

# THE NAVIGATION OF MOBILE ROBOTS IN NON-STATIONARY AND NON-STRUCTURED ENVIRONMENTS

VICTOR VLADAREANU, GABRIELA TONT, LUIGE VLADAREANU,  
FLORENTIN SMARANDACHE

## Acknowledgements

This work was supported in part by the Romanian Academy, the Romanian Scientific Research National Authority under Grant PN-II-PT-PCCA-2011-3, 1-0190 and the FP7 IRSES RABOT project no. 318902/2012-2016.

## 1 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 is the skills needed to recognise the landmarks and objects that surround it, and to be able to localise 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 recognised objects are fixed and persist in the scene, defined as stationary environments, they can be regarded as landmarks, and can be used as reference points for self localisation. 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 (Looney, 1994). The approach of the localisation 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 localisation 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 centralised 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 (Jalilvand et al., 2009).

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 (Vladareanu et al., 2011; Kim et al., 2007). The localisation of a mobile robot is made using a particle filter that updates the belief of localisation which, and estimates the maximal posterior probability density for localisation. The causal and contextual relations of the sensing results and global localisation 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. This means moving the robot in sloping terrain, on steps or uneven environments which leads to modifications in the projection of the robot support surface and variable loads on the robot legs during movement. 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 localisation and mapping (SLAM).

## 2 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 decentralisation 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 (Vladareanu et al., 2011, 2012).

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 zero moment point (ZMP) method (Vladareanu et al., 2010b; Vladareanu and Capitanu, 2012). 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 (Vladareanu et al., 2009b; Capitanu et al., 2008).

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 analysed for dynamic walking of robots based on sensory tools such as force/torque and inertial sensors (Vladareanu et al., 2010a; Raibert and Craig, 1981; Zhang and Paul, 1985). Distributed control system architecture was integrated into the HFPC architecture so that it can be controlled with high efficiency and high performance.

## 3 Simultaneous localisation and mapping

A precise position error compensation and low-cost relative localisation method is studied in Kim et al. (2007) for structured environments using magnetic landmarks and hall sensors. The proposed methodology can solve the problem of fine localisation as well as global localisation by tacking landmarks or by utilising various patterns of magnetic landmark arrangement. The research in localisation and tracking methods using WSN have been developed based on radio signal strength intensity (RSSI) and ultrasound time of flight (TOF). Localisation based on radio frequency identification (RFID) systems have been used in fields such as logistics and transportation but the constraints in terms of range between transmitter and reader limits its potential applications (Yoshikawa and Zheng, 1993).

Many efforts have been devoted to the development of cooperative perception strategies exploiting the complementarities among distributed static cameras at ground locations, among cameras mounted on mobile robotic platforms, and among static cameras and cameras onboard mobile robots.

Computation-based closed-loop controllers put most of the decision burden on the planning task. In hazardous and populated environments mobile robots utilise 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 (Stankovski et al., 2002; Vladareanu et al., 2010b; Deng et al., 2011; Fei et al., 2012).

