NEW APPROACHES ON MODULAR WALKING ROBOTS WITH FORCE-POSITION HYBRID CONTROL

LUIGE VLĂDĂREANU*, ION ION**

Abstract. The paper presents the fundamental and applicative research regarding the elaboration of new concepts and new approaches on *modular walking robots force-position hybrid control* by developing and experimenting with a multiprocessor system, in real-time control and open architecture HFPC MERO. There are analyzed and developed the compliant control methods of modular walking robots with tracking functions, the "multi-stage" fuzzy control with two fuzzy control loops, one in position and the other in force and the open architecture multiprocessor system. With this approach, the HFPC MERO walking robot's control system ensures flexibility, short execution time, target precision and repeatability of the moving programs, eliminating the closed systems with projects meant for specified applications.

Key words: robotics, real time control, kinematics, open architecture systems.

1. INTRODUCTION

The locomotion activity of walking robots is in the category of high automation movements. The mechanical system needs to have a great number of mobility ranges, in order to form high complexity synergies, respectively the realization of coordinated motion of the legs. By using the walking robots as a means of transportation, a portion of the parameters which characterize its dynamic features can be submitted to modifications to a greater extent. For example, the occurrence of an extra load on board will modify its weight, the positioning of the load centre and the rotating moment of the robot's platform. The walking robot can be influenced by a series of environment factors, the wind and other forces, their influence being hard to anticipate. Some of these interferences could be the cause of considerable variations of real motions in comparison to those estimated that could lead to robot control drift.

The instability problems appeared ever since real-time control systems were first used, during the '50s and the '60s, together with the first remote-controlled manipulators. For the force-position control, the analysis of the load and strategy generation remained for a long period out of the researchers preoccupations, until

* Romanian Academy, Institute of Solid Mechanics, E-mail: luigiv@arexim.ro
** "Politehnica" University of Bucharest

Rev. Roum. Sci. Techn. – Méc. Appl., Tome 55, Nº 2, P. 167–182, Bucarest, 2010

in the ‘70s the first operation control in force by computer assistance was tested. The new occurring problems about stability have been partially solved by using sophisticated leading algorithms, more rapid computation and flexible force cell, while the issue of the automatic generation of compliant leading strategies still awaits a full solution.

Later on there have been developed many more leading methods using force information: in damping leading method: Whitney, 1977; active compliance: Salisbury, 1980 and Craig, 1982; passive compliance (Watson, 1976, Drake and Simunovic, 1977), in impedance leading method (energetic methods) (Hanafusa and Asada, 1977, Hogan, 1980), in force implicit leading method (Borell, 1979). The subsequent studies on in force leading robots focused on the tests on automatic generation of leading strategies, Mason, 1981, sequel to the papers of Whitney and elaborates a general theory on force and movement constraints. Whitney and Edstall, in 1984, apply the signal processing and the estimation theory on data chains regarding the forces (couplings), following the analysis and high signal filtration, as well as a greater understanding of the robot’s behaviour and the evolution of meeting the tasks. On the compliant leading stability, Stepien, in 1985, analyzed the group robot-compliant load and gives a path for projecting the regulator needed to assure the system’s stability, at the same time, Roberts, Paul and Hillberry, study the effect of interposing of the elasticity inside the force transducers structure and of the robot on the in force leading performances.

The compliance in walking robots is necessary in order to avoid forces of greater impact, to rectify any positioning errors of robots and to permit the relaxation of the margin of the components. The compliance can be delivered either through passive compliance, such as Remote Centre of Compliance (RCC), either using active methods in controlling the force. In any case, there are some fundamental issues on both techniques, when they are implemented on correct applications. The passive compliance can tire the positioning ability of the robot, while the active compliance can cause instability in a rigid medium. Therefore, although there have been recently reported many on-going investigations with the same research purpose, still simpler, more economic and trusty methods are being sought.

The basic walking robots are considered as a group of articulated rigid solid bodies that represent the platform and the elements of the legs. The more the number of legs of a walking robot increases, the more the driving and command system becomes more intricate. On the other hand, due to the large number of supporting points, the static and quasi-static movement stabilizes itself even more. The movement of the quadruped robot is stable only when certain conditions are met, that are quite restrictive. The issue of static stability is resolved by calculating the position of the end of each leg in proportion to the axes system attached to the platform, originating in its centroid. As for the construction of the mathematic model, based on the quasi-dynamic analysis, each leg is considered by the authors as a function generator, with a limited precision in constructing the walking systems. If the number of mobility range is equal to n and if the internal constraints have the form of:

$$F_j(x_1, x_2, ..., x_n) = 0,\ j = \overline{1,\ m},\ n \geq m, \tag{1}$$

then within the differential equations structure:

$$\frac{\mathrm{d}x_j}{\mathrm{d}t} = f_i(x_1, x_2, ..., x_n,\ u_1,\ u_2, ..., u_s, t),\ i = \overline{1,\ n}, \tag{2}$$

exist random coefficients u_i – used for identifying the walking algorithm. To those differential equations apply, in the first place, the constraints imposed by the general base (the platform), onto which the legs are fixed, and in the second place, the constraints assessed by the bearing surface.

