementation, are:

- Due to the highly nonlinear behavior of hydraulic actuators and robot mechanics for the closed-loop position control of the robot joints sophisticated *model-based concepts* are required.

- *Adaptive or supervised control* techniques result suitable to cope with these systems difficulties. These methods can be used with online and/or offline parameter estimation. If the conditions (load, position, velocity, etc) are well known, it is possible to use the offline identification experiments to adjust appropriately the controller parameters in a simpler way.

- The use of an *external sensor* that provides the real time TCP position results in a much higher control performance regarding the static deflections.

In this application Fuzzy techniques has been used to online adjustment of an adaptive control scheme.

4.- CONTROL ARCHITECTURE AND HARDWARE DESCRIPTION

4.1.- Basic Control Scheme

In order to present a simple but general scheme of the complete system it has been used a modular approach that represents the different existing subsystems. Figure 2 shows the proposed basic robot control scheme including the external position sensor for the industrial building application robot.

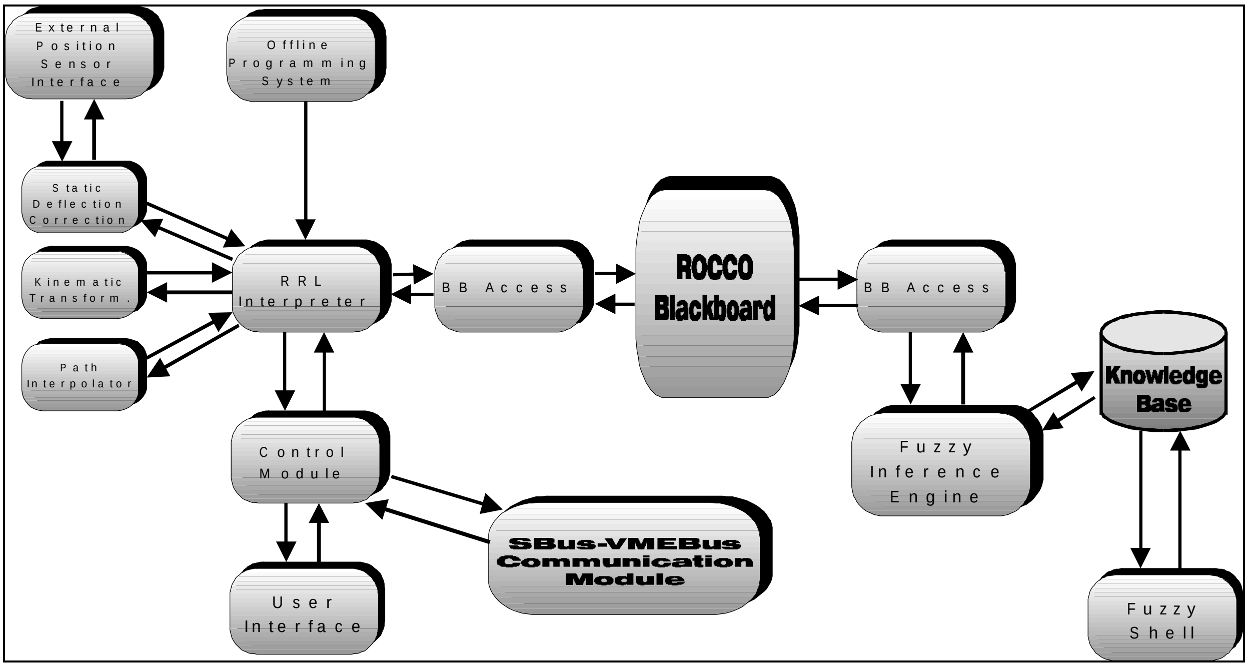

Figure 2. ROCCO General Architecture

- *Mechanical Structure (*Industrial Building Application)

The mechanical structure is formed by a 6 degrees of freedom (DOFs) robot. The robot has a maximum working radius of about 8.5 meters and it is able to support up to 500 kg .

- *Hydraulic actuators*

The robot has hydraulic actuators (Mannesman Rexroth). This motion system includes nonlinearities and elasticities that require a more sophisticated control to get a good response in spite of the changing parameters of the robot.

- *Axis Position Sensors*

Each robot axe has its own digital position sensor in order to know the actual position of the axe which permits to close the control loop.

- *Servovalves*

Each hydraulic actuator (Mannesman Rexroth) is controlled by means of a servovalve that supplies the appropriate oil flow to the actuator.

- *Servovalve electronics*

Each servovalve has an amplifier (Mannesman Rexroth) that receives the control signals from the robot and the vehicle control system and supplies the appropriate intensity value to the servovalve. It includes other important functions such as dither for the servovalves.

-*Robot Control System*

This system performs the robot and vehicle motion control by means of different boards attached to the bus. The programmable multiaxis controller board (Baldor-Delta Tau PMAC) performs the robot axis motion control in a coordinated way. Each robot axis has its own control loop and controller. In order to control the vehicle movement there is a VME CPU board (VRTX 32 operating system) and different input/output boards to handle signals between the control and the vehicle.

- *Supervisory Control System*

The on-line control system software is into the workstation, which will be described with detail later.

