THE NAVIGATION MOBILE ROBOT SYSTEMS USING BAYESIAN APPROACH THROUGH THE VIRTUAL PROJECTION METHOD

LUIGE VLADAREANU, GABRIELA TONT, VICTOR VLADAREANU, FLORENTIN SMARANDACHE, LUCIAN CAPITANU

***Abstract*. The paper presents the navigation mobile walking robot systems for movement in non-stationary and non-structured environments, using a Bayesian approach of Simultaneous Localization and Mapping (SLAM) for avoiding obstacles and dynamical stability control for motion on rough terrain. By processing inertial information of force, torque, tilting and wireless sensor networks (WSN) an intelligent high level algorithm is implementing using the virtual projection method. The control system architecture for the dynamic robot walking is presented in correlation with a stochastic model of assessing system probability of unidirectional or bidirectional transition states, applying the non-homogeneous/non-stationary Markov chains. The rationality and validity of the proposed model are demonstrated via an example of quantitative assessment of states probabilities of an autonomous robot. The results show that the proposed new navigation strategy of the mobile robot using Bayesian approach walking robot control systems for going around obstacles has increased the robot's mobility and stability in workspace.**

I. INTRODUCTION

Walking robots, unlike other types of robots such as those with wheels or tracks, use similar devices for moving on the field like human or animal feet. A desirable characteristic a mobile robot must have the skills needed to recognize the landmarks and objects that surround it, and to be able to localize itself relative to its workspace. This knowledge is crucial for the successful completion of intelligent navigation tasks. But, for such interaction to take place, a model or description of the environment needs to be specified beforehand. If a global description or measurement of the elements present in the environment is available, the problem consists on the interpretation and matching of sensor readings to such previously stored object models. Moreover, if we know that the recognized objects are fixed and persist in the scene, they can be regarded as landmarks, and can be used as reference points for self localization. If on the other hand, a global description or measurement of the elements in the environment is not available, at least the descriptors and methods that will be used for the autonomous building of one are required [1].

The approach of the localization and navigation problems of a mobile robot which uses a WSN which comprises of a large number of distributed nodes with low-cost cameras as main sensor, have the main advantage of require no collaboration from the object being tracked.

The main advantages of using WSN multi-camera localization and tracking are:

1) the exploit of the distributed sensing capabilities of the WSN;

2) the benefit from the parallel computing capabilities of the distributed nodes. Even though each node have finite battery lifetime by cooperating with each other, they can perform tasks that are difficult to handle by traditional centralized sensing system.;

3) the employ of the communication infrastructure of the WSN to overcome multi-camera network issues. Also, camera-based WSN have easier deployment and higher re-configurability than traditional camera networks making them particularly interesting in applications such as security and search and rescue, where pre-existing infrastructure might be damaged [2].

Robots have to know where in the map they are in order to perform any task involving navigation. Probabilistic algorithms have proved very successful in many robotic environments. They calculate the probability of each possible position given some sensor readings and movement data provided by the robot [5]. The localization of a mobile robot is made using a particle filter that updates the belief of localization which, and estimates the maximal posterior probability density for localization. The causal and contextual relations of the sensing results and global localization in a Bayesian network, and a sensor planning approach based on Bayesian network inference to solve the dynamic environment is presented. In the study is proposed a mobile robot sensor planning approach based on a top-down decision tree algorithm. Since the system has to compute the utility values of all possible sensor selections in every planning step, the planning process is very complex.

The paper first presents the position force control and dynamic control using ZMP and inertial information with the aim of improving robot stability for movement in non-structured environments. The next chapter presents the mobile walking robot control system architecture for movement in non-stationary environments by applying

Wireless Sensor Networks (WSN) methods. Finally, there are presented the results obtained in implementing the interface for sensor networks used to avoid obstacles and in improving the performance of dynamic stability control for motion on rough terrain, through a Bayesian approach of Simultaneous Localization and Mapping (SLAM).