2. KINEMATICS MODELLING OF THE LEGS' MECHANISMS

The converse kinematic analysis of the walking system gear of the robot is monitored through variables of active couples and their derivables and reported to time variable, in accordance to position point P_i of the object (3) as part of the leg structure of the walking robot that touches the ground, meaning the bearing phase, and respectively its acceleration. On an uneven ground, the movement of the walking robot must have at least three mobility ranges. In this way, the end point P_i in the leg ($i = \overline{1,\ 4}$ for the quadruped robot and $i = \overline{1,\ 6}$ for the six-legged robot, can move in a given space, that forms its working space, on a certain trajectory, with an imposed movement law. Dependent of them, one can determine the independent kinematic parameters in the three conducting couples inside the leg mechanism, through the conversed kinematic analysis. During the analysis, to each element of the mechanism is attached a system of coordinate axis, according to Hartenberg-Denavit formalism [8].

The position of the bearing point P is determined taking into account the Hartenberg-Denavit system. $O_4X_4Y_4Z_4$ attached to the element (3) of the leg mechanism, and is an invariant (Fig. 1). The linear equation system derived is calculated depending on the unknown variables $\theta_1, \theta_2, \theta_3$, using the Newton-Raphson method.

2.1. KINEMATIC MODELLING OF THE BEARING POINTS

Direct and inversed kinematic modelling of the bearing points P_i of the legs depend on the geometric centre O of the walking robot, which can be seen in the kinematic Fig. 2 and reported to Fig. 1. As it can be seen, it has 3 modules each two-legged connected to 3 motor rotating couples. Each leg consists of 3 elements connected to 3 rotating couples. To determine the bearing position of the legs related to kinematic couple, they are directly connected to the platform using the homogeneous coordinates matrix method with Denevit-Hartenberg rotations. Direct kinematic analysis means to determine the end coordinates in the leg compared to the referential system chosen by the robot when the parameters are known θ_i^j, $i = 4, 7, j = 1, 6$. θ_0 is the constructive constant and θ_i, $i = 1, 3$.

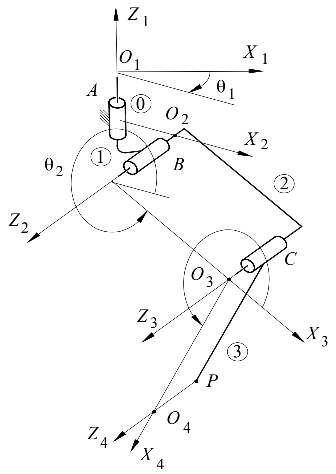

Fig. 1 – Kinematics modelling of the leg's mechanisms.

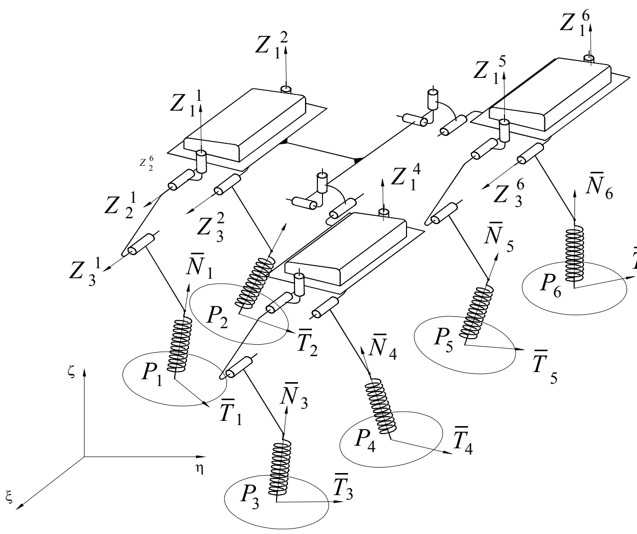

Fig. 2 – Mathematic modelling of the gravity centre position.

Using the conversed kinematic analysis, knowing the parameters of the bearing points $X_O{}^i{}_P$, $Y_O{}^i{}_P$, $Z_O{}^i{}_P$, there has been determined the mathematic model with variable parameters in the mechanism of the couples in the leg $\theta\,{}_i^j$ where $j = 1, 6$, $i = 4, 6$. Thus, the trajectories described by the legs being known, it was possible to determine the relevant parameters of the control system in real time control of the robot. The direct kinematic analysis calculates the $X_O{}^i{}_P$, $Y_O{}^i{}_P$, $Z_o{}^i{}_P$, while the conversed kinematic analysis permits the processing of $\theta_4{}^i$, $\theta_5{}^i$, $\theta_6{}^i$. The relative speed and acceleration are calculated by derivation relative to the equations mentioned above. The general case of walking robots with 2^n legs, with n connected modules, miriapode robots, follow the same algorithm as with the kinematic of the bearing points in the leg mechanism in respect to the referential system established.

2.2. MATHEMATIC MODELLING OF THE GRAVITY CENTRE POSITION

Mathematic modelling of the position of the centre of gravity, which allows the walking robots to shift on ground with difficult configuration, is presented in Fig. 2. The geometric centre position O is defined as the diagonals' intersection point of the polygon formed by the points where the legs are connected to the platform, and $G_{(GxGyG,zG)}$ as the gravity centre position of robot.

