A Fuzzy-CMAC Based Hybrid Intuitive Approach for Biped Robot's Adaptive Dynamic Walking

Christophe Sabourin, Kurosh Madani
Images, Signals and Intelligence Systems Laboratory (LISSI / EA 3956)
PARIS XII University, Senart-Fontainebleau Institute of Technology,
Bât.A, Av. Pierre Point, F-77127 Lieusaint, France,
{sabourin ; madani }@univ-paris12.fr

Olivier Bruneau
Laboratoire Vision et Robotique,
Ecole Nationale Supérieure d'Ingénieurs de Bourges,
10 Boulevard Lahitolle,18020 Bourges Cedex, France
Olivier.bruneau@ensi-bourges.fr

Abstract: *In this paper, we have proposed a new neural network based hybrid intuitive approach for biped robot's adaptive walking. Our approach takes advantage on the one hand from a Fuzzy-CMAC based stage and on the other hand from high level intuitive control strategy involving only the regulation of the robot's average velocity. The main interest of this approach is to proffer to the walking robot autonomy and robustness, involving only one parameter: the average velocity. Experimental results validating the proposed intelligent hybrid control strategy have been presented and discussed.*

Keywords: Artificial Neural Networks, Fuzzy-CMAC, Autonomous Walking, Biped Robots.

I. INTRODUCTION

One of the most challenging topics, over the recent decades, in the field of robotics concerned the design and the control of biped robots. Several potentialities make this foremost research area particularly appealing in the frame of middle and long term projection. On the fundamental side, advances in this research area can lead to a better comprehension of the human locomotion mechanisms. From, the applicative point of view, it could concern a wide spectrum of applications among which: the design of more efficient prosthesis and the construction of more sophisticated humanoid robots for interventions in hostile environments.

However, if autonomy and decision ability of such biped humanoid robots is a vast dare, their basic locomotion remains still today a big challenge. If it is true that a number of already constructed prototypes (Asimo [1] and HRP-2P [2], have proved the feasibility of such robots, it is also factual that the performances of these walking machines are still far from equalizing the human's dynamic locomotion process. That is why the design of new control schemes allowing adaptability to complex environment for real dynamic walking is thus today fundamental. In fact, such robots must be able to adapt themselves automatically to indoor and outdoor human environments. Consequently, it is necessary to develop appropriated control strategies in order to allow them, on the one hand to adapt their gait to the complex environment and on the other hand, to counteract external perturbations.

In the field of bipeds' locomotion, the control strategies could be classified in two main categories. The first one is based on a kinematics and dynamic modeling of the whole robot's mechanical structure, implying to identify perfectly the intrinsic parameters of its mechanical structure. However, additionally to high precision measurements (of the limbs' angles, velocities and accelerations) requirements and to a precise evaluation of interaction forces (between feet and ground), the modeling of whole biped robot taking into account real environment remains a very complex task. That is why the computing of the on-line trajectories are generally performed using simplified models ([3], [4], [5], [6]), making this first strategy not always well adapted when biped robots move in real environment including internal and external perturbations, and changes of the foot/ground interactions. The second solution consists to use the soft-computing techniques (fuzzy logic, neural networks, genetic algorithm, etc...) and heuristically established rules resulting from the expertise of the walking human. Two main advantages distinguish this second class of approaches. Firstly, it is not necessary to know perfectly the characteristics of the mechanical structure. Secondly, this category of techniques takes advantage from learning capabilities. This last point constitutes a key point for autonomy of the biped robot.

We are investigating fully autonomous biped robot's walking based on a soft-computing approach. In this paper, we present a CMAC (Cerebellar Model Control Articulation) neural network based adaptive control strategy able to generate changing walking gait. This neural network has already used to generate the joint trajectories of the swing leg with fixed (constant) gait [7].

In this paper, we show how it is possible to change the walking gait by using a fuzzy based fusion of different trajectories learned by a set of CMAC neural networks. The validation of proposed approach has been done on an under-actuated robot: RABBIT [8], [9]. This robot constitutes the central point of a project, within the framework of CNRS (Centre Nationale de la Recherche Scientifique) ROBEA (ROBotique et Entité Artificielle) program [10], concerning the control of walking and running biped robots, involving several French laboratories.