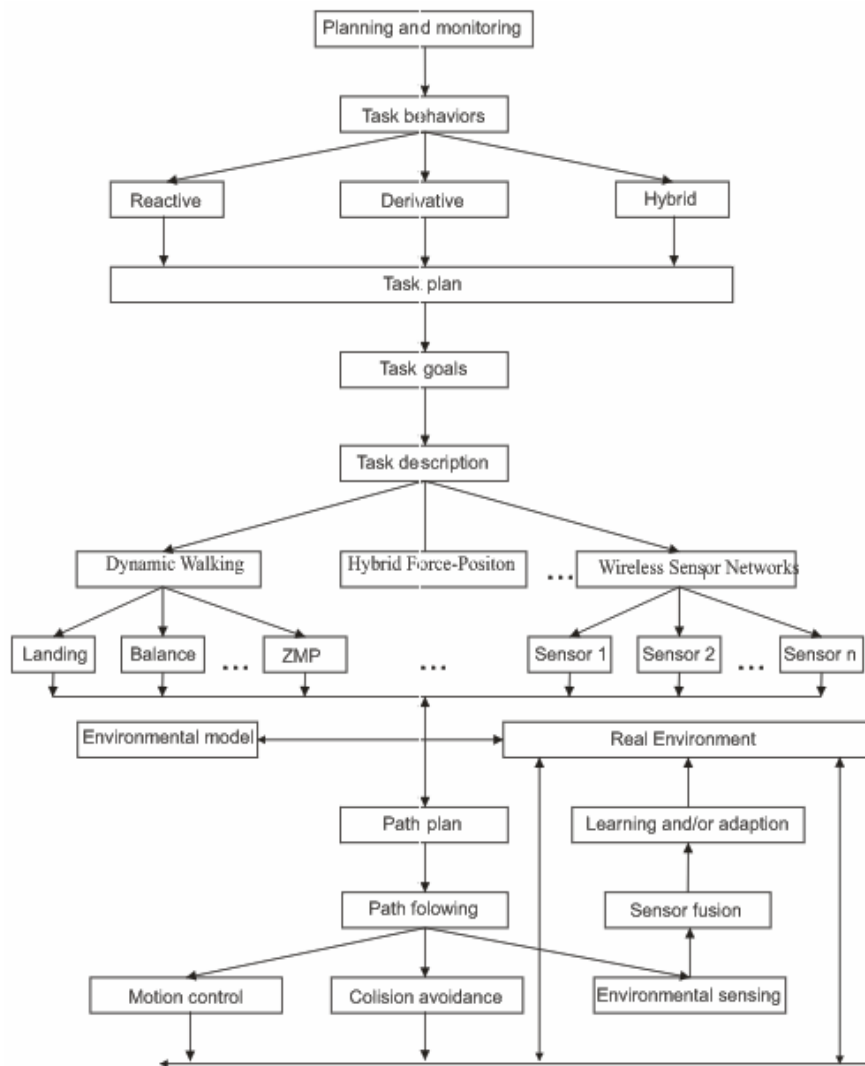
To find collision-free trajectories, in static or dynamic environments containing some obstacles, between a start and a goal configuration, the navigation of a mobile robot comprises localisation, motion control, motion planning and collision avoidance. Its task is also the online real-time

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 (Shihab, 2005; Iliescu et al., 2010). The techniques include computation-based closed-loop control, cost-based search strategies, finite state machines, and rule-based systems (Boscoianu et al., 2008; Rummel and Seyfarth, 2008).

Computation-based closed-loop controllers put most of the decision burden on the planning task. In hazardous and populated environments mobile robots utilise 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 localisation, motion control, motion planning and collision avoidance (Vladareanu et al., 2009b; Shihab, 2005). 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 modelling of the environment; path planning and selection; path traversal and collision avoidance into which the navigation problem is decomposed, is shown in Figure 1.

Figure 1 Mobile robot control system architect



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

#### 4 Strategy for dynamical stability

The walking robot is considered as a set of articulated rigid bodies, which are standing as a platform and leg elements. The static stability problem is solved by calculating the extremity of each leg position according to the system of axes attached to the platform, with origin at the centre of gravity of it.

The proposed walking robot control strategy is based on three approaches, for conforming to movement characteristics: real-time balance control, walking pattern control and predictable motion control. The first main task, balance control, leads to a control model that periodically modifies the walking scheme, depending on the sensory information received from the robot transducers.

In this paper we take into account the real-time balance control.

*Real-time balance control.* The balance control, leads to a control model that periodically modifies the walking scheme, depending on the sensory information received from the robot transducers. Real-time balance control presented in Figure 2 contains four types of reactive loops: damping control, ZMP compensation control, walk timing control and walk orientation control. The second main task, walking scheme control, represents a real-time control of the robot equilibrium using the reactions of inertial sensors. The walking control scheme can be changed periodically in accordance with the information received from the inertial sensors during the walking cycle, by processing them into

two real-time loops: platform swing amplitude control and platform rotation/advance control.