Considering the positions of the walking robot's legs as X_{Pi}, Y_{Pi}, Z_{Pi} a mathematic model has been developed in order to express the kinematics characteristics of the walking robot's centre of gravity. In order to determine the position of the supporting polygon against the platform, through the Denevit-Hartenberg method, for Z_{ij} ($i = 1, 6$ or $1, 4$, $j = 1, 3$), and m_{ij} ($i = 1, 6$, $j = 1, 3$) the leg mechanism mass, the coordinates of P_i have been transformed from the system $O_{4x4y4z4}$ into the system O_{xOyOzO}.

The coordinates of P_i have been transformed from the system $O_{4x4y4z4}$ into the system O_{xOyOzO} in order to determine the position of the support polygon against the platform. The vertical projection of the G system centre of gravity on the supporting surface must be inside the supporting polygon in order to obtain stability. $A_O{}^i$, $A_1{}^i$, $A_2{}^i$, $A_3{}^i$ are given where i = 1–6 for the hexapod walking robot and i = 1–4 for the quadruped walking robot. Knowing the position of the centre of gravity, the speed $\dot{X}_G^k$ and the acceleration $\ddot{X}_G^k$ have been established by derivation.

3. ROBOT POSITION-FORCE HYBRID CONTROL

Theoretical and experimental studies have demonstrated that, depending on the leading system architecture, force response, implemented either as an analogical adjusting loop, or as a numeric calculus algorithm, can introduce, as an adjusting loop, an instability inside the system. The constructive measures, that favours the stability of the driving robot, approach two solutions (West and Asada, 1985): using reduced length kinematic elements, with large band-pass and the interposition of a compliant device into the robot's leg closure.

Hybrid position-force control of industrial robots equipped with compliant joints must take into consideration the passive compliance within the system [1, 7]. The generalized surface on which the robot labours must be defined into a constraint space with 18 degrees of freedom (DOF), with position constraints along the normal to this surface and force constraints along the tangents. Out of simplification considerations the coordinate transformations are not noted. The variables X_C and F_C represent the Cartesian position, respectively the Cartesian force exerted upon the environment.

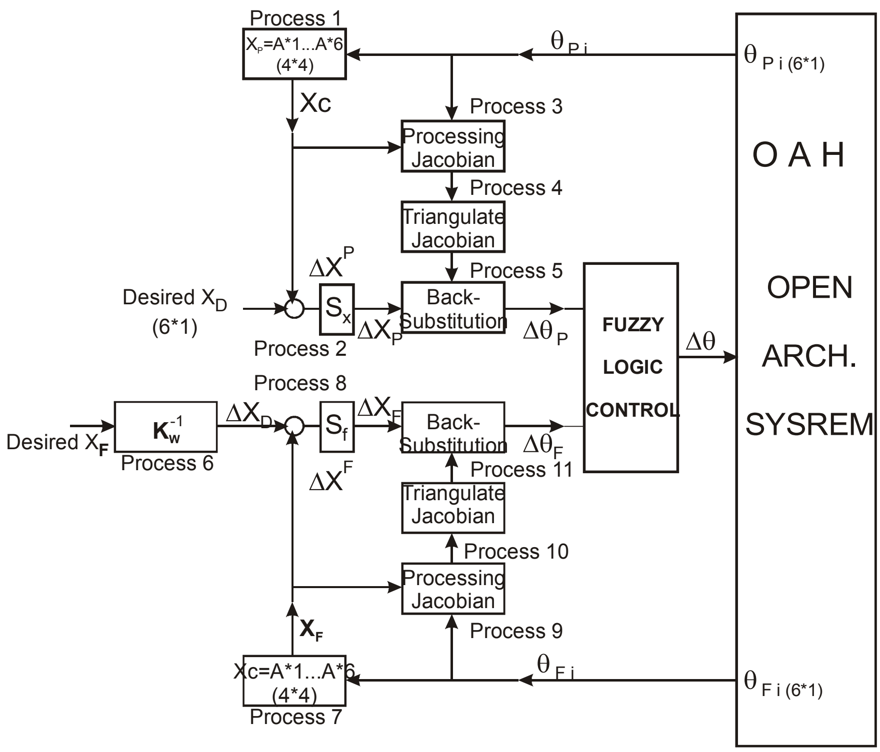

Fig. 3 – Hybrid position-force control based on the Denevit-Hartenberg model.

Considering X_C and F_C as expressed in the environment-specific coordinates, the selection matrixes $\mathbf{S}_x$ and $\mathbf{S}_f$ can be determined, which are diagonal matrixes with 0 and 1 as diagonal elements, and fulfil the relation: $\mathbf{S}_x + \mathbf{S}_f = \mathbf{I}_d$. In more recent approaches $\mathbf{S}_x$ and $\mathbf{S}_f$ are methodically deduced from the kinematical constraints imposed by the working environment. The final results lead to the following conclusion: with the aid of hybrid control, the robot's gripper acts as a rigid body, without mass, submitted to an external force f_{des} and connected through an ideal kinematical constraint to another body, whose velocity is v_{des}.