This paper is organized as follows. Before describing our Fuzzy-CMAC hybrid strategy in section IV, section II presents the RABBIT robot and the numerical model simulating the dynamic behavior of this under-actuated robot. Then section 3 reminds structure and principles of the CMAC-like neural network. Section V gives the main obtained validation results. Finally, conclusions and further perspectives of the presented work are given by the last section.

II. RABBIT ROBOT AND ITS BEHAVIOR SIMULATION TOOL

This robot is composed of two legs and a trunk and has no foot as shown on figure 1. The characteristics (masses and lengths of the limbs) of this biped robot are summarized in table 1.

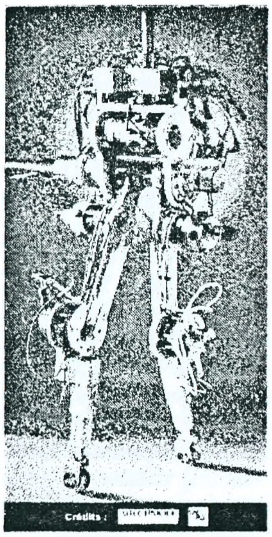

Fig.1 – RABBIT prototype's photograph.

Table 1. Masses and lengths of the robot's limbs

Limb	Weight (kg)	Length (m)
Trunk	12	0.2
Thigh	6.8	0.4
Shin	3.2	0.4

The motions of this robot are included in the sagittal plane by using a radial bar link fixed at a central column that allows one to gait the direction of progression of the robot around a circle. Since the contact between the robot and the ground is just one point (passive Degree Of Freedom), the robot is under-actuated during the single support phase: there are only two actuators (at the knee and at the hip of the contacting leg) to control three parameters (vertical and horizontal position of the platform and pitch angle). If it is true, from design point of view, that RABBIT is simpler compared to a robot with feet, from the control theory point of view, the control of this robot is a more challenging task, particularly because, in phase of single support, the robot is under-actuated. It is interesting to note that this robot is minimal system able to generate a dynamic biped walking and running gaits.

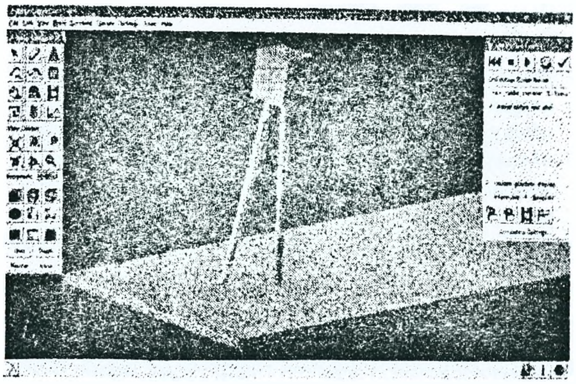

Fig.2 – Modeling RABBIT's behavior with ADAMS.

A numerical model of the previously described robot has been implemented within the ADAMS software. This software is able to simulate RABBIT's dynamic behavior and namely to calculate the absolute motions of the platform and the relative motions of the limbs when torques are applied on the joints by the virtual actuators (figure 2). The model used to simulate the interaction between feet and ground is exposed in [11]. The normal contact force is given by equation (1), where y and $\dot{y}$ are foot's position and velocity (with regard to the ground), respectively. k_c^n and λ_c^n represent the generalized stiffness and damping of the normal forces, respectively. They are chosen in order to avoid the rebound and to limit the penetration of the foot in the ground. The tangential contact forces are computed using equation (2) in the case of a contact without sliding or with the equation (3) if sliding occurs. x and $\dot{x}$ are respectively the position and the velocity of the foot with regard to the position of the contact point x_c at the instant of impact with ground. k_c^t and λ_c^t are respectively the generalized stiffness and damping of the tangential forces. λ_g is the coefficient of dynamic friction depending on the nature of surfaces in contact and μ_g a viscous damping coefficient during sliding. The interest of this model is that it is possible to simulate walking with or without phases of sliding allowing us to evaluate the robustness of the control.