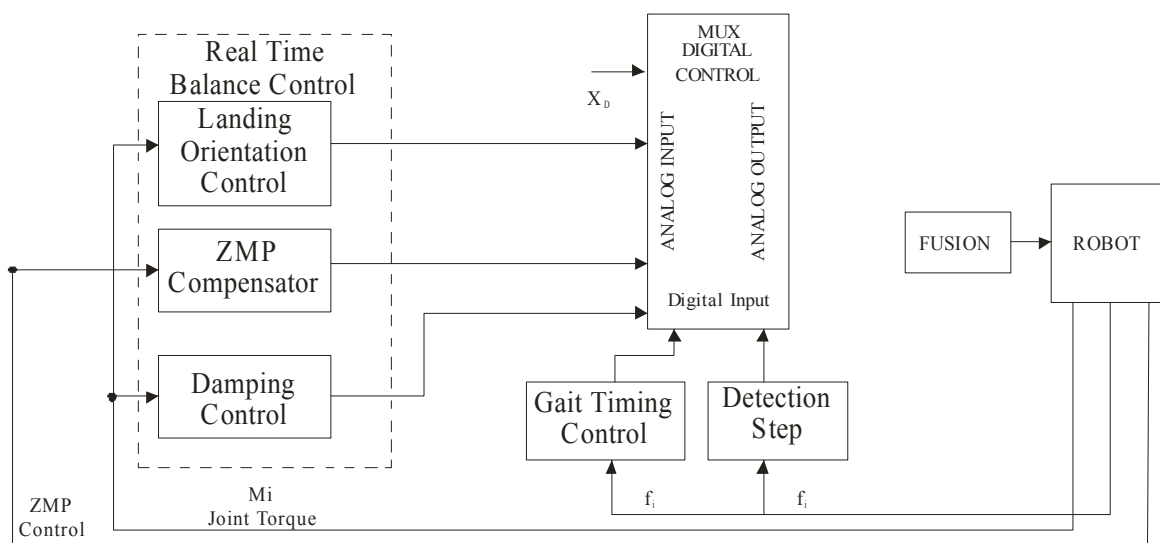
The third main task, predictable movement control, represents the control of predictable movement based on a fast decision from previous experimental data. Our research considers the following five dynamic control loops. *Landing orientation control* is achieved by integrating the torque measured for the entire gait and achieving a stable contact with the two ground surface by controlling the leg joint. A stable contact is obtained by adapting the leg articulations at ground surface, when an obstacle is preventing moving the leg on the required trajectory. The motion control will lead to a smooth walking. The control law for the landing orientation is:

$$u_c = u + \frac{T(s)}{C_L s + K_L} \tag{1}$$

where  $T$  is the measured torque,  $CL$  is the damping coefficient,  $KL$  is the rigidity,  $u$  is the leg reference angle and  $u_c$  is the leg's joint compensated reference angle.

*Damping control* aims to eliminate the oscillations that occur in the single support phase. The oscillations amplitude is measured in real-time by a torque transducer mounted on the robot joints, having compliant control functions of robot movement. A simple inverted pendulum equation with a joint in the single support phase, which opposes the damping forces of the leg joints was adopted for robot motion modelling. *ZMP compensation control* strategy consists in mathematical modelling of ZMP compensator through the spring-loaded inverted pendulum. A ZMP compensator is developed in single support phase (FSU), where the platform will move back and forth according to ZMP dynamics, because the damping loop is not sufficient to maintain a stable walking motion due to the ZMP movement influences.

**Figure 2** Real-time balance control of the walking robots motion



Regarding the construction of mathematical model, based on quasi-dynamic analysis, each leg is considered as a generator function, with limited accuracy for displacement systems. If the number of degrees of mobility is equal to  $n$  and if the interior limitations have the following form:

$$F_j(x_1, x_2, \dots, x_n) = 0, \quad j = \overline{1, m}, \quad n \geq m \quad (2)$$

then, in the differential equations structure:

$$\frac{dx_j}{dt} = f_i(x_1, x_2, \dots, x_n, u_1, u_2, \dots, u_s, t), \quad i = \overline{1, n} \quad (3)$$

there are arbitrary  $u_i$  coefficients, which are used to obtain the ‘stepping’ algorithm. For differential equations, the limitations imposed by the general (platform) base, where the legs are fixed, are applied in first case, and the limitations imposed on the supporting surface, secondly.

### 5 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 SLAM for avoiding obstacles in non-stationary environments. By processing inertial information of force, torque, tilting and WSN an intelligent high level algorithm is implementing using the virtual projection method.

The virtual projection method, presented in Figure 3, patented by the research team (Vladareanu et al., 2009a), 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, MS $m$  servomotors, where  $m$  is the number of the robot’s degrees of freedom. These signals are sent to a virtual control interface (VCI), which processes them and generates 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.