4.2.- Architecture Control System

Figure 3 shows a scheme of the general hardware architecture of the complete industrial building robot control system. The workstation (Sun Sparcstation 10) works as the main (host) computer (coordinator). It has attached a SBus-VMEbus adapter board (Performance Technologies PT-SBS915) and a serial-parallel controller patch panel.

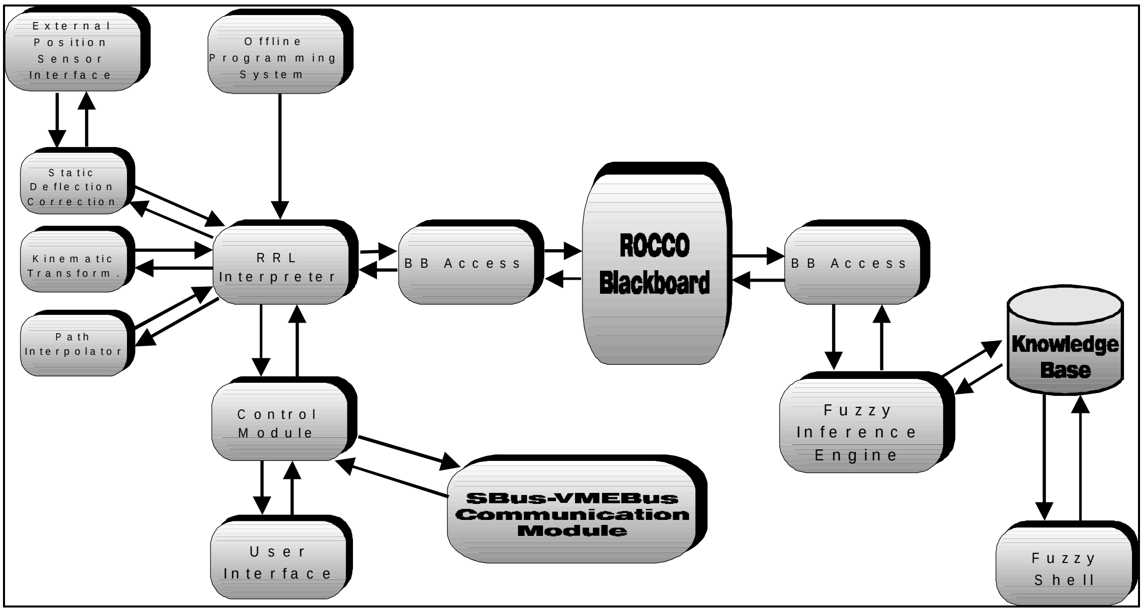

Figure 3. ROCCO General Architecture

There are another two boards into the VME bus chassis. The first one is a power supply and the second one is the Programmable Multiaxis Controller Board (Baldor Delta Tau PMAC). This board has a Motorola DSP 56001 Digital Signal Processor and it is able to control up to 8 axis with great flexibility and a lot of desirable features for the ROCCO robot system. The servo update rate is 40/60 microseconds/axis.

The PMAC board is connected to the servo hydraulic drives of the robot by using a servovalve amplifier for each one of them (only one is shown in the figure). The servo hydraulic drive is composed by a servovalve, an hydraulic drive and position measurement system (encoder).

In the other side, the serial-parallel patch panel attached to the workstation has connected an telemetric position sensor (GEODIMETER SYSTEM 600) that supply continuous information about the actual position of the robot end TCP (see deliverable 3.5). This information is used by the system to correct deviations of the robot due to statical errors and as a security level (to prevent big deviations of the prespecified trajectory). The telemetric position sensor is able to follow the robot TCP. The measurement time is, in normal mode, 8-10 seconds and could be up to 0.4 seconds in fast tracking mode. The precision of the position is ±6 mm (±15 mm in fast tracking mode) which results more than sufficient for the accuracy required.

The Table 1 shows the main sample times and positioning precision data of the involved systems.

PMAC	Servo update rate	40/60 :s/axis
GEODIMETER	fast tracking measurement time	0.4 s
	precision	"6 mm
	precision (fast tracking)	"15 mm
Sbus-VME bus adapter	transfer rate	11 MBytes/s

Table 1

The optical position sensor station is connected to the workstation by a serial RS-232 connection by means of a serial parallel controller patch panel. This system permits to place the optical sensor in the appropriate situation in order to get a good robot TCP autotracking

There is a specific system to connect the workstation with the VME bus. This system has two boards, one plugged into the workstation and one into the VME bus, that permit the Sbus to VME bus interconnection sharing data buses. This connection allows a flexible interconnection between both systems with a transfer rate capability up to 11 MBytes/sec. VME bus "masters" in an external chassis may access "slave" functions in the SBus environment. Conversely, Sbus "masters" may access VME bus "slaves" across the adapter. This provides a fully orthogonal hardware interconnect between the buses, allowing a high degree of flexibility to the user.

Finally, the workstation is connected via Ethernet network to the offline programming system.

5.- SOFTWARE ARCHITECTURE

The control system has a modular structure. The implementation is divided between the host computer and the PMAC motion control board. The modules implemented in the host are showed in figure 4.