$$F_c^n = \begin{cases} 0 & \text{If } y > 0 \\ -\lambda_c^n |y| \dot{y} + k_c^n |y| & \text{If } y \le 0 \end{cases} \quad (1)$$

$$F_c^t = \begin{cases} 0 & If \quad y > 0 \\ -\lambda_c^n \; x + k_c^t (x - x_c) & If \quad y \le 0 \end{cases} \quad (2)$$

$$F_c^t = \begin{cases} 0 & If \quad y > 0 \\ -\operatorname{sgn}(\dot{x})\lambda_g \, F_c^n - \mu_g \, \dot{x} & If \quad y \le 0 \end{cases} \quad (3)$$

Within the framework of a real robot's control, the morphological description of this one is insufficient. It is thus necessary to take into account the technological limits of the actuators in order to implement the control laws used in simulation on the experimental prototype. From the characteristics of servo-motor RS420J used for RABBIT, we thus choose to apply the following limitations:

- when velocity is included in [0 , 2000] rpm, the torque applied to each actuator is limited to 1.5 Nm what corresponds to a torque of 75 Nm at the output of the reducer (ration gear equal to 50),
- when velocity is included in [2000 , 4000] rpm the power of each actuator is limited to 315 W,
- when the velocity is bigger than 4000 rpm, the torque is imposed to be equal to zero.

III. CMAC NEURAL NETWORK

The CMAC is a neural network imagined by Albus from the studies on the human cerebellum [12] [13]. Despite its biological relevance, its main interest is the reduction of the training and computing times in comparison to other neural networks [14]. This is of course a considerable advantage for real time control.

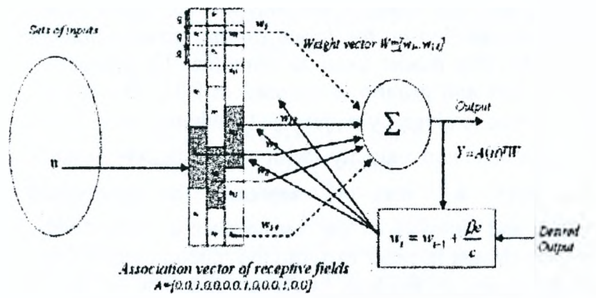

Fig.3 – Bloc- diagram showing example of a three layers CMAC ANN.

CMAC is an associative memory type neural network. Its structure includes a set of N_d detectors regularly distributed on several C_l layers. The receptive fields of these detectors cover the totality of the input signal but each field corresponds to a limited range of the inputs. On each layer, the receptive fields are shifted of a quantification step q. Consequently, the widths of the receptive fields are not always equal. When the value of the input signal is included in the receptive fields of a detector, this one is activated. For each value of the input signal, the number of activated detectors is equal to the number of layers C_l (parameter of generalization). Figure 3 shows a simplified organization of the receptive fields having 14 detectors distributed on 3 layers. Taking into account the receptive fields' overlapping, neighbouring inputs will activate common detectors. Consequently, this neural network is able to carry out a generalization of the output calculation for inputs close to those presented during learning (local generalization).

The output Y of the CMAC ANN (Artificial Neural Network) is computed using two mappings. The first mapping projects an input space point u into a binary associative vector $A=[a_1,\ldots,a_N]$. Each element of A is associated with one detector. When one detector is activated, the corresponding element in A of this detector is 1 otherwise it is equal to 0. The second mapping computes the output Y of the network as a scalar product of the association vector A and the weight vector $W=[w_1,\ldots w_N]$ according to the relation (4), where $(u)^T$ represents the transpose of the input vector.

$$Y = A(u)^T \, W \quad (4)$$

The weights of CMAC ANN are updated by using equation (5). $w(i)$ and $w(i-1)$ are, respectively, the weights before and after the training at each sample time i (discrete time). C_l is the generalization number of each CMAC and β is a parameter included in [0,1]. e is the error between the desired output Y^d of the CMAC and the computed output Y of the corresponding CMAC.

$$w(i) = w\ (i-1) + \frac{\beta e}{c_l} \quad (5)$$

The intrinsic structure of CMAC neural model is relatively well adapted for the control of complex systems and has already been subject of some researches in the field of the control of biped robots [15], [16]. However, the memory size depends firstly of the input signal's quantification step and secondly of the input space's size. For real applications, the CMAC memory size is very large because the quantification step must be small in order to increase precision and generally the size of the input space is greater than two. In order to solve this problem, hashing function is used. But in this case, because the weight memory is smaller than virtual addressing memory, some collisions can occur. Another problem occurring in the case of multi-input CMAC is the necessity to use a learning database covering the totality of the input space. This is due to the local generalization abilities of the CMAC and implies to do a lot of simulation.

IV. PROPOSED APPROACH

In this paper, we propose a new approach allowing a mixture of local and global generalization: the Fuzzy-CMAC. Our Fuzzy-CMAC approach is based on a fusion of the all outputs of each single-input CMAC. This fusion is carried out by using Takagi-Sugeno FIS (Fuzzy Inference System). This allows to decrease the memory size and to increase the generalization abilities in comparison to a multi-input CMAC.

Figure 4 gives the bloc diagram of the proposed hybrid architecture. Two main parts compose this architecture.

The first one computes the trajectory of the swing leg using several CMAC neural networks' outputs and a Fuzzy Inference System. The second one allows regulating the average velocity from a modification of the desired pitch angle at each new step. In this section, we present essentially the Fuzzy-CMAC. For more information about the control strategy, please see [7].

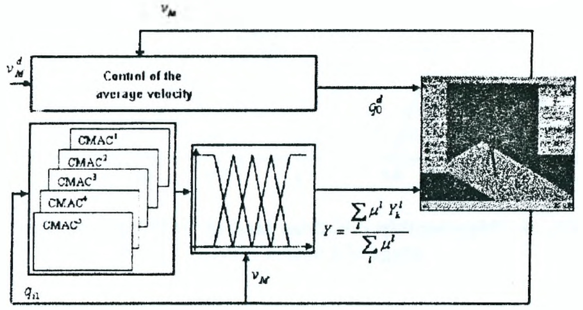

Fig.4 – Bloc- diagram of the Fuzzy-CMAC based hybrid control strategy.

The Fuzzy-CMAC approach requires two stages:

- during the first stage, we carry out the training of each CMAC. The data learned by the CMAC correspond at the trajectories of the swing leg.
- during the second stage, we use a fusion of the trajectories learned by CMAC ANN.

A. LEARNING CMAC ANN

During the learning phase, we use a set of pragmatic rules allowing generation of a stable dynamic walking with velocity variations corresponding to different step lengths [17]. This intuitive control strategy is based on three points:

- the observation of the relations between joint movements and the evolution of the parameters describing the motions of the robot platform,
- an interpretation of the muscular behaviour,
- the analysis of the intrinsic dynamics of a biped.

The goal is to generate the legs' movements by using a succession of passive and active phase based on parametric rules determined from the three aforementioned points. Also, it becomes possible to modify step length, average velocity by an adjustment of different parameters allowing a set of reference trajectories (data) which are learned, as it has been previously indicated, by a set of CMAC neural networks. However, during this first stage, we consider that the virtual robot move in an ideal environment (without disturbance). We also assume that frictions are negligible. Figure 5 shows the bloc diagram of the training strategy. The trajectories of the swing leg (in terms of joint positions and velocities) are learned by four "single-input/single-output" $CMAC_k$ with k=1,..,4 neural networks (four trajectories to learn). The learned trajectories are joint angles q_{i1} and q_{i2}, and the two corresponding angular velocities $\dot{q}_{i1}$ and $\dot{q}_{i2}$. q_{i1} and q_{i2} are respectively the measured angles at the hip and the knee of the leg i. In the same way, $\dot{q}_{i1}$ and $\dot{q}_{i2}$ are respectively the measured angular velocities at the hip and the knee of the leg i (see figure5).

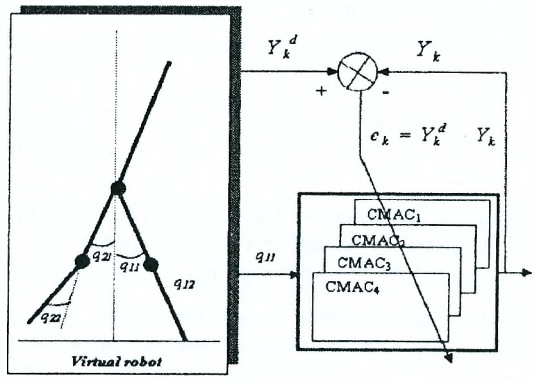

Fig. 5 – Learning strategy principle's bloc diagram.

When leg 1 is in support (e.g. $q_{12} = 0$), the input of each CMAC ANN is the angle q_{11} (e.g. $u = q_{11}$) and when leg 2 is in support ($q_{22} = 0$), the CMAC networks' input is the angle q_{21} (e.g. $u = q_{21}$). Consequently, the trajectories learned by the neural networks are not function of time but depends on robot's geometrical patterns. Furthermore, we consider that the trajectories of each leg in swing phase are identical. This allows reducing on the one hand the required number of CMAC neural networks (reducing by two) and on the other hand the training time. Each $CMAC_k$ have 6 C_l layers. The width of the receptive fields, L_f, is equal to 1.5° and the quantification step q is equal to 0.25°. The input signal is included in [q_{il}^{max} - q_{il}^{min}].

During the training stage, five trajectories corresponding to five different average velocity values (V_M measured in m/s) included in [0.4 , 0.8] interval are learned by five CMAC based modules. Each module (labelled $CMAC^l$, with $l \in \{1,2,3,4,5\}$) includes four $CMAC_k$ neural networks (corresponding to the four above-mentioned robot's trajectories). Table 2 gives the main parameters, used in the intuitive control, during the learning phase according to the desired average velocity V_M.

Table 2. Parameters used during the learning phase

	V_M (m/s)	q_r^d (°)	q_{sw}^d (°)	q_0^d (°)
$CMAC^1$	0.4	20	-7	3.5
$CMAC^2$	0.5	25	-10	3.0
$CMAC^3$	0.6	30	-15	2.5
$CMAC^4$	0.7	35	-20	8.0
$CMAC^5$	0.8	40	-25	8.0

q_r^d and q_0^d are respectively the desired relative angle between the two thighs and the desired pitch of the trunk.

q_{sw}^d corresponds to the desired angle of the knee at the end of the knee extension of the swing leg just before the double contact phase. V_M is computed by using relation (6) where L_{step} is the distance between the two feet at the moment of double impact and t_{step} is the duration of the step (from takeoff to landing of the same leg).

$$V_M = \frac{L_{step}}{t_{step}} \tag{6}$$

The number of the receptive field N_d for each reference walking is given by equation (7). N_d depends of the limited range [q_{i1}^{max} , q_{i1}^{min}] of the input signal, the width of the receptive fields L_f and the number of layer C_l. Table 3 gives the number of the receptive fields necessary of each CMAC[l] in function of q_{i1}^{min} and q_{i1}^{max} .

$$N_d = \frac{q_{i1}^{max} - q_{i1}^{min}}{L_f} C_l + C_l - 1 \tag{7}$$

Table 3. Number of the receptive fields for each CMAC[l]

	V_M (m/s)	q_{i1}^{min} (°)	q_{i1}^{max} (°)	N_d
$CMAC^1$	0.4	-8	11.5	83x4
$CMAC^2$	0.5	-10.5	15	107x4
$CMAC^3$	0.6	-12	18	125x4
$CMAC^4$	0.7	-15	21	149x4
$CMAC^5$	0.8	-16.5	24	167x4

B. FUZZY-CMAC

The final desired trajectory Y and the corresponding angular positions and velocities are computed by the Fuzzy based level, inherent to the Fuzzy-CMAC architecture, on the basis of predefined membership functions. These angular positions (q_{i1} and q_{i2}), and the two corresponding angular velocities ($\dot{q}_{i1}$ and $\dot{q}_{i2}$) are carried out by using fusing the five aforementioned learned trajectories. This fusion is realized by using a Fuzzy Inference System (FIS), composed of the five following rules, where Y^l corresponds to the output of $CMAC^l$ with $l \in \{1,2,3,4,5\}$:

- IF V_M IS VerySmall THEN $Y = Y^1$
- IF V_M IS Small THEN $Y = Y^2$
- IF V_M IS Medium THEN $Y = Y^3$
- IF V_M IS Big THEN $Y = Y^4$
- IF V_M IS VeryBig THEN $Y = Y^5$

Figure 6 gives the membership functions corresponding to the upper-indicated FIS rules. The average velocity is modelled by five fuzzy sets ("VerySmall", "Small", "Medium", "Big", "VeryBig"). The desired trajectory Y is computed by using equation (8), where μ^l represents the fuzzy membership parameter.

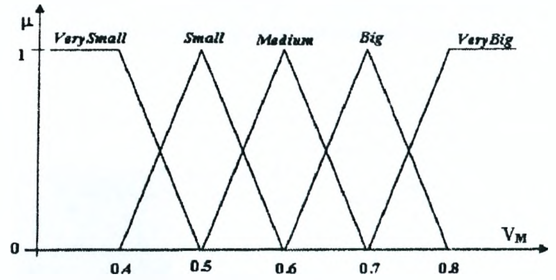

Fig. 6 – Membership functions used by Fuzzy Inference stage of Fuzzy-CMAC.

$$Y = \frac{\sum_{l=1}^{5} \mu^l Y^l}{\sum_{l=1}^{5} \mu^l} \tag{8}$$

V. VALIDATION RESULTS

Figure 7 gives the stick diagram of the biped robot's walking sequence when the desired average velocity increases. It must be noticed that the control strategy allows adapting automatically the pitch angle and the step length as the human being.

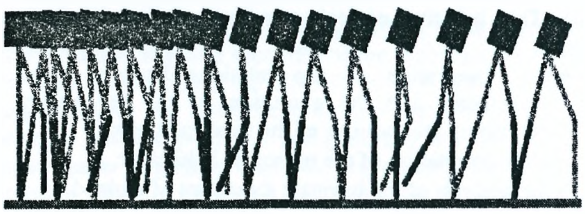

Fig. 7 – Stick diagram showing a walking sequence of the biped robot with increasing average velocity increases.

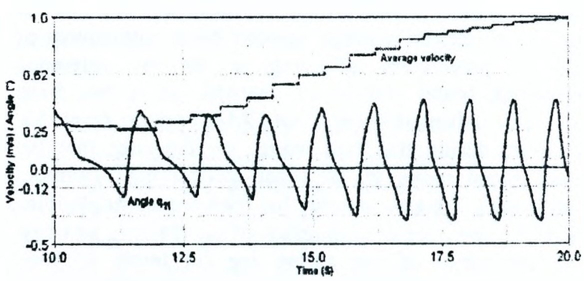

Fig. 8 – Evolution of the angle q_{11} when V_M increase.

Figures 8 and 9 show the evolution of the angle q_{11} and q_{12} when the average velocity increases. It is interesting to note that, as it has been mentioned before, the trajectory depends on the one hand on the stance leg's geometrical position and on the other hand on the measured average velocity. The regulation at each step of the average velocity is obtained thanks to an adequate adjustment of the pitch angle (see figure 4). In fact, the biped robot is able to adapt his gait with only one parameter: the real average velocity. Furthermore, if the robot is pushed forwards, the average velocity increases and consequently the step length increases in order to compensate this kind of perturbations.

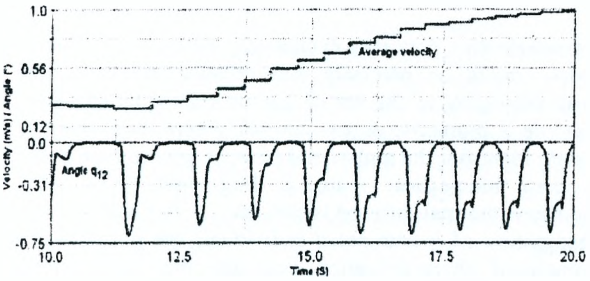

Fig. 9 – Evolution of the angle q_{12} when V_M increase.

Consequently, the proposed approach allows us, with a limited number of learned walking, to control biped walking robot in autonomous manner. Furthermore, the Fuzzy-CMAC permits to reduce considerably the size memory of the ANN.

VI. CONCLUSIONS

In this paper, we have proposed a new neural network based hybrid intuitive approach for biped robot's adaptive walking taking advantage on the one hand from a Fuzzy-CMAC issued computation of robot's swing leg's desired trajectory and on the other hand from high level intuitive control strategy involving only the regulation of the robot's average velocity. The main interest of this approach is to proffer to the walking robot autonomy and robustness. The obtained results show the adaptability of the walking step length. Furthermore, the Fuzzy-CMAC approach allows decreasing the memory size in comparison to the traditional multi-input CMAC ANN.

Future works will focus firstly on the extension of the Fuzzy-CMAC approach in order to increase the autonomy of the walking robot according to the nature of the environment (get up and down stairs for instance), avoidance and dynamic crossing obstacles and secondly on the experimental validation of our approach.

VII. REFERENCES

[1] Y. Sakagami, R. Watanabe, C. Aoyama, S. Matsunaga, N. Higaki, K. Fujimura. The intelligent ASIMO: system overview and integration. Proc. IEEE Conf. on Intelligent Robots and Systems, 2002, pp. 2478-2483.

[2] K. Kaneko, F. Kanehiro, S. Kajita, H. Hirukawa, T. Kawasaki, M. Hirata, K. Akachi, T. Isozumi. Humanoid robot HRP-2. Proc. IEEE Conf. on Robotics and Automation, 2004, pp. 1083--1090.

[3] M. Vukobratovic, B. Borovac. Zero moment point – thirty five years of its live. International Journal of Humanoid Robotics, 2004, Vol.1 N°1, pp. 157-173.

[4] S. Kajita, F. Kaneniro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi and H. Hirukawa. Biped walking pattern generation by using preview control of Zero-Moment Point. Proc. IEEE Conf. on Robotics and Automation, 2003, pp. 1620 -1626.

[5] Q. Huang, K. Yokoi, S. Kajita, K. Kaneko, H. Arai, N. Koyachi, K. Tanie. Planning walking patterns for a biped robot. IEEE Transactions on Robotics and Automation, 2001, Vol.17, N°3, pp. 280-289.

[6] K. Hirai, M. Hirose, Y. Haikawa, T. Takenaka. The development of honda humanoid robot. Proc. IEEE Conf. on Robotics and Automation, 1998, pp. 1321-1326.

[7] C. Sabourin, O. Bruneau. Robustness of the dynamic walk of a biped robot subjected to disturbing external forces by using CMAC neural networks. Robotics and Autonomous Systems, 2005, Vol.23, pp. 81-99.

[8] C. Chevallereau, G. Abba, Y. Aoustin, F. Plestan, E.R. Westervelt, C. Canudas-de-Wit, J.W. Grizzle. RABBIT: A testbed for advanced control theory. IEEE Control Systems Magazine, 2003, Vol.23, N°5, pp. 57-79.

[9] http://robot-rabbit.lag.ensieg.inpg.fr/.

[10] http://www.laas.fr/robea/

[11] O. Bruneau, F.B. Ouezdou. Distributed ground/walking robot interactions. Robotica, Cambridge University Press, 1999, Vol.17, N°3, pp. 313-323.

[12] J. S. Albus. A new approach to manipulator control: the Cerebellar Model Articulation Controller (CMAC). Journal of Dynamic Systems, Measurement and Control, (1975), pp. 220--227.

[13] J. S. Albus, Data storage in the cerebellar model articulation controller (CMAC), Journal of Dynamic Systems, Measurement and Control, 1975, pp. 228-233.

[14] W. T. Miller, F. H. Glanz, L. G. Kraft, CMAC: An associative neural network alternative to backpropagation, Proceedings of the IEEE, Special Issue on Neural Networks}, vol.78, N°10, 1990, pp. 1561-1567.

[15] A. L. Kun, T. Miller, The design process of the unified walking controller for the UNH biped, Proc. IEEE Conf. on Humanoid Robots, 2000.

[16] A. Brenbrahim, J. Franklin, Biped dynamic walking using reinforcement learning, Robotics and Autonomous Systems, 1997, Vol.22 , pp. 283-302.

[17] C. Sabourin, O. Bruneau, J-G. Fontaine, Start, stop and transition of velocities on an underactuated bipedal robot without reference trajectories, International Journal of Humanoid Robotics, 2004, Vol.1, N°2, pp. 349-374.