3.1. THE MATHEMATICAL EQUATIONS FOR HYBRID POSITION-FORCE CONTROL

If only the position control is considered on the directions established by the selection matrix $\mathbf{S}_x$, both the desired differential motions of the end-effector corresponding to control in position can be determined from the relation: $\Delta\mathbf{X}_\mathbf{P} = \mathbf{K}_\mathrm{P}\Delta\mathbf{X}^\mathrm{P}$, where $\mathbf{K}_\mathrm{P}$ is the gain matrix, as well as the desired joint angles on the axis controlled in position:

$$\Delta\theta_\mathrm{P} = \mathbf{J}^{-1}(\theta) \times \Delta\mathbf{X}_\mathrm{P}. \tag{3}$$

In continuing, also taking into consideration the force control on the other directions left, the relation between the desired angular motion of the end-effector and the force error $\Delta\mathbf{X}_F$ is given by the relation:

$$\Delta\theta_\mathrm{F} = \mathbf{J}^{-1}(\theta) \times \Delta\mathbf{X}_\mathrm{F}, \tag{4}$$

where the positioning error due to force $\Delta\mathbf{X}_\mathrm{F}$ is the motion difference between $\Delta\mathbf{X}^\mathrm{F}$ – the current position deviation measured by the command system which generates the position deviation for the axis controlled in force and $\Delta\mathbf{X}_\mathrm{D}$ – the position deviation due to the desired residual force.

Finally, there results the motion variation on the robot axis in relation to the end-effector motion variation from the relation:

$$\Delta\theta = \mathbf{J}^{-1}(\theta)\Delta\mathbf{X}_\mathrm{F} + \mathbf{J}^{-1}(\theta)\Delta\mathbf{X}_\mathrm{P}. \tag{5}$$

The architecture of the hybrid position-force control system of robots with six degrees of freedom based on the Denevit-Hartenberg transformations is presented in Fig. 3. The device sensors are used in two ways. In position control, the information obtained from the sensors is used to compensate the deviation of the robots' joints, due to the load created by external forces, so that the apparent stiffness of the robot's joint system is emphasised. In force control, the joint is used as a force sensor, so that the manipulator is led in the same direction as the force received from the sensors, allowing the desired contact force to be maintained. In the following, these two control modes are executed concurrently.

3.2. COMPLIANT CONTROL METHODS WITH TRACKING FUNCTIONS ON MODULAR WALKING ROBOTS

Modular robots used for transportation activities on rough ground need additional compliant functions in order to correct the positioning error, to create a tolerance relaxation among the components and to absorb the impact forces with the elements of the robot [4, 5]. The automation system with open architecture will permit an adaptation of the compliant. The tracking process on one axis of the robot having the force control on the other two axes is represented in Fig. 4. Monitoring is done by the movement along the robot's leg, maintaining the contact in the other two directions. As seen in Fig. 4, the contact forces are controlled on *Z* and *X* directions, while *Y* directions is position controlled. The monitoring process on the movement of the walking robot can be divided in three phases: approaching, searching and platform movement.

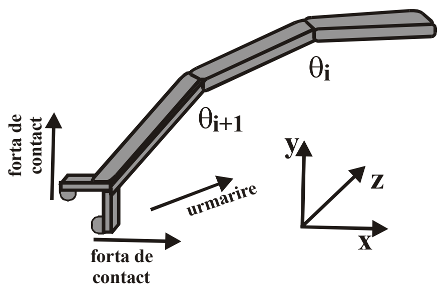

Fig. 4 – The tracking process on one robot axis with the force control.

The approach phase: the robot approach the edge with a specific v_a speed with position control on all the axes, according to a position control process. When the feeler sensor makes contact with the bearing surface, a contact force is detected. Through this force and with the help of the force translator, the movement in that direction is being stopped, and the controlling system will secure: a) the force control on that direction and b) position control in all directions.

Searching phase: at the end of the approaching phase, one arm of the feeler makes contact with the bearing surface, while the other arm is near the bearing surface. In this situation, corresponding to the searching phase, the command system will generate with the purpose of approach, a trajectory of vertical movement cu specified searching speed v_s. When even this segment of the feeler is in contact with the bearing surface, the searching phase ends, and the force control is assigned towards the walking direction of the robot.

Platform movement phase: when the other two phases end, the bearing surface is in contact to the feeler of the leg. From this moment starts the platform movement phase with controlled force on the normal directions of two surfaces, and the position is controlled in the direction of the movement of the robot platform.

Implementing the method into the automation system with open architecture might realize in real time without hard modifications of the command system. To this reason is being introduced a new task in the structure of the OAH module

(PC–Open Architecture) in Fig. 8. The role of this task is to generate the positions and the forces in accordance to the presented tracking algorithm. To realize the application, we need to take into account that the contact force must be selected in conformity with the load, and the physical rigidity of the compliant device is different o each contact direction, which determines diverse deviations from and around every axis.