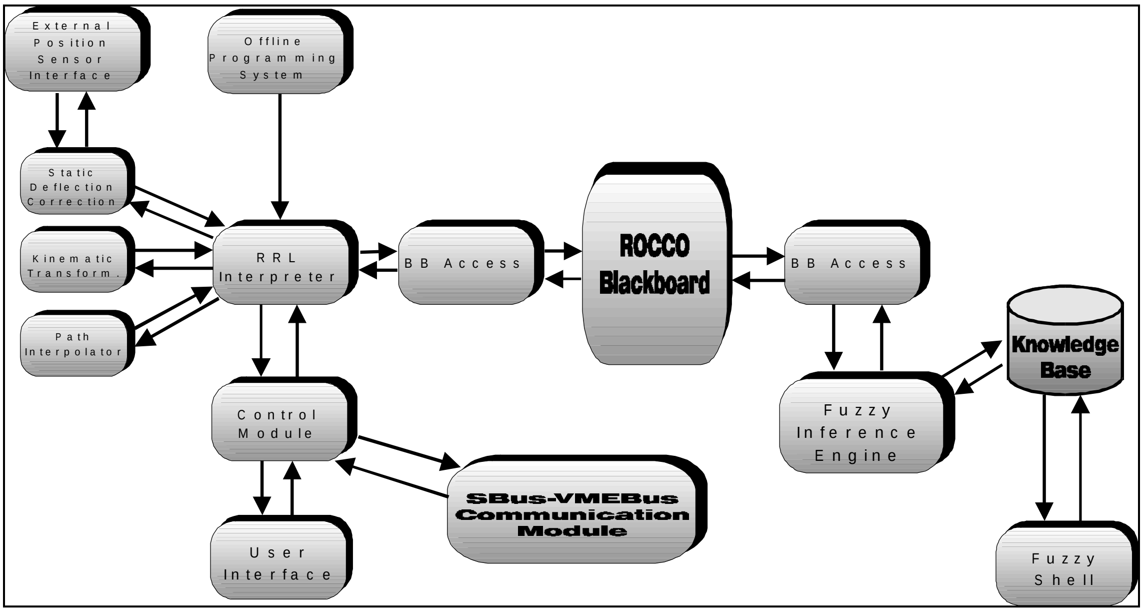

Figure 4 ROCCO Software Architecture

5.1.- Software Architecture of the High Level Control System (host)

Offline Programming System: provides procedures written in a specific command language (RRL). This enables a very flexible handling of unknown situations. If a minor fault like a position deviation occurs, this can be compensated automatically by the online system in real time. If a major fault occurs, the offline programming system is in charge of rebuild the corresponding program or to reparametrize the actions manually if necessary. Then it switches back to the online control for execution.

External Position Sensor Interface: receives, via RS-232 line, contiuous information about the real robot TCP position which is used in the security system and in to compensate the structure static deflections. This module makes a previous treatment of the sensor data to filter them and to resolve problems such as aliasing.

Stactic Deflection Correction Module: is in charge of the generation of trajectories that compensate possible TCP deviations. It receives information about the robot TCP real position and the expected position from the kinematic transformation of the axis sensors (encoders) to generate the trajectory correction.

Kinematic Transformation: performs the robot direct and inverse kinematic transformations

Path Interpolator: performs the high level interpolation of the robot trajectories.

RRL Interpreter: is in charge of the interpretation of the RRL instructions to PMAC language instructions. The RRL instructions are grouped in order to be sent to the motion control board forming a motion program.

Control Module: is in charge of to manage the communication between the different subsystems and the global system control. The error handling is performed in this module.

Users Interface: allows the communication with the user by means of a window based system. The User Interface allows to introduce commands and to control the global process.

Sbus-VMEBus Communication Module: performs the data interchange between the host computer and the motion control board (PMAC). The data are transmited in two different forms.A specific driver is needed to allow communication between the host bus (Sbus) and the bus adapter board (VMEbus).

The fuzzy logic based Model-Based Adjustment System is formed by:

Fuzzy Inference Engine: Implements the validation algorithms. It is based on specific kwoledge stored in the run time **Knowledge Base**.

Fuzzy Shell: off-line module that allows the expert to introduce and to modify the Knowledge Base by means of a very friendly graphical interface. The Fuzzy Shell is an expert oriented tool. The tool is in essence a knowledge base editor and C code generator.

ROCCO Blackboard: The comunication with the Model-Based Adjustment System is performed by means of a Blackboard (shared memory) which access is controlled by the **BlackBoard Access Modules**. This system provides fast data interchange as well as synchronization of the read/write procedures between the processes.

5. 2.- Software Architecture of the Low Level Control System

At the low level, the motion control board executes some programs to control the robot movements. Several programs are stored in the board EPROM and are able to manage the different signals and memory transformations to allow communication with the robot and the host. The programs that define the different robot trajectories are provided directly by the host via the Sbus-VMEbus Communication Module. This programs are not stored in the motion control board after executed.

The robot calibration is performed by means of EPROM stored programs which are activated by the user from the user interface. Other stored programs are automatically executed when the system is started to provide communication and signal processing.