#### 5.1 Bayesian Interface for sensor networks

To determine the priors for the model parameters and to calculate likelihood function (joint probability) we define a

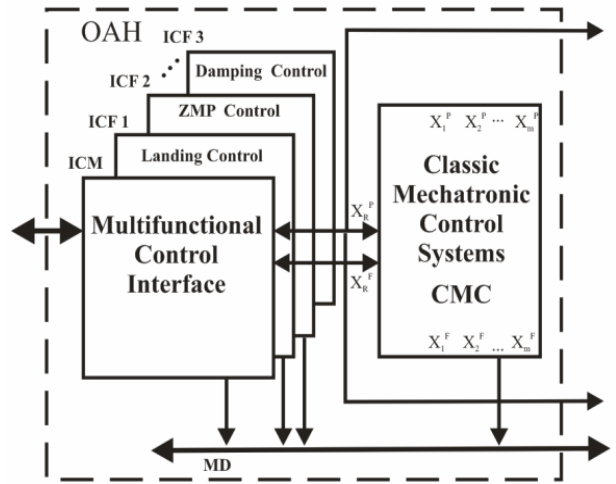
given random variable  $x$  whose probability distribution depends on a set of parameters  $P = (P_1, P_2, \dots, P_p)$ . Exact values of the parameters are not known with certainty, Bayesian reasoning assigns a probability distribution of the various possible values of these parameters that are considered as random variables. Bayes’ theory is generally expressed through probabilistic statements as following:

$$P(A|B) = P(A) \times \frac{P(B|A)}{P(B)} \quad (4)$$

$P(A|B)$  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  $B_1, B_2, \dots, B_n, \dots$ , expressed according to successive equations:

$$\begin{aligned} P(A|B_1) &= P(A) \frac{P(B_1|A)}{P(B_1)} \\ P(A|B_1, B_2) &= P(A) \frac{P(B_1|A)}{P(B_1)} \frac{P(B_2|A)}{P(B_2)} \\ P(A|B_1, B_2, \dots, B_n) &= P(A|B_1, B_2, \dots, B_{n-1}) \frac{P(B_n|A)}{P(B_n)} \end{aligned} \quad (5)$$

Figure 3 The virtual projection method



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  $\gamma$ . Recursive Bayesian updating is made under the Markov assumption:  $z_n$  is independent of  $z_1, \dots, z_{n-1}$  if we know  $x$ .

$$\begin{aligned} P(x|z_1, \dots, z_n) &= \frac{P(z_n|x) P(x|z_1, \dots, z_{n-1})}{P(z_n|z_1, \dots, z_{n-1})} \\ &= \eta P(z_n|x) P(x|z_1, \dots, z_{n-1}) \\ &= \eta_{1..n} \prod_{i=1..n} P(z_i|x) P(x) \end{aligned} \quad (6)$$

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 Looney (1994). 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:

$$P(B_{Si} | D) = P(B_{Si}, D) / \sum_{B_{Si} \in Q} P(B_{Si}, D) \quad (7)$$

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:

$$P(B_{Si} | D) / P(B_{Sj} | D) = P(B_{Si}, D) / P(B_{Sj}, D) \quad (8)$$

which we can derive that:

$$P(B_{Si}, D) = P(D | B_{Si}) P(B_{Si}) \quad (9)$$

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,  $MB(h)$  (Looney, 1994; Vladareanu et al., 2011). 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:

$$P(D | B_s) = \sum_{G_1} \dots \sum_{G_u} f(G_1, \dots, G_u) \int_{B_p} \left[ \prod_{t=1}^m P(C_t h_t | B_s, B_p) \right] f(B_s B_p) dB_p \quad (10)$$

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 realised* 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 (10). 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 (10) 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 (Vladareanu et al., 2011).

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 4 is considered a robotic system characterised by three states: operating at full capacity ( $F$ ), defect ( $D$ ) and intermediate ( $I$ ). A generalised diagram of states is shown in Figure 5, which included three intermediate states.