II. Dynamical Stability Control

The research evidences that stable gaits can be achieved by employing simple control approaches which take advantage of the dynamics of compliant systems. This allows a decentralization of the control system, through which a central command establishes the general movement trajectory and local control laws presented in the paper solve the motion stability problems, such as: damping control, ZMP compensation control, landing orientation control, gait timing control, walking pattern control, predictable motion control (see ICAMechS 2011, Zhengzhou [3]).

In order to carry out new capabilities for walking robots, such as walking down the slope, going by overcoming or avoiding obstacles, it is necessary to develop high-level intelligent algorithms, because the mechanism of walking robots stepping on a road with bumps is a complicated process to understand, being a repetitive process of tilting or unstable movements that can lead to the overthrow of the robot. The chosen method that adapts well to walking robots is the ZMP (Zero Moment Point) method. A new strategy is developed for the dynamic control for walking robot stepping using ZMP and inertial information. This, includes pattern generation of compliant walking, real-time ZMP compensation in one phase - support phase, the leg joint damping control, stable stepping control and stepping position control based on angular velocity of the platform. In this way, the walking robot is able to adapt on uneven ground, through real time control, without losing its stability during walking [13].

Based on studies and analysis, the compliant control system architecture was completed with tracking functions for HFPC walking robots, which through the implementation of many control loops in different phase of the walking robot, led to the development of new technological capabilities, to adapt the robot walking on sloping land, with obstacles and bumps. In this sense, a new control algorithm has been studied and analyzed for dynamic walking of robots based on sensory tools such as force / torque and inertial sensors [3,13]. Distributed control system architecture was integrated into the HFPC architecture so that it can be controlled with high efficiency and high performance.

III. Simultaneous Localization and Maping

A precise position error compensation and low-cost relative localization method is studied in [5] for structured environments using magnetic landmarks and hall sensors [6]. The proposed methodology can solve the problem of fine localization as well as global localization by tacking landmarks or by utilizing various patterns of magnetic landmark arrangement. The research in localization and tracking methods using Wireless Sensor Networks (WSN have been developed based on Radio Signal Strength Intensity (RSSI) [7] and ultrasound time of flight (TOF) [8]. Localization based on Radio Frequency Identification (RFID) systems have been used in fields such as logistics and transportation [9] but the constraints in terms of range between transmitter and reader limits its potential applications. Many efforts have been devoted to the development of cooperative perception strategies exploiting the complementarities among distributed static cameras at ground locations [10], among cameras mounted on mobile robotic platforms [11], and among static cameras and cameras onboard mobile robots [12]. Computation-based closed-loop controllers put most of the decision burden on the planning task. In hazardous and populated environments mobile robots utilize motion planning which relies on accurate, static models of the environments, and therefore they often fail their mission if humans or other unpredictable obstacles block their path. Autonomous mobile robots systems that can perceive their environments, react to unforeseen circumstances, and plan dynamically in order to achieve their mission have the objective of the motion planning and control problem [4, 9].

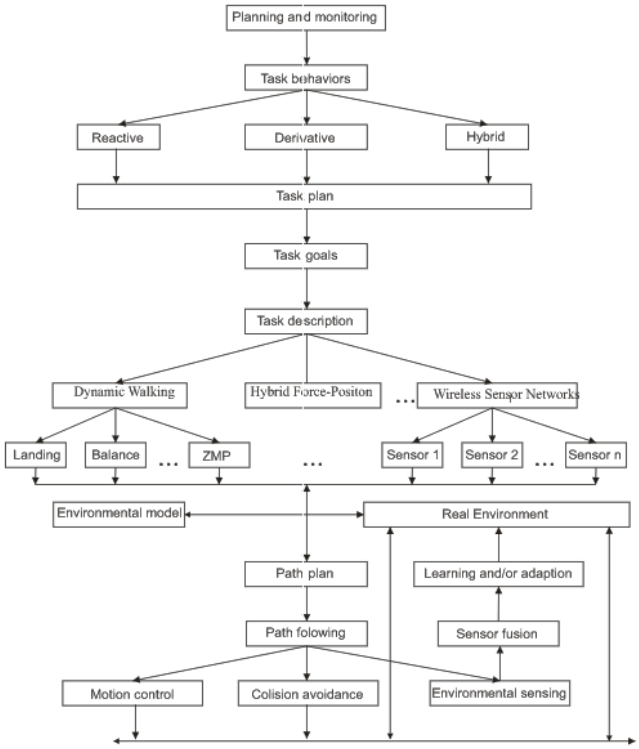

Figure 1 Mobile robot control system architecture

To find collision-free trajectories, in static or dynamic environments containing some obstacles, between a start and a goal configuration, the navigation of a mobile robot comprises localization, motion control, motion planning and collision avoidance. Its task is also the online real-time re-planning of trajectories in the case of obstacles blocking the pre-planned path or another unexpected event occurring. Inherent in any navigation scheme is the desire to reach a

destination without getting lost or crashing into anything. The responsibility for making this decision is shared by the process that creates the knowledge representation and the process that constructs a plan of action based on this knowledge representation. The choice of which representation is used and what knowledge is stored helps to decide the division of this responsibility. Very complex reasoning may be required to condense all of the available information into this single measure [4, 14]. The techniques include computation-based closed-loop control, cost-based search strategies, finite state machines, and rule-based systems [17].

Computation-based closed-loop controllers put most of the decision burden on the planning task. In hazardous and populated environments mobile robots utilize motion planning which relies on accurate, static models of the environments, and therefore they often fail their mission if humans or other unpredictable obstacles block their path. Autonomous mobile robots systems that can perceive their environments, react to unforeseen circumstances, and plan dynamically in order to achieve their mission have the objective of the motion planning and control problem. To find collision-free trajectories, in static or dynamic environments containing some obstacles, between a start and a goal configuration, the navigation of a mobile robot comprises localization, motion control, motion planning and collision avoidance [15, 16]. A higher-level process, a task planner, specifies the destination and any constraints on the course, such as time. Most mobile robot algorithms abort, when they encounter situations that make the navigation difficult. Set simply, the navigation problem is to find a path from start (S) to goal (G) and traverse it without collision. The relationship between the subtasks mapping and modeling of the environment; path planning and selection; path traversal and collision avoidance into which the navigation problem is decomposed, is shown in Figure 1.

Motion planning of mobile walking robots in uncertain dynamic environments based on the behavior dynamics of collision-avoidance is transformed into an optimization problem. Applying constraints based on control of the behavior dynamics, the decision-making space of this optimization.

IV. Virtual Projection Method

A virtual projection architecture system was designed which allows improvement and verification of the performance of dynamic force-position control of walking robots by integrating the multi-stage fuzzy method with acceleration solved in position-force control and dynamic control loops through the ZMP method for movement in non-structured environments and a bayesian approach of simultaneous localization and mapping (SLAM) for avoiding obstacles in non-stationary environments. By processing inertial information of force, torque, tilting and wireless sensor networks (WSN) an intelligent high level algorithm is implementing using the virtual projection method.

The virtual projection method, presented in Figure 2, patented by the research team, tests the performance of dynamic position-force control by integrating dynamic control loops and a bayesian interface for the sensor network. The CMC classical mechatronic control directly actions the MS1, MSm servomotors, where m is the numbber of the robot's degrees of freedom. These signals are sent to a virtual control interface (VCI), which processes them and genrates the necessary signals for graphical representation in 3D on a graphical terminal CGD. A number of n control interface functions ICF1-ICFn ensure the development of an open architecture control system by intergrating n control functions in addition to those supplied by the CMC mechatronic control system. With the help of these, new control methods can be implemented, such as: contour tracking functions, control schemes for tripod walking, centre of gravity control, orientation control through image processing and Bayesian interface for sensor networks. Priority control real time control and information exchange management between the n interfaces is ensured by the multifunctional control interface MCI, interconnected through a high speed data bus.

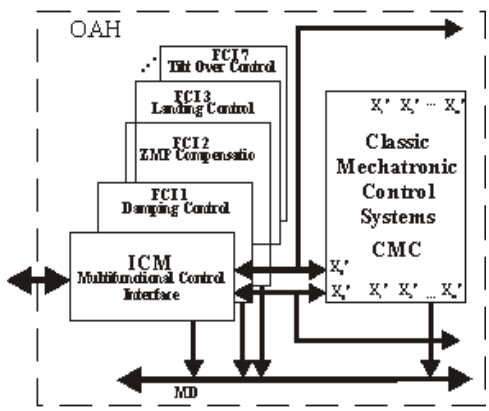

Fig. 2. The virtual projection method

Bayesian Interface for sensor networks.

To determine the priors for the model parameters and to calculate likelihood function (joint probability) we define a given random variable *x* whose probability distribution depends on a set of parameters $P = (P_1, P_2, \ldots P_p)$. Exact values of the parameters are not known with certainty, Bayesian reasoning assigns a probability distribution of the various possible values of these parameters that are considered as random variables. Bayes' theory is generally expressed through probabilistic statements as following:

$$P(A \mid B) = P(A) \quad \frac{P(B \mid A)}{(B)} \qquad (1)$$

P (A | B) is the probability of A given the event B occurs or the posteriori probability. Using Bayes' theory may be recurring, that if exist an a priori distribution (P (A) and a series of tests with experimental results B_1, $B_2,\ldots,B_n$..., expressed according to successive equations:

$$P(A \mid B_1) = P(A)\frac{P(B_1 \mid A)}{(B_1)}$$
$$P(A \mid B_1, B\ ) = P(A)\frac{P(B_1 \mid A)}{P(B_1)}\frac{P(B_2 \mid A)}{P(B\ )} \qquad (2)$$

$$P(A \mid B_1, B_2, ...B_n) = P(A \mid B_1, B_2, ...B_{-1}) - \frac{P(B_n \mid A)}{(B_n)}$$

A posteriori distribution called also belief, is used when the test results are known, being obtained as a new function a priori. The start of operations sequences in the Bayesian method regards the transformation γ. Recursive Bayesian updating is made under the Markov assumption: z_n is independent of $z_1,...,z_{n-1}$ if we know x.

$$\begin{aligned} P(x \mid z_1, \ldots, z_n) &= \frac{P(z \mid x)\, P(x \mid z_1, \ldots, z_{n-})}{P(z \mid z_1, \ldots, z_{n-})} \\ &= \eta\, P(z_n \mid x)\, P(x \mid z_1, \ldots, z_{-}) \quad (3) \\ &= \eta_{1\ldots n} \prod_{i=1\ldots} P(z_i \mid x)\, P(x) \end{aligned}$$

When there are no missing data or hidden variables the method for calculating $P(B_{si}, D)$ for some belief-network structure B_{Si} and database D is presented in [12].Let Q be the set of all those belief-network structures that have a non-zero prior probability. We can derive the posterior probability of B_{Si} given D as:

$$P(B_{Si} \mid D) = P(B_{Si}, D) / \sum B_{Si} \in QP(B_{Si}, D) \quad (4)$$

The ratio of the posterior probabilities of two belief-network structures can be calculated as a ratio for belief-network structures B_{si} and B_{sj}, using the equivalence:

$$P(B_{Si} \mid D) / P(B_{Sj} \mid D) = P(B_{Si}, D) / P(B_{Sj}, D) \quad (5)$$

which we can derive that:

$$P(B_{Si}, D) = P(D \mid B_{Si}) P(B_{Si}) \quad (6)$$

Term $P(B_{si})$ represents prior probability that a process with belief-network structure B_{Si}. To designate the possible values of *h*, ca be used the Markov blanket method, *MB(h)* [12, 13]. Suppose that among the *m* cases in *D* there are *u* unique instantiations of the variables in *MB(h).* Given these conditions it follows that:

$$P(D \mid B_S) = \sum_{G_i} ... \sum_{G} f(G_1, ..., G_u) \int_{B_p} [\prod_{t=1}^{m} P(C_t h_t \mid B_S, B_P)] f(B_S \mid B_P) dB_P \quad (7)$$

where G_i is a given group contains c_i case-specific hidden variables. Recall that u denotes only the number of unique instantiations *actually realized* in database D of the variables in the Markov blanket of hidden variable h. The number of such unique instantiations significantly influences the efficiency with which we can compute Equation 7.

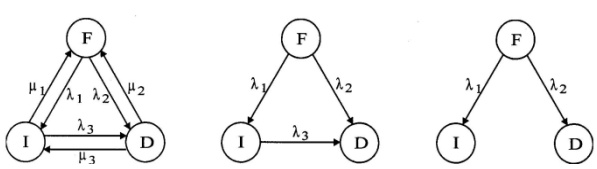

Fig.3 The model with three states for the robotic system

For any finite belief network, the number of such unique instantiations reaches a maximum regardless of how many cases there are in the database. That r denotes the maximum number of possible values for any variable in the database. If u and r are bounded from above, then the time to solve Equation 7 is bounded from above by a function that is polynomial in the number of variables n and the number of cases m. If u or r is large, however, the polynomial will be of high degree [12].

To model a robotic system requires considering in-between the two states of operating and faulting one or more intermediate states of partial success. In figure 3 is considered a robotic system characterized by three states: operating at full capacity (F), defect (D) and intermediate (I).

A generalized diagram of states is shown in figure 4, which included three intermediate states.

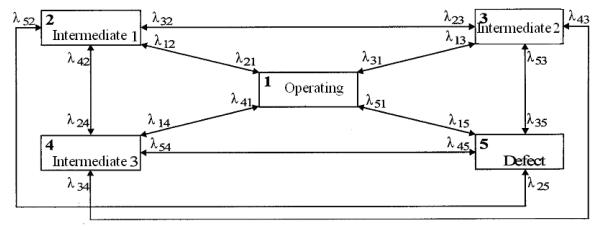

Fig. 4. Generalized diagram of states with three intermediate states

The Markov modeling technique requires to identify each intermediate state (in practice, more neighboring levels can be grouped together), to know the occupancy status of each component (Ti) and the number of transitions between states (Nij), which can calculate as follows:

- occupancy probability of "i"state: $P_i = \frac{T_i}{T_A}$

- transition intensity from state *"i"* in *"j":* $\lambda_{ij} = \frac{N_{ij}}{T_i}$,

where: $T_A = \sum_i T_i$ is analyzed time interval.

The number of intermediate states to be modeled in order to obtain a more accurate assessment of the reliability group is necessary to consider more than one intermediate state. Figure 5 presents a model with six states to assess the predictable transitions in a robotic system. The six states of the system are:

1 - operational state of robot;
2 - landing control
3 - balance control
4 - advance control
5 - wireless sensor networks (WSN) control
6 - unpredict event

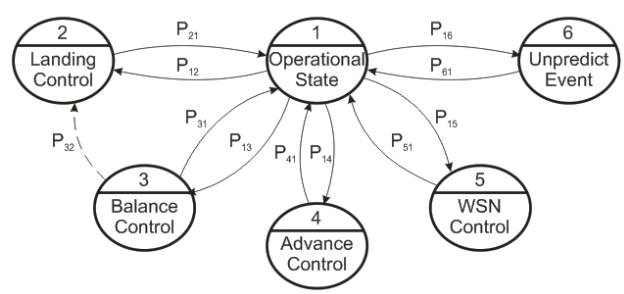

Fig.5. Modeling the states with possible transitions for robot

Based on the surveillance data in operation regime of robot were determined transition probabilities using of the relationship: $\hat{p}_{ij} = \frac{n_{ij}}{n_i}$, where n_{ij} is the transition from state

"i" in *"j"* in the analysis time interval; n_i is the number of all transitions from state *"i"* in any other states.

Values of these transition probabilities are: $\hat{p}_{12} = 0,247$; $\hat{p}_{13} = 0,32$; $\hat{p}_{14} = 0,125$; $\hat{p}_{15} = 0,205$; $\hat{p}_{16} = 0,103$; By applying the method Markov chains are obtain the occupancy probability of the sates for the robot: P_1=0,31; P_2=0,208; P_3=0,115; P_4=0,205; P_5=0,102; P_6=0,06.

The working diagram of the Petri network is presented in figure 6 (http://www-dssz.informatik.tu-cottbus.de). A token is assigned to P_3, and is assumed that the localizer initially knows its position. The Warning event t_5 fires when the localizer fails in estimating robot's accurate position for several steps. Two navigation primitives can be modeled as P_1, P_2, respectively. Initially, the robot selects its motion by a random switch comprising the transitions t_1 and t_2 with corresponds to probabilities P_1' and P'_2, respectively. The transition between them takes place according to the change of localizer states. The immediate transition t_3 means that the robot takes Contour tracking as soon as the localizer Warning event fires.

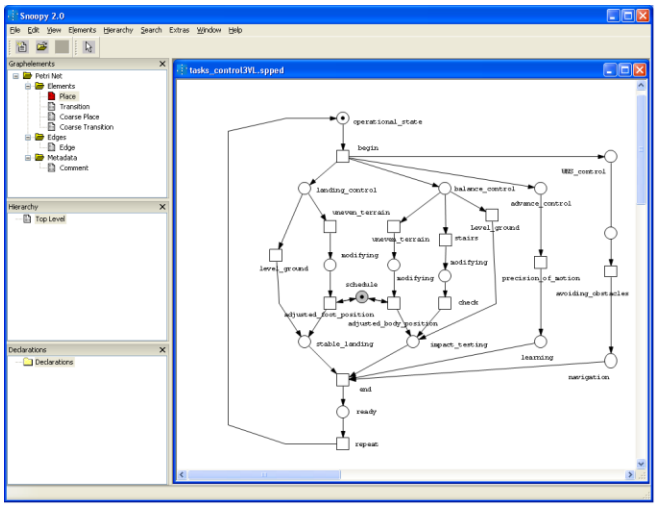

Fig.6. The Petri network diagram

The other transition between two primitives, t_2 and t_4, are modeled as timed transitions in order to express that the robot can change its current navigation primitive during the localizer Success state, if necessary.

V. Results and Conclusion

The control for walking robots is achieved by a control system with three levels. The first level is to produce control signals for motor drive mounted on leg joints, ensuring the robot moving in the direction required with a given speed. The language for this level is that of differential equations. The second level controls the walking, respectively it coordinates the movements, provides the data necessary to achieve progress. At this level, work is described in the language of algorithms types of walking. The third level of command defines the type of walking, speed and orientation.

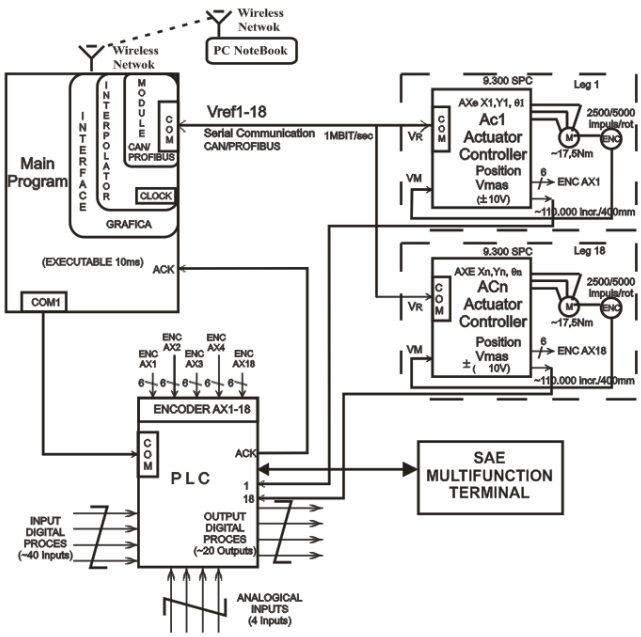

Fig.7. Open architecture system of the walking robot

At this level, the command may be provided by an operator who can use the control panel, in pursuit of its link with the robot, to specify the type of running and passing special orders (for the definition of the vector speed of movement).

To maintain the platform in a horizontal position, the information provided by the horizontality transducers (or verticality) is used, that sense walking robots deviation platform to the horizontal position. Restoring the horizontal position of the platform is achieved at the expense of vertical movement of different legs of support, as decided by the block to maintain balance. Returning to the fixed height of the platform is achieved by using information provided by the height transducer of the platform and by simultaneous control of vertical movement of all legs in support phase. From the analysis performed results the effectiveness of the proposed control strategy for a walking robot. The position of each actuator is controlled by a PD feedback loop, using encoder like transducers.

In HFPC control system, the PC system sends the reference positions to all actuators controllers simultaneously at an interval of 10 ms (100 Hz). Reference positions for the control of 18 actuators and actual positions on each axis robot obtained through interpolation are processed at an interval of 1 ms (1 kHz). Figure 1 shows the general configuration of the HFPC system for ZMP control method. The control system is distributive with multi-processor devices for joint control, data reception from transducers mounted on the robot, peripheral devices connected through a wireless LAN for off-line communications and CAN fast communication network for real time control. The HFPC system was designed in a distributed and decentralized structure to enable development of new applications easily and to add new modules for new hardware or software control functions. Moreover, the short time execution will ensure a faster feedback, allowing other programs to be performed in real

time as well, like the apprehension force control, objects recognition, making it possible that the control system have a human flexible and friendly interface.

ACKNOWLEDGMENT

This work was supported in part by the Romanian Academy and the Romanian Scientific Research National Authority under Grant 005/2007-2010.

REFERENCES

[1] *Raibert M.H., Craig J.J. - Hybrid Position / Force Control of Manipulators, Trans. ASME, J. Dyn. Sys., Meas., Contr., 102, June 1981, pp. 126-133.*

[2] *H. Zhang and R. P. Paul, Hybrid Control of Robot Manipulators, in International Conference on Robotics and Automation, IEEE Computer Society, March 1985. St. Louis, Missouri, pp.602-607.*

[3] *Luige Vladareanu, Gabriela Tont, Hongnian Yu and Danut A. Bucur, The Petri Nets and Markov Chains Approach for the Walking Robots Dynamical Stability Control, Proceedings of the 2011 International Conference on Advanced Mechatronic Systems, IEEE Sponsor, Zhengzhou, China, August 11-13, 2011.*

[4] *Vladareanu L., Ovidiu I. Sandru, Lucian M. Velea, Hongnian YU, The Actuators Control in Continuous Flux using the Winer Filters, Proceedings of Romanian Academy, Series A: Mathematics, Physics, Technical Sciences, Informantion Science, Volume: 10 Issue: 1 Pg.: 81-90, 2009.*

[5] *Jung-Yup Kim, Ill-Woo Park, Jun-Ho Oh, Walking Control Algorithm of Biped Humanoid Robot on Uneven and Inclined Floor, Springer Science, J. Intell Robot Syst (2007) 48:457–484, DOI 10.1007/s10846-006-9107-8.*

[6] *Luige Vladareanu1, Gabriela Tont, Ion Ion, Victor Vladareanu, Daniel Mitroi, Modeling and Hybrid Position-Force Control of Walking Modular Robots, ISI Proceedings, Recent Advances in Applied Mathematics, Harvard University, Cambridge, USA, 2010, pg. 510-518, ISBN 978-960-474-150-2, ISSN 1790-2769.*

[7] *Yoshikawa T., Zheng X.Z. - Coordinated Dynamic Hybrid Position/Force Control for Multiple Robot Manipulators Handling One Constrained Object, The International Journal of Robotics Research, Vol. 12, No. 3, June 1993, pp. 219-230.*

[8] *Vladareanu, L., Tont, G., Ion, I., Munteanu, M. S., Mitroi, D., "Walking Robots Dynamic Control Systems on an Uneven Terrain", Advances in Electrical and Computer Engineering, ISSN 1582-7445, e-ISSN 1844-7600, vol. 10, no. 2, pp. 146-153, 2010, doi: 10.4316/AECE.2010.02026.*

[9] *M.J. Stankovski, M.K. Vukobratovic, T.D. Kolemisevska-Gugulovska, A.T. Dinibutun, Automation System Redesign Using Manipulator for Steel-pipe Production Line, Proceedings of the 10th Mediterranean Conference on Control and Automation - MED2002 Lisbon, Portugal, July 9-12, 2002.*

[10] *Gabriela Tont, Luige Vladareanu, Mihai Stelian Munteanu, Dan George Tont Hierarchical Bayesian Reliability Analysis of Complex Dynamical Systems Proceedings of the International Conference on Applications of Electrical Engineering (AEE '10), pp 181-186, 6 pg., ISSN: 1790-2769, ISBN: 978-960-474-171-7, Malaysia, March 23-25, 2010.*

[11] *Abolfazl Jalilvand, Sohrab Khanmohammadi, Fereidoon Shabaninia Hybrid Modeling and Simulation of a Robotic Manufacturing System Using Timed Petri Nets, WSEAS Transactions on Systems, Issue 5, Volume 4, May 2005.*

[12] *C.G. Looney, Fuzzy Petri nets and applications, Fuzzy Reasoning in Information, Decision and Control Systems, Kluwer Academic Publisher, pp. 511-527, 1994*

[13] *Vladareanu Luige, Lucian M. Velea, Radu Ioan Munteanu, Adrian Curaj, Sergiu Cononovici, Tudor Sireteanu, Lucian Capitanu, Mihai Stelian Munteanu, Real time control method and device for robot in virtual projection, patent no. EPO-09464001, 18.05.2009.*

[14] *Khalil Shihab, Simulating ATM Switches Using Petri Nets", pp.1495-1502, WSEAS Transactions on Computers, Issue 11, Volume 4, 2005.*

[15] *Luige Vladareanu, Ion Ion, Marius Velea, Daniel Mitroi, The Robot Hybrid Position and Force Control in Multi-Microprocessor Systems, WSEAS Transation on Systems, Issue 1, Vol.8, 2009, pg.148-157, ISSN 1109-2777.*

[16] *J. Rummel, A. Seyfarth, Stable Running with Segmented Legs, The International Journal of Robotics Research 2008; 27; 919, DOI: 10.1177/027836490895136.*

[17] *Luige Vladareanu, Gabriela Tont, Radu A. Munteanu, et.all., Modular Structures in the Distributed and Decentralized Architecture, Proceedings of the International Conference On Parallel And Distributed Processing Techniques And Applications, ISBN: 1-60132-121-X, 1-60132-122-8 (1-60132-123-6)Nevada, USA, Published by CSREA Press, pp. 42-47, 6 pg., Las Vegas, Nevada, SUA, July 13-16, 2009*

Published in "Int. J. Advanced Mechatronic Systems", Vol. 5, No. 4, 2013, pp. 232-242, 11 p.