Developing a new multi-stage (MS) fuzzy control method. A new control method, which practically eliminates jamming and has a fast response of the control loop, is presented in Fig. 5. It consists of a multi-stage fuzzy control (MS), which entails the realization of two fuzzy control loops, one in position and the other in force on two different decision stages in order to determine the speed of the feeler. MS fuzzy control has multiple rule bases, where the result of an inference of the rule base is transmitted to the next stage. In this way the most important dimensions of the inference can be grouped into smaller sets and combined with base rules. In the MS structure, the results of the rule base of the control of position P are transmitted to the rule base of the PF position-force control.

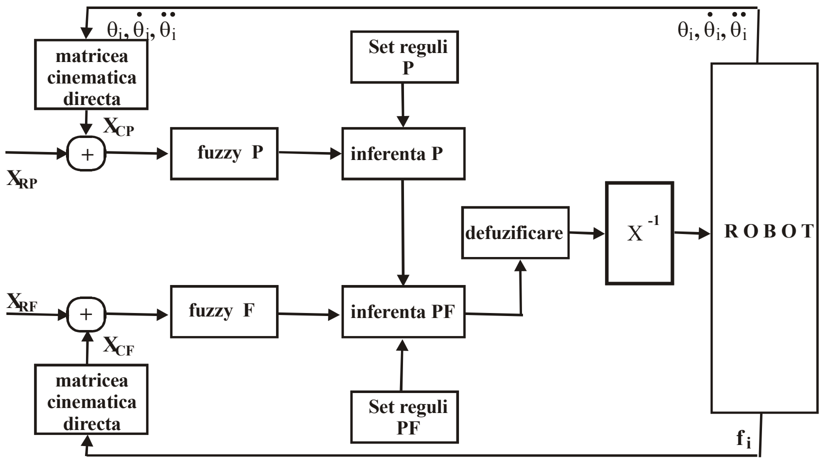

Fig. 5 – Fuzzy multi-stage control (FMS) for tracking processes.

This structure is similar to a hierarchical structure of the regulators, based on characteristics [2, 6]. In terms of control, based on the characteristics of the positioning functions, P is of high level, it ensures system control disturbance dynamics and it should appear if the commands are generated and the main function is forced and generally controls the system to avoid jamming. The control returns the functions base P when the system dynamic is settled. The basic idea of the controller is to assign the speed on each axis for the given deflection in the corresponding direction in a heuristic way, in which a human operator might accomplish the insertion. The task of the controller is to assign the measured deflection of the fuzzy variables, *i.e.* positive great (PM), and to evaluate the decision rules through inference. So that, it can finally establish the value of the output variable, *i.e.* the velocity as fuzzy variable, which best follows the controlled parameter. There are taken into consideration the deviation in position of the compliant joints *e,* the rate of position deflection Δe and the contact force Δf as entry data. The values of deflection detected through sensors are quantified in a points number corresponding to the discourse universe elements, and then the values are assigned as grades of membership in a few fuzzy subsets. The relationship between the inputs, *i.e.* the measured deflections, and outputs, *i.e.* velocity, and the grades of membership can be defined in conformity to the operator's experiences and the demands of the robot charge. There are defined empirically the membership functions for all input and output elements. The fuzzy variables have been chosen as follows: NM – negative great, N_M – negative medium, Nm – negative small, ZO – zero, Pm – positive small, P_M – positive medium, PM – positive great. By analyzing the rule base, it can be observed that the force feedback is function of the inference results of the fuzzy control P component.

The rule base P is easily modified from a typical linear rule base, allowing for the replacement of all Zero (ZO) values, except the centre of the rule base. In this fashion, the P rule base will pass onto value ZO only when both the error, as well as changes in the error terms are ZO, which is to indicate that the system has been settled. For a certain set of inputs, *i.e.* the measured deflection, the evaluation of fuzzy rules produces a fuzzy membership set for system control. In order to take a concrete action, one of these values must be chosen. In this application, the control value with the highest degree of membership has been selected.

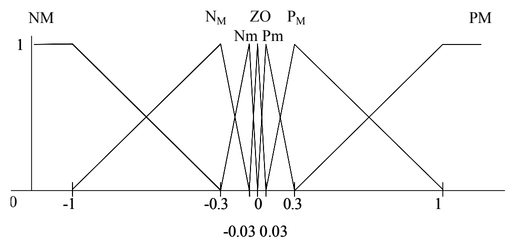

Fig. 6 – Error input fuzzy sets.

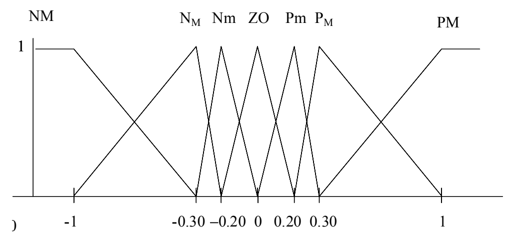

Fig. 7 – Rate error input fuzzy sets.

The rules are evaluated at equal intervals, in the same way as a conventional control system. Figures 6–7 present the membership function sets for inputs. The result of logical inference also represents fuzzy values which are applied to the defuzzify mode. The defuzzification represents a transformation of the fuzzy variables defined on the output discourse universe in a numerical value. This processing is necessary due to the fact that the control in the case of fuzzy regulators is done exclusively with encrypted values. Choosing as the defuzzification method the weight centre of aria, the calculation of the defuzzification output is given. Furthermore, they get a rapid response of the control loop and the instability is practically eliminated.

3.3. MULTIPROCESSOR SYSTEM WITH OPEN ARCHITECTURE FOR FORCE-POSITION HYBRID CONTROL OF MODULAR ROBOTS HFPC MERO

On the basis of fundamental and applicative research in mathematic modelling and modular walking robots control MERO, is conceived a control system in real time with open architecture (OAH-Open Architecture) (Fig. 5). The system has four main components: the programmable logical controller (PLC0), PC system with open architecture (PC-OPEN), multiprocessor system PLC (SM-PLC) and the position control system ($\mathbf{CP}^k_i$, i = 1–3, k = 1–6). The system ensures the implementation of the algorithm for robot position control in Cartesian coordinates through real time processing of the Jacobean matrix obtained by direct kinematics of the robot, respectively of the invert Jacobean matrix for feedback. The method consists in reducing the matrix $\mathbf{J}(\theta)$ to an upper triangulate form and finding errors in $\delta\theta$ joint coordinates using back-substitution [3, 4]. The joint angle errors $\delta\theta$ can be used directly as control signals for robot actuators. Joint angular errors for actuators on each robot degrees of freedom axis corresponding to the robot's new desired position generated in Cartesian coordinates are obtained by processing of Jacobean and inverted substitution. Below you can find the presentation of the main components of the real time control system with open architecture for the MERO modular walking robot (Fig. 8).

The programmable logical controller in decentralized and distributed structure (PLC0) ensures the control of the degrees of freedom axis for the robot execution elements. The current angle motion values in absolute value $\boldsymbol{\Sigma\theta_{ci}}$ are transmitted from PLC0 to PC-OPEN. From the PC to the PLC are continuously transmitted the reference positions on each axis X_{Di} according to the trajectory generating program X_{Pi}. The PLC0 is processing 24 analogue inputs, 24 digital output, 18 analogue outputs for robot position and stability control, and other 64 digital inputs/outputs configurable for auxiliary functions: the hydraulic group control, motion limitation devices, limitation devices for homing etc. Moreover, the following analogical and digital input signals are controlled: position transducers ($\mathbf{TP}^k_i$, i = 1–3, k = 1–6), proximity transducers, $2\times D^k$ horizontal and $2\times D^k$ on vertical for orientation, force transducers ($\mathbf{T}^k{}_F$, k = 1–6), vertical control transducers (X_M, Y_m) for robot stability.

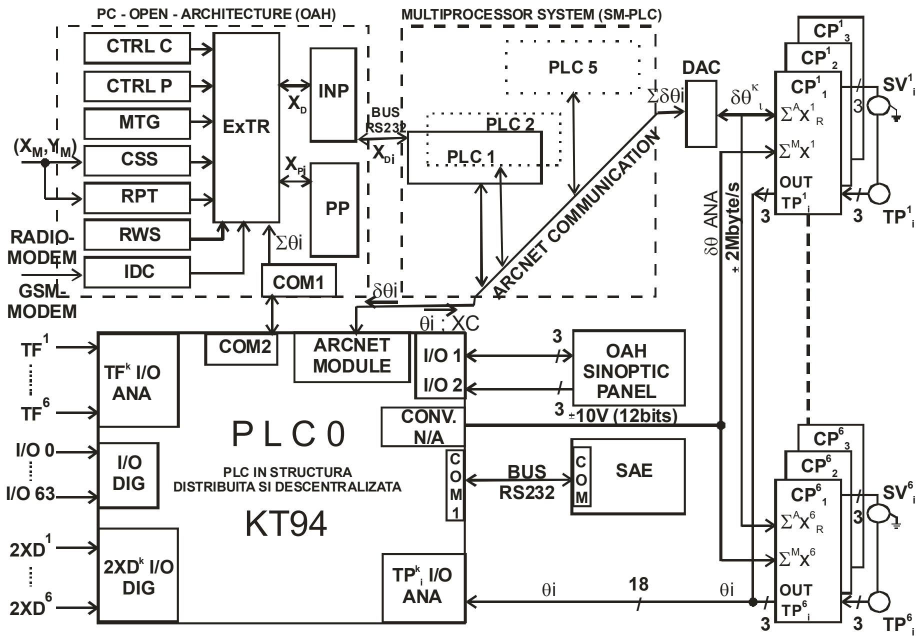

Fig. 8 – Real time control system with open architecture for the MERO modular walking robot.

The automat provides a maximum of 62 digital inputs and another 80 digital inputs/outputs configurations, controlling the following analogical and digital input signal:

– increment position detectors, that measure the variables of the couples, related to the relative positions of the kinematic elements in the conducting couples;
– sensors ($2\times D^k$ for the horizontal and $2\times D^k$ on vertical), placed at the legs ends, and signalling the presence of the obstacles on the bearing surface;
– force cells ($T^1{}_F$,…., $T^6{}_F$), determine the resolution of 12 bits, the reaction forces in the ground and the legs, and the points in their ends;
– vertical detectors (hold) X_M, Y_m to determine de stability of the robot and the horizontal position of the platform.

Position Controllers ($\mathbf{CP}^k{}_i$, i = 1–3, k = 1–6) have 18 hydraulic servo-valves ($\mathbf{SV}^k{}_i$, k = 1–6, i = 1–3) as actuators, providing movement control on each free axis, based upon angular deviations $\delta\theta^K{}_i$.

The system PC-OAH (PC-OPEN Architecture) allows the introduction of new control functions based on supplementary programs. Due to the great processing speed with operating systems, which allow for programming in evolved programs, the basic functions can be implemented, using an ExTR real time multitask executive: interpolation (INP), main program (PP) of the operator interface, the movement trajectory generating program (MTG), the static stability control (SSC), robot platform control (RPT), the robot's walking shaping program (RWS), compliant function control program (CTRL C) and the prehensive control program (CTRL P). Supplementary, an interface with digital camera (IDC) could be installed, for objects recognition and the communication interface using radio modem, GSM or wireless systems. The PC-OAH system performs the following operations: shapes walking stereotypes, determines the walking stability limits, ensures the horizontal position of the platform, maintains the established height of the platform and interprets the operator orders.

The Interpolator modulus (ITP) provides the generating of intermediary points between two references of position. The programming is made in BORLAND C++ and allows the processing of over 200 new references in an interval of 100 μs.

The Main Program (PP) provides a graphic interface between operator and robot. The robot has a menu allowing him to: choose the walking program, the 3D linear and circular interpolating simulation program, generating alarms for over load parameters, trajectory generating simulation, step by step or automatic functioning. The program allows the graphical simulation of the chosen walking type.

The Multiprocessor System PLC (SM-PLC) and PLC0 has the role to send in real time, through the ARCNET fast communication network, the angular reference positions for the PIDT position regulator. For feedback, the current values θ^{K}_{ci}, (i = 1–3, k = 1–6), received from the position transducers ($\mathbf{TP}^{k}_{i}$, i = 1–3, k = 1–6) are transmitted by PLC0 through ARCNET. Five processes for implementing the mathematical model of robot control have been identified. In the active topology for (1) process, each PLC generates an ascendant data flux from PLC0 to PLC5. By calculating the transformed matrix ${}^{i}\mathbf{A}_{j}$ for the leg k (k = 1–3), from the axis i to the axis j, we obtain the coordination matrix in axis j, and finally the resulting coordinates for the robot environment $\mathbf{X}^{k}_{C} = {}^{1}\mathbf{A}^{k}_{3} = \mathbf{A}_{1}\times\mathbf{A}_{2}\times\mathbf{A}_{3}$. In the active topology for the process (2), the matrix ${}^{i-1}\mathbf{A}^{k}_{i}$ is stored for each PLC, the Cartesian coordinates $\mathbf{X}^{k}_{i}$ in i axis, by multiplying with ${}^{1}\mathbf{A}^{k}_{j-1}$, are determined and the position variation $\delta\mathbf{X}^{k}_{C}$ is calculated. The Jacobean matrix is obtained during the process (3) by an ascending data flow, correlated with process (1). During the matrix calculation ${}^{1}\mathbf{A}^{k}_{j}$, the PLC0 is assigned to $\mathbf{J}(\theta_1)$, matrix of (3×1), respectively PLC5 to $\mathbf{J}(\theta_3)$. The active topology of process (4) brings the Jacobean matrix to triangular form by determining the pivot element and the new matrix $\mathbf{A}^{k}_{ij}$ elements.

To each PLC there has been allotted a column of the Jacobean matrix, the data flux going from PLC0 to PLC5 for triangulation and from PLC5 to PLC0 for inverted substitution. With the help of the relations from the mathematical model the execution program for the PLC0-PLC5 has been conceived and executed, in which each central unit has the role of MASTER communication by data flux through the ARCNET network (Fig. 8). The active process (5) took place within a sequential process with process (4), determining the value of the angular error $\delta\theta^{\kappa}{}_{i}$ by inverted substitution.

The robot's walking types are issued from 3 block-programs located in PC-OAH system, namely: the block of walking shaping, which determine the succession and the way of moving the legs, the block of static stability control, granting the robot shift so that the system centre of gravity projection may remain inside the convex polygon formed by the leg's, the block of platform control, maintaining the prescribed height and the horizontal position of the platform.

4. CONCLUSION AND EXPERIMENTAL DATA

Creating a new autonomous methatronic system, endowed with objects handling functions, locomotion, perception, navigation, reasoning, information memory and intelligent control, set to accomplish missions of cluster sets of a dynamic universe, is the objective towards which, many research teams from the most prestigious universities all over the world, concentrate their efforts. In agricultural and forestry activities, due to specific conditions of vegetation, the properties of the land and the protection of the environment, in order to maintain and protect the ecological systems, the cars with wheels or roller-chain track used for this kind of activities, have a limited mobility and have a significant destructive effect on the environment, vegetation and young trees, when passing across them. The mobile methatronic systems protect the environment better, because the ground contact is delicate, therefore highly limiting the crushed surface. The weight of the mobile methatronic system can be best distributed on the bearing surfaces through a force control. The variation of the distance against the ground allows the robot to pass over the young trees and other vegetation growing in the transit area. Avoiding these obstacles, such as logs and trunks, represents a great advantage.

In the field of services, it is very attractive when using the mobile methatronic systems, due to the increment of sensors and their performance. These systems can be used for medical assistance in ambulatory systems, assistance for people with disabilities, assistance for educational programs, etc.

The MERO walking robot's control system in real time with Open Architecture (OAH) ensures flexibility, short time execution, the precision targets and repeatability of the moving programs, eliminating completely the closed systems with projects meant for specified applications. Supplementary developments in order to increase the performances or new functions adding are possible only by modifying the software relating to the control modules in PC-OAH for laborious computations, respectively in the PLC multiprocessor for complex real time control. Besides, addition of new physical modules could easily be done because of the communication between the programmable logical controller (PLC) and the input/output modules made by a bus with 3 conductors. For other exemplifications (Figs. 9–10), the simulations of the walking robots functioning in two different activities: obstacle by-pass, respectively, radioactive load transportation.

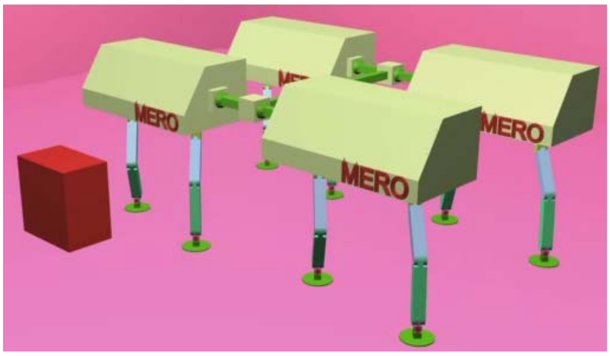

Fig. 9 – Obstacle by-pass with force sensors.

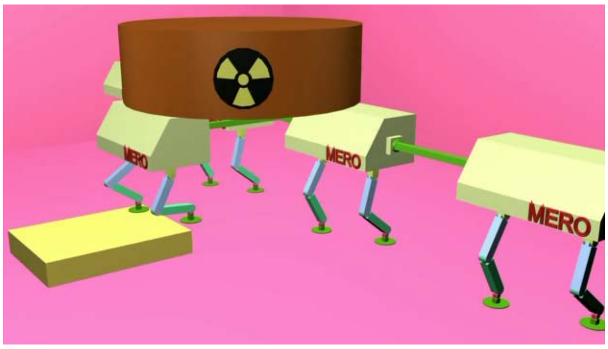

Fig. 10 – Radioactive load transportation in uneven environments.

Owing to the great computation speed of microprocessor systems and serial connection links for data transmission, the time necessary for establishing the inverted matrix is short enough to allow the robot control in real time, with no influence in performing the other programs. It has been estimated a considerable reduction of the operation period of the walking robots control program MERO. Moreover, the short time execution will ensure a faster feedback, allowing other programs to be performed in real time as well, like the prehensive force control, objects recognition, making it possible that PC-OHA system have a human flexible and friendly interface.

ACKNOWLEDGEMENTS. The authors wish to express their gratitude to the Research and Educational Ministry (MEC) and to the Romanian Academy for their support of the program of work reported herein. This work was supported by CNCSIS –UEFISCSU, project number PNII – IDEI 005/2007.

REFERENCES

1. Canudas de Wit C, B. Siciliano, G. Bastin, *Teory of robot control*, Springer-Verlag, London, 1996.
2. J. Yen, N. Pfluger, *A Fuzzy Logic Based Extension to Payton and Rosenblatt´s Command Fusion Method for Mobile Robot Navigation*, IEEE Transactions on Systems, Man, and Cybernetics, **25**, *6*, pp. 971-978, 1995.
3. Vladareanu L., Curaj A., Ion I., Velea L.M., Munteanu M., Mitroi D., Gal A.,Vasile A., *Fundamentale and Aplicative Researche for Hybride Force-Position Control of Modular Walking Robots* (phase 2/2008 and 3/2009), Research Project no. 263/2007-2010 in the framework of Grants of CNCSIS Program IDEAS, PN II (National Program for Scientific Research and Innovation Technologies).
4. Vlădăreanu L., *Open Architecture Systems for the Compliance Robots Control*, WSEAS Transactions on Systems, **5**, *9*, 2006, pp. 2243-2249.
5. Wu C-J, Ching-Huo H, *Parameter Identification of Direct Drive Robots*, Journal of Intelligent and robotic Systems, **16**, pp. 45-64, 1996.
6. Vlădăreanu L., Tont G., Ion I., Vlădăreanu V., Mitroi D., *Modeling and Hybrid Position-Force Control of Walking Modular Robots*, ISI Proceedings: *Recent Advances in Applied Mathematics*, Harvard University, Cambridge, USA, 2010, pp. 510-518.
7. Fisher W.D., Mujtaba M.S., *Hybrid Position/Force Control: A Correct Formulation*, The International Journal of Robotics Research, **11**, *4*, August 1992, pp. 299-311.
8. Denavit J., Hartenberg R.S., *A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices*, J. Appl. Mech., June 1955, pp. 215-221.