Figure 4 The model with three states for the robotic system

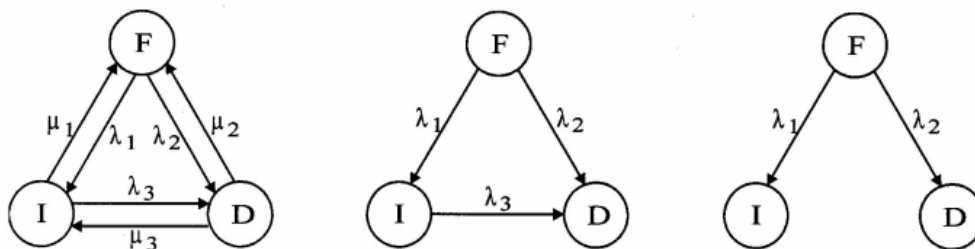
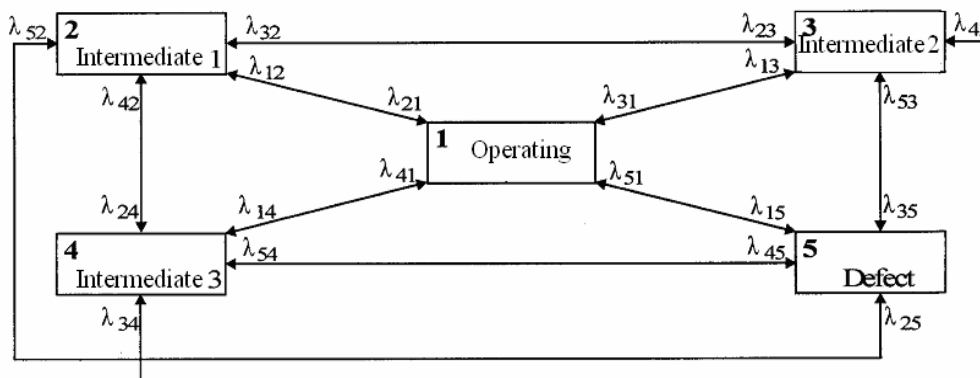


Figure 5 Generalised diagram of states with three intermediate states



The Markov modelling technique requires to identify each intermediate state (in practice, more neighbouring levels can be grouped together), to know the occupancy status of each component ( $T_i$ ) and the number of transitions between states ( $N_{ij}$ ), 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 analysed time interval.

The number of intermediate states to be modelled in order to obtain a more accurate assessment of the reliability group is necessary to consider more than one intermediate state. Figure 6 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 WSN control
- 6 unpredicted event.

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:  $\hat{p}_{12} = 0,217$ ;  $\hat{p}_{13} = 0,29$ ;  $\hat{p}_{14} = 0,135$ ;  $\hat{p}_{15} = 0,235$ ;  $\hat{p}_{16} = 0,123$ . By applying the method Markov chains are obtain the

occupancy probability of the sates for the robot:  $P_1 = 0.27$ ;  $P_2 = 0.19$ ;  $P_3 = 0.115$ ;  $P_4 = 0.235$ ;  $P_5 = 0.122$ ;  $P_6 = 0.068$ .

The working diagram of the petri network is presented in Figure 7 (<http://www-dssz.informatik.tu-cottbus.de>).

A token is assigned to  $P_3$ , and is assumed that the localiser initially knows its position. The warning event  $t_5$  fires when the localiser fails in estimating robot's accurate position for several steps. Two navigation primitives can be modelled 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 localiser states.

The immediate transition  $t_3$  means that the robot takes Contour tracking as soon as the localiser Warning event fires. The other transition between two primitives,  $t_2$  and  $t_4$ , are modelled as timed transitions in order to express that the robot can change its current navigation primitive during the localiser Success state, if necessary.

## 6 Results and conclusions

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. 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).

Figure 6 Modelling the states with possible transitions for robot

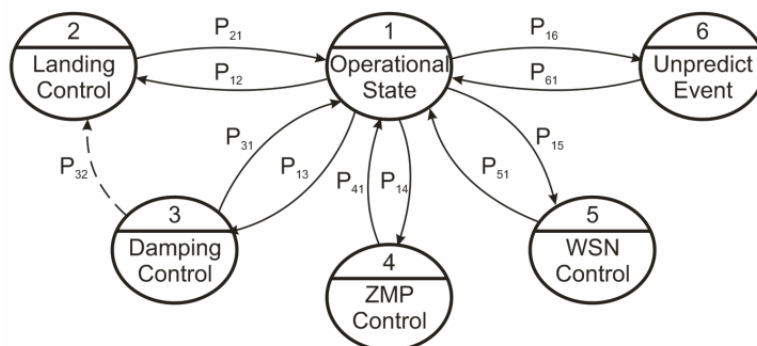
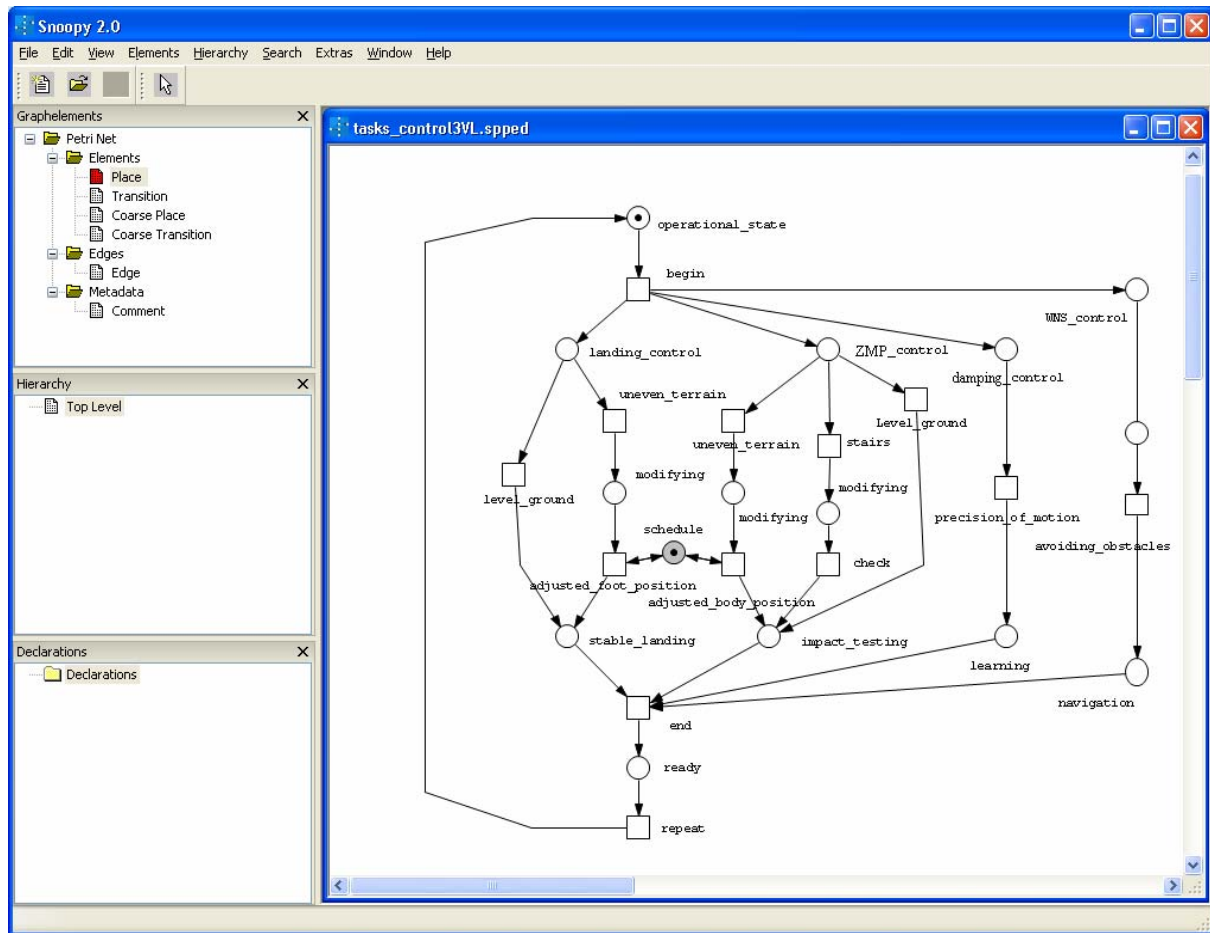


Figure 7 The petri network diagram (see online version for colours)



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).

The ready to walk position is a robot base position, before the actual walk. For this position, the robot lowers the platform by bending the leg joints. The reason is to

prevent the singular problem of the inverse cinematic and to achieve a stable walking with a constant platform height from the ground. To be observed that the platform height is linked to the dynamic properties of the robot. When the robot walks, it is periodically in the unique support phase. In this phase, the robot can be similar to the simple inverted pendulum model on the coronal plane and its natural frequency is:

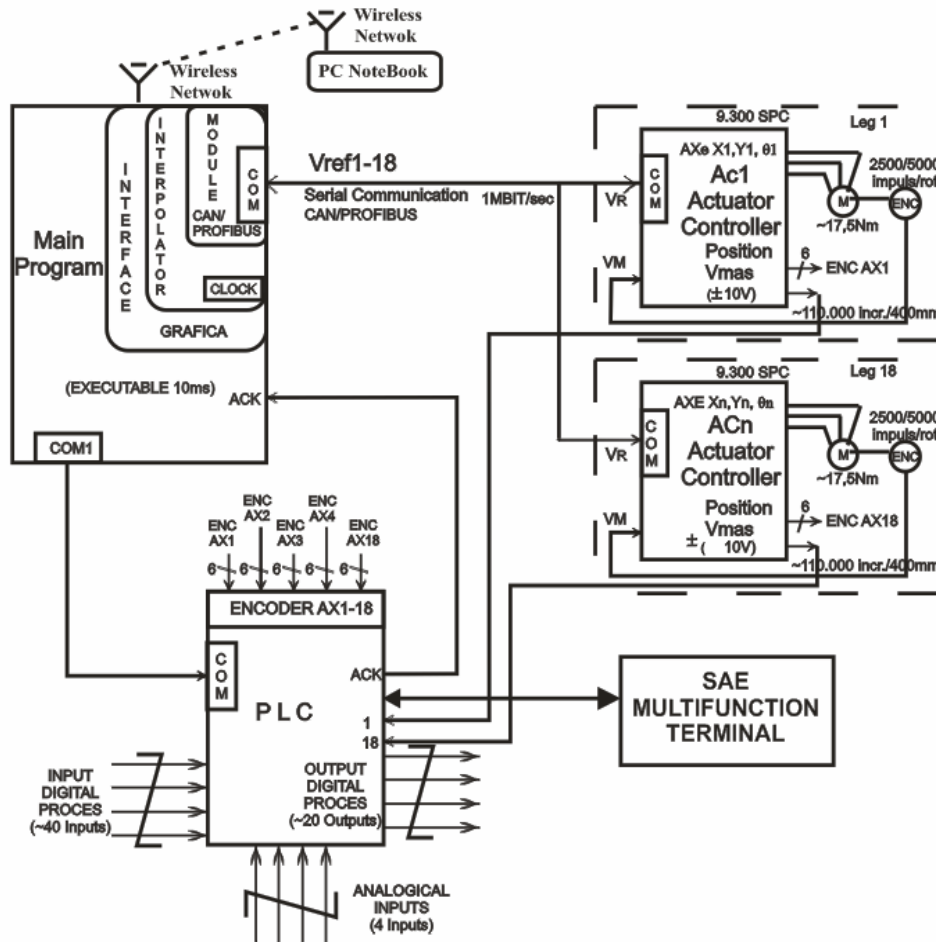
$$f = \frac{1}{2\pi} \sqrt{\frac{g}{l}} (Hz) \tag{11}$$

where  $g$  and  $l$  are the desired acceleration due to gravity and respective the height from the ground of the robot's centre of mass.

Certainly, the natural frequency of the simple inverted pendulum exists in theory, because the robot's tilt is limited by a specific angle. Thus, one can determine the walking period for a smooth motion in two phases (for the tripod walking) and efficient power consumption. For example, for a robot with the height  $l$  of approximately 900 mm and the balance of 40 mm results the natural frequency of 0.526 Hz. Figure 8 shows the general configuration of the HFPC system for ZMP control method.



Figure 8 Open architecture system of the walking robot



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 offline communications and CAN fast communication network for real-time control. The HFPC system was designed in a distributed and decentralised structure to enable development of new applications easily and to add new modules for new hardware or software control functions.

The proposed petri nets and Markov chains approach provides a promising solution towards the development quantitative approach of dynamic discreet/stochastic event systems of task planning of mobile robots. For a deeper insight into control and communication of governing task assignment of the robot, the entire discrete-event dynamic evolution of task sequential process have to be linguistically described in terms of representations.

This approach has the potential to model more complex relationships between target parameters. 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.

### References

Boscoianu, M., Molder, C., Arhip, J. and Stanciu, M.I. (2008) 'Improved automatic number plate recognition system', *Proceedings of International Conference on Mathematical Methods, Computational Techniques, Non Linear Systems, Intelligent Systems*, pp.234–239, Corfu, Greece, October 20–22, ISBN, ISSN 1790-2769.

Capitanu, L., Vladareanu, L., Onisoru, J., Iarovici, A., Tiganesteanu, C. and Dima, M. (2008) 'Mathematical model and artificial knee joints wear control', *Journal of the Balkan Tribological Association*, Vol. 14, No. 1, pp.87–101.

Deng, M., Jiang, C. and Inoue, A. (2011) 'Operator-based robust control for nonlinear plants with uncertain non-symmetric backlash', *Asian Journal of Control*, Vol. 13, No. 2, pp.317–327.

Fei, M., Deng, W., Li, K. and Song, Y. (2012) 'Stabilisation of a class of networked control systems with stochastic properties', *Int. J. of Modelling, Identification and Control*, Vol. 17, No. 1, pp.1–7, DOI:10.1504/IJMIC.20048634.

- Iliescu, M., Vladareanu, L. and Spanu, P. (2010) 'Modelling and controlling of machining forces when milling polymeric composites', *Materiale Plastice*, Vol. 47, No. 2, pp.231–235.
- Jalilvand, A., Khanmohammadi, S. and Shabaninia, F. (2009) 'Hybrid modeling and simulation of a robotic manufacturing system using timed petri nets', *WSEAS Transactions on Systems*, May, Vol. 4, No. 5, pp.273–282.
- Kim, J-Y., Park, I-W. and Oh, J-H. (2007) 'Walking control algorithm of biped humanoid robot on uneven and inclined floor, Springer Science', *J. Intell Robot Syst.*, Vol. 48, No. 1, pp.457–484, DOI 10.1007/s10846-006-9107-8.
- Looney, C.G. (1994) *Fuzzy Petri Nets and Applications, Fuzzy Reasoning in Information, Decision and Control Systems*, Norwill, (Ed.), pp.511–527, Kluwer Academic Publisher.
- Raibert, M.H. and Craig, J.J. (1981) 'Hybrid position/force control of manipulators', *Trans. ASME, J. Dyn. Sys., Meas., Contr.*, June, Vol. 103, No. 2, pp.126–133.
- Rummel, J. and Seyfarth, A. (2008) 'Stable running with segmented legs', *The International Journal of Robotics Research*, Vol. 27, No. 8, p.919, DOI: 10.1177/027836490895136.
- Shihab, K. (2005) 'Simulating ATM switches using petri nets', *WSEAS Transactions on Computers*, Vol. 4, No. 11, pp.1495–1502.
- Stankovski, M.J., Vukobratovic, M.K., Kolemisevska-Gugulovska, T.D. and Dinibutun, A.T. (2002) '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.
- Vladareanu, L. and Capitanu, L. (2012) 'Hybrid force-position systems with vibration control for improvement of hip implant stability', *Journal of Biomechanics*, Vol. 45, Suppl. 1, S279, ISSN: 0021-9290, FI 2011: 2,434, Proceedings of ESB2012 – 18th Congress of the European Society of Biomechanics, Lisbon, Portugal.
- Vladareanu, L., Munteanu, R., Curaj, A. and Ion, I.N. (2009b) 'Open architecture systems for Mero walking robots control', *Proceedings of the European Computing Conference*, Vol. 28, No. 6, pp.437–443.
- Vladareanu, L., Tont, G., Ion, I., Velea, L.M., Gal, A. and Melinte, O. (2010a) 'Fuzzy dynamic modeling for walking modular robot control', *Proceedings of the 9th WSEAS International Conference on Application of Electrical Engineering*, pp.163–170.
- Vladareanu, L., Tont, G., Ion, I., Vladareanu, V. and Mitroi, D. (2010b) 'Modeling and hybrid position-force control of walking modular robots', *ISI Proceedings, Recent Advances in Applied Mathematics*, Harvard University, Cambridge, USA, pp.510–518, ISBN 978-960-474-150-2, ISSN 1790-2769.
- Vladareanu, L., Tont, G., Vladareanu, V., Smarandache, F. and Capitanu, L. (2012) 'The navigation mobile robot systems using Bayesian approach through the virtual projection method', *The 2012 International Conference on Advanced Mechatronic Systems*, pp.498–503, ISSN: 1756-8412, 978-1-4673-1962-1, INSPEC Accession Number: 13072112, 18–21 September 2012, Tokyo.
- Vladareanu, L., Tont, G., Yu, H. and Bucur, D.A. (2011) '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.
- Vladareanu, L., Velea, L.M., Munteanu, R.I., Curaj, A., Cononovici, S., Sireteanu, T., Capitanu, L. and Munteanu, M.S. (2009a) *Real Time Control Method and Device for Robot in Virtual Projection*, Patent No. EPO-09464001, 18.05.2009.
- Yoshikawa, T. and Zheng, X.Z. (1993) 'Coordinated dynamic hybrid position/force control for multiple robot manipulators handling one constrained object', *The International Journal of Robotics Research*, June, Vol. 12, No. 3, pp.219–230.
- Zhang, H. and Paul, R.P. (1985) *Hybrid Control of Robot Manipulators*, in *International Conference on Robotics and Automation*, March, pp.602–607, IEEE Computer Society, St. Louis, Missouri.