Realization of Primitives by Using SVM for Humanoid Robot Walk Generation

Branislav Borovac, Mirko Raković, and Milutin Nikolić

***Abstract*—To ensure motion of bipedal humanoid robots in unstructured environment, in general, it is not possible to use preprogrammed trajectories. This paper describes the approach to motion generation from primitives that ensures on-line modification of locomotor activity. Primitives represent simple movements that are either reflex or learned. Each primitive has its parameters and constraints that are determined on the basis of the movements capable of performing by a human. The set of all primitives represents the base from which primitives are selected and combined for the purpose of performing the corresponding complex movement. SVM regression is used to learn primitives which are verified in the task of walk generation**

***Index Terms*—Humanoid robot, locomotion, primitives, SVM.**

I. Introduction

THE important features of biped humanoid robots are human-like shape and movements. It is expected from a robot to be able to perform motion the way people do. For successful operation in human environment the robot should be able to walk and move like a human: to climb up stairs, to use doors, to detect and avoid obstacles, walk on uneven terrain, run, etc. In the realization of their movements, most humanoid robots perform motions that are synthesized in advance. In [1-4], the main goal in their realization being to prevent fall, i.e. to preserve dynamic balance, and then, realize the intended movement in a most faithful way. However, for use of humanoids in unstructured environment realization of appropriate motion is main problem.

Humanoid robot motion in unstructured environment is considered in [5, 6]. In [5] is described 3D linear inverted pendulum mode to generate walking pattern. Authors in [6] use predefined zero moment point trajectory and the foot landing position is calculated using short cycle pattern generator.

Motivation for motion generation using primitives is found in [7]. Authors in [7] showed that electrical micro stimulation of same spinal interneuronal region of spinal cord evoked synergistic contractions that generate forces that direct hand limb toward an equilibrium point in space. The collection of the measured forces corresponded to a well-structured spatial pattern (vector field) that was convergent and characterized by single equilibrium point. Second, there are important findings about modular organization of spinal motor systems in the frog spinal cord. These experiments found that only a few distinct types of motor outputs could be evoked by such stimulation. However, when stimulation was applied simultaneously, to two different sites in the spinal cord, each of which when stimulated individually produced a different type of motor output, the resulting motor output was a simple combination of separate motor outputs [7]. Based on these observations it was proposed that complex movements might be produced by the flexible combination of a small number of spinally generated motor patterns.

There are differences among researchers in the use of term primitive movement. For example, in [8-11], the authors used the entire movement as a primitive (overall gait, transition from standing to walking, etc.). In [8] authors used a library of motion primitives where each primitive is a single step. Library of motion primitives actually represents set of different steps. Depending on the requirements and robot state, each time a new primitive is selected from library. In [9] authors describe approach to generate walking primitive databases where each primitive is cyclic walking pattern with different parameters. They also generate a different primitive for transition from one walking pattern to another. In [10] was presented a general framework for learning motor skills which is based on a thorough, analytically understanding of a robot task representation and execution. In [11] was presented an approach for on-line segmentation of whole body human motion observation and learning.

An essential difference between approach in [8-11] and the one proposed in this work is that the on-line motion is formed by a combination of set of basic movements which are such composed to constitute walking pattern and not of the complex movements recorded in advance. Complex movements were decomposed into simple movements, called primitives (e.g. leg stretching, leg bending, hip turning, etc.). The basic idea is to

This work was funded by the Ministry of Science and Technological Development of the Republic of Serbia in part under contract TR35003 and in part under contract III44008 and by Provincial Secretariat for Science and Technological Development under contract 114-451-2116/2011. Some results of this paper were presented at 55th ETRAN Conference, Banja Vrućica, June 6-9, 2011.

B. Borovac is with the Faculty of Technical Sciences, University of Novi Sad, Serbia (e-mail: borovac@uns.ac.rs).

M. Raković is with the Faculty of Technical Sciences, University of Novi Sad, Serbia (e-mail: rakovicm@uns.ac.rs).

M. Nikolić is with the Faculty of Technical Sciences, University of Novi Sad, Serbia (e-mail: milutinn@uns.ac.rs).

enable system to learn to execute on-line each primitive with different parameters (let say, leg bending to different knee and hip angles) from different initial positions. Movement may be continued (if needed) with another primitive (also, on-line selected) to perform complex movement. For example, movement of leg in swing phase during walk consists of leg bending immediately followed by leg stretching. In this work we will be focused to introduce idea of composing complex movements from simple building blocks, and basic explanations of the notion and forms of primitives. Our approach will be illustrated on the example of walk realization.

II. Humanoid Robot Model

Model of humanoid robot consists of the main link to which are connected several open kinematic chains [12-13]. The links are interconnected by simple rotational joints with only one degree of freedom (DOF). Fig. 1 shows the mechanism structure with 46 links. The joints with more DOFs (ankle and hip) are modeled as a series of one-DOF joints, connected by massless links of infinitesimal length (fictitious links). For example, the spherical joint at the left hip is modeled by three simple joints (the orts of rotation axes $\vec{e}_{10}$, $\vec{e}_{11}$ and $\vec{e}_{12}$ are mutually orthogonal), connected by the links having zero length, mass and moment of inertia (fictitious links 10 and 11).

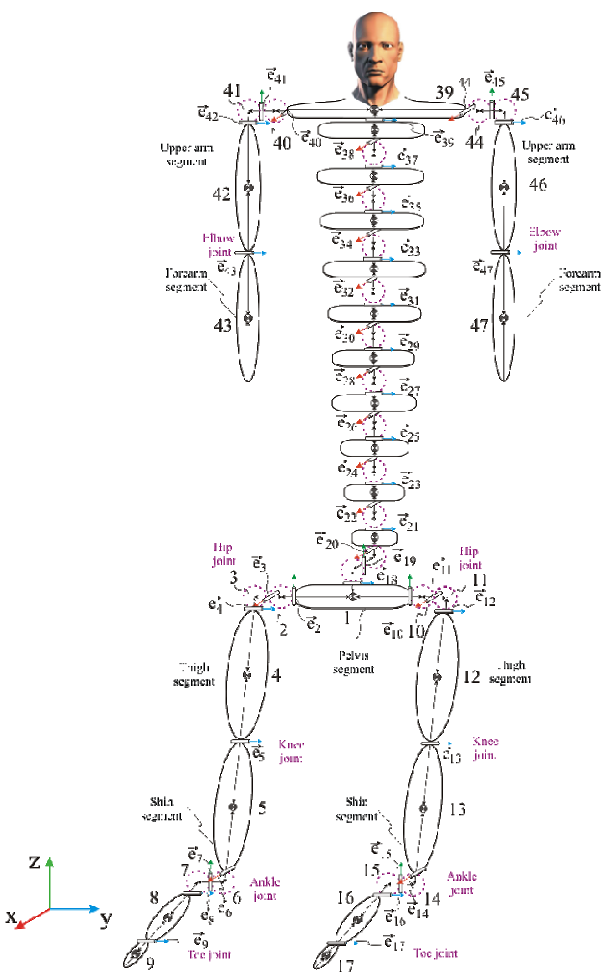


Fig. 1. Mechanical structure of the robot.

The pelvis link was chosen as the base one, and its position and orientation in the space are presented by $\mathbf{X}=[x,y,z,\theta,\varphi,\psi]^T$ three translatory and three angular coordinates. To the pelvis are connected the other kinematic chains, of which the first chain (links 1-9) represents the right leg; the second (links 1, 10-17) stands for the left leg; the third (links 1, 18-43) represents the 10-link trunk and the right arm, whereas the fourth kinematic chain represents the 10-link trunk (the links are interconnected by 2-DOF joints) and the left arm (links 1, 18-39, 44-47). The feet are two-link ones. Between the foot body and toes there exists one DOF.

The motion of each joint is described by one coordinate (q_1, q_2....q_{45}). Taking into account 6 coordinates needed for the positioning of the base link in the space we obtain that the overall number of DOFs of the described mechanism is 51, whereas the system's position is described by the following expression:

$$\mathbf{Q}=[\mathbf{X}^T,\mathbf{q}^T]^T=[x,y,z,\theta,\varphi,\psi,q_1,\cdots,q_{46}]^T$$

Each joint has its own actuator that generates the driving torque τ_j, whereas the first six DOFs of the vector $\mathbf{Q}$, i.e. x,y,z,θ,φ and ψ are unpowered. The vector of torques at the actuated joints is $\boldsymbol{\tau}=[\tau_1,\tau_2,\cdots,\tau_{46}]^T$, and the expanded vector of generalized forces is

$$\mathbf{T}=[0,0,0,0,0,0,\tau_1,\tau_2,\cdots,\tau_{46}]^T.$$

During locomotion the system passes from the single-support to the double-support phase. In the beginning, the leg that performs stepping is not in contact with the ground, the contact is being realized when the stepping is ended. Also, in the course of the contact realization, slipping may occur between the foot and the ground, and during the settling phase the rear leg can separate from the support. Because of that, it is of great importance to model appropriately the foot-ground contact, i.e. describe in the most reliable way all the effects that may arise between the two bodies in contact.

A. *Modeling of the contact*

The two-link foot is rectangular, with a flat contact area, so that for an exact identification of the contact type it suffices to observe only six characteristic points shown in Fig. 2. Four contact points (1-4 in Fig. 2) are the foot body corners, whereas the points 5 and 6 are at the top of the toes. By observing only these six points we can describe all the possible configurations of the foot-ground contact. If three or more points (which are not collinear) are in contact with the support,

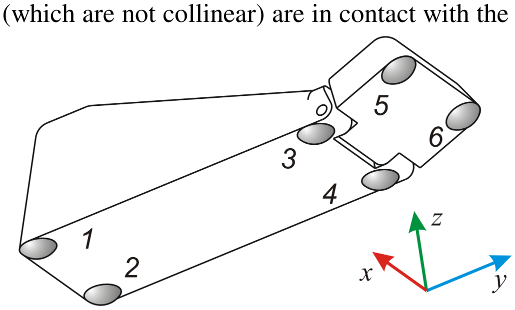


Fig. 2. Two-link foot with marked contact points.

the contact is planar; if two points are involved, it is a linear and, finally, there may exist one-point contact.

Since there are two feet, the total number of contact points that are to be observed is 12. However, not all the points are always in contact with the ground. To model the contact of a particular point with the ground use was made of the model of rigid body with a visco-elastic layer. [14-16].

B. *Overall system model*

By uniting the model of the robot's mechanical structure and of the elastic foot-ground contacts, the overall system can be described by the following set of differential equations:

$$\mathbf{H}\ddot{\mathbf{Q}} + \mathbf{h_0} = \mathbf{T} + \sum_{i \in S} \mathbf{J}^{i\,T} \begin{bmatrix} \mathbf{F}^i \\ \boldsymbol{\delta}^i \times \mathbf{F}^i \end{bmatrix} \quad (1)$$

where $\ddot{\mathbf{Q}}$ represents the vector of generalized accelerations of the mechanical system; $\mathbf{H}$ is the system inertia matrix; $\mathbf{h_0}$ is the column vector of which includes moments induced by Coriolis, centrifugal and gravitational forces. $\mathbf{T}$ is the column vector of driving torques; S is the set of points in contact with the ground; $\boldsymbol{\delta}^i$ and $\mathbf{F}^i$ are the deformation and contact force at the i-th contact point, respectively; $\mathbf{J}^i$ is the Jacobi matrix calculated for the i-th contact point. The contact force and deformation derivative for each contact point are calculated by according to the contact model. [15-16]

III. Motion Decomposition and Synthesis

Motion can be considered as composed from set of basic movements which can be learned and easily combined. In this chapter we will show how walk on flat surface can be combined from basic primitives.

A. *Primitives- basic*

The term primitive stands for a simple reflex or learned movement that a human or robot is capable to realize [17]. A primitive itself should be simple in order it could be easily performed by simultaneous and synchronized action of one or more joints and easily combined with the other primitives. Selection of movement to be adopted as a primitive is not unique but is based on our expertise. This implies that certain primitive can be included in different complex movements.

Each primitive is parameterized and has the following parameters: intensity of the movement in the span of 0-1 (which determines the extent to which, for example, a leg is to be bent or stretched), time instant from which the primitive execution should be started, and duration of the primitive execution. Each of the primitives is realized by activating one or more DOFs. Thus, for example, the primitive for bending the leg in swing phase involves activation of the joints at the hip, knee and ankle of the swing leg. It is worth noting that the primitives can be changed very easily in the sense of varying the range of the changing angle (by multiplying with a factor smaller than 1), as well as by changing the duration of its realization (faster or slower movement execution). Also, it is possible to change parameters of particular primitive during its execution.

Considering kinematics, each primitive consists of of simple “s” shape motion at each activated joint. However, motion is realized by applying appropriate driving torques at joints, and for same shape of movement (with different starting points) driving torques at joints varies. Exact driving torque depends of “gravity contribution” to joint load which vary with different position of leg in space etc. This means that shape of motor control variable do not correspond to movement shape.

B. *Walk composed from primitives*

In this section is presented horizontal flat surface walking motion entirely synthesized using primitives. At the initial moment robot is standing still with both feet in contact with ground (double support phase). When motion starts, robot first transfer complete weight to right leg and started first half-step by lifting up left leg (single support phase) and moving it to front position. When left leg touched ground double-support phase was established again. Walking process continued and left leg become support leg, right leg was deployed from ground (right leg become leg in swing phase) and it was transferred from back to front position.

In Fig. 3 are shown stick diagrams of synthesized motion.

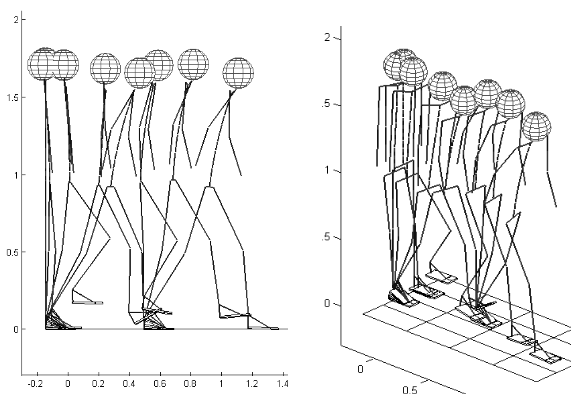


Fig. 3. Side view (left) and perspective view (right) at the stick diagram representing starting walking.

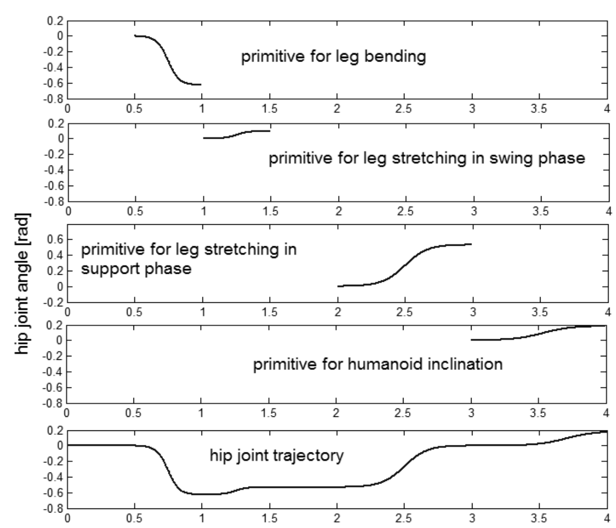


Fig. 4. Diagrams of composing the motion at the at the hip joint of left leg by superimposing primitives.

Primitives were applied at each joint. To obtain highly anthropomorphic walk it is very important to define appropriate starting and ending points for each primitive and motion intensity.

In Fig. 4. Is illustrate resulting trajectory at the hip joint of left leg for rotation about axis orthogonal on the motion direction. The following primitives were applied. Leg bending was applied with intensity 0.6, followed by leg in swing phase stretching with intensity 0.15. Then, for next 0.5s no additional primitives were applied and joint keep its current position. Then, leg at swing phase come to contact with ground and it become support leg and primitive for support leg stretching with intensity 1 was applied. Last primitive applied is inclination of overall system with intensity 0.35. Complete movement duration is 4s.

This example is given just to explain and illustrate basic idea of composing movements. However, primitives are not composed by selecting motion for each joint separately, but by selecting motion „as a whole“. For example, motion of leg in swing phase is composed by realization of two primitives: leg bending followed by leg stretching. Each primitive ensures synchronized motion of all joints involved.

IV. Motion Learning

Simple motion of human limbs (for example, 3D human hand motion along straight line) requires very complex, simultaneous and well synchronized change of joints angles. Task is becoming even more complex when it is clear that it has to be synthesized and performed on-line on the basis of current state of the robot-environment interaction and can't be prepared in advance.

Having in mind findings reported in [7] we believe that human learn some basic motions and when needed it just have to be recalled and replayed. Another important point of [7] is that same stimuli drive limb to same point irrespective of its initial position (whole vector field is formed). And finally, those vector fields are additive. We believe approach with similar characteristics is needed for robots, too.

It have to be underlined difference between kinematic and dynamic composition of primitives. In case of kinematic composition just shape of movement have to be achieved while dynamics composition suppose learning of driving torques to achieve desired movement. Let us explain approach we advocate on the example of bending leg in swing phase. The first characteristic of this movement is that spatial trajectory of ankle is approximately straight line during normal walking without obstacles. Bending intensity and bending speed is choice of human, and it is possible to realize movement with different motion parameters and from different initial positions.

In Fig. 5 are illustrated two different examples of leg bending. In both cases small circles denote initial and terminal position of swing leg ankle (terminal position is defined by bending intensity). If bending intensity is larger ankle terminal position is higher.

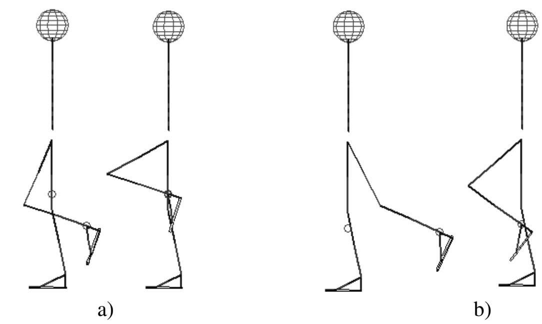


Fig. 5. Examples of leg bending with different starting positions and different intensities; a) leg bending intensity is 91%, b) leg bending intensity is 50%

For leg bending realization it is necessary to define set of inputs and outputs. As inputs were adopted: bending intensity, movement velocity (specified by movement duration), current velocity of ankle, current values of angles and current angular velocities at knee and hip (3 DoFs - at hip rotation is about x-axis (joint unit vector is parallel to walking direction) and y-axis (joint unit vector is orthogonal to walking direction), and at knee rotation is about y axis). Outputs are driving torques at hip and knee. At starting position leg is not moving and driving torques has to be such defined to ensure that (just to compensate gravity). Then, torques should be such to ensure linear ankle trajectory whose terminal point is defined by bending intensity. When ankle terminal point is reached torques should be such to keep leg at this position.

If movement velocity increase time needed for motion realization (t_{prim}) is shorter, but intensity of maximal ankle velocity is increased. On the basis of:

$$\dot{\mathbf{q}}_L = \mathbf{J}_L^{-1} \cdot \mathbf{v}_{skz} \tag{2}$$

angular velocities $\dot{\mathbf{q}}_L$ at knee and hip joint can be determined. $\mathbf{v}_{skz}$ denotes desired ankle velocity, while values of Jacobian matrix represent relationship between leg's joints angular velocities and ankle linear velocity. From (1) we see that driving torques can be calculated as:

$$\mathbf{T} = \mathbf{H}\ddot{\mathbf{Q}} + \mathbf{h_0} - \sum_{i \in S} \mathbf{J}^{i\,T} \begin{bmatrix} \mathbf{F}^i \\ \boldsymbol{\delta}^i \times \mathbf{F}^i \end{bmatrix} \tag{3}$$

To use SVM regression for obtaining necessary driving torques we need to ensure proper training set. Let us remind once more that inputs for primitives (and also for SVM) are bending intensity, movement velocity (specified by movement duration), current ankle velocity, current leg posture, i.e. current values of angles and current angular velocities of performing leg, while outputs are driving torques (Fig. 6).

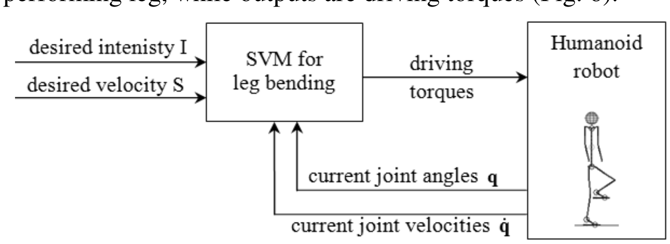


Fig. 6. Illustration of control scheme for leg bending primitive realization.

Because bending can be performed from any possible initial position training set should span, as much as possible, over initial postures which may appear. Training set should also span whole range of all possible bending intensities and velocities. First, leg initial posture, bending intensity and movement velocity are randomly determined.

The procedure of determining the input and output quantities for training data set is as follows:

1. Starting posture of the robot's leg is determined by random selection of hip and knee joint angles from predefined range.
2. Intensity (I) of leg bending and movement duration (S) are also selected randomly.
3. Desired velocity profile of ankle is calculated.
4. Then, procedure for driving torques computation at each time instant is following:
   a. Desired ankle velocity profile is known. In each iteration corresponding angular velocities at leg's joints are calculated from (2).
   b. After that, at each joint are calculated angle and angular acceleration.
   c. Since whole system state is known (angles, angular velocities and accelerations at all joints) driving torques are calculated from (3).

   In this way all input and output data for SVM training set for this time instant are specified. Then, procedure continue for next time instant
5. Procedure is repeated till ankle is sufficiently close to its terminal position, and when ankle velocity is sufficiently low. Then, procedure for this movement is stopped, new ankle starting position is randomly selected (target point is defined by intensity of leg bending) and steps 1-4 are repeated.
6. After sufficient number of performed movements, the procedure is stopped.

In this way the leg bending is simulated from the arbitrary starting point to the target point defined by intensity of leg bending. For each time instant, the values of all input and output quantities needed for SVM training are obtained. The input vector $[I\ S\ \mathbf{v}_{skz}\ \ \mathbf{q}_L\ \ \dot{\mathbf{q}}_L\ ]^T$ for the training set is of dimension 11 whereas the dimension of the output vector $\boldsymbol{\tau}$ is 3.

A. SVM Regression

Since, we are using SVMs to calculate driving torques for primitives it is necessary to briefly describe what SVM represents and how it works.

There are a number of algorithms for approximating the function for establishing the unknown interdependence between the input ($\mathbf{x}$) and output (y) data, but an ever-arising question is how good is the approximation of the function $f_a(\mathbf{x})$. In determining the approximation function, it is necessary to minimize some of the error functions. The majority of the algorithms for the function approximation minimize the empirical error.

With the function approximation algorithms that minimize only the empirical error, there arises the problem of a large generalization error. The problem appears when the training set is small compared to the number of different data that can appear at the input. Structural Risk Minimization (SRM) [19] is a new technique of the statistical learning theory, which apart from minimizing the empirical errors, also minimizes the generalization errors (elements of the weight matrix $\mathbf{w}$). Hence, it follows that the structural error will be minimized by minimizing function of the form:

$$R = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{l}\left|y_i - f_a(\mathbf{x}_i, \mathbf{w})\right|_\varepsilon \tag{4}$$

In (4), the error function with the ε- insensitivity zone was used as the norm. The parameter C is the penalty parameter which determines the extent to which the empirical error is penalized relatively to the penalization of the large values in the weighting matrix. Network input is denoted by $\mathbf{x}$, and desired output is denoted by y. Approximating function is denoted by $f_a(\mathbf{x}, \mathbf{w})$ and it has to be chosen in advance. Since case considered is highly nonlinear, for approximating function we have chosen radial basis function network (RBF network) with Gaussian kernel, for which output is calculated by:

$$f_a(\mathbf{x}, \mathbf{w}) = \sum_{i=1}^{N} w_i \exp(-\gamma\|\mathbf{x} - \mathbf{c}_i\|^2) + \rho$$

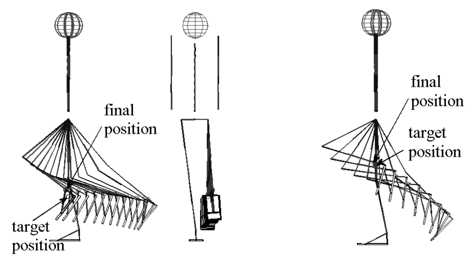
The nonlinear SVM regressions (minimization of (4)) determines the elements of the weight matrix $\mathbf{w}$ and bias ρ. During the SVM training, support vectors ($\mathbf{c}_i$) are chosen from set of input training data. Design parameter ρ defines the shape of RBFs, and it is experimentally chosen to minimize VC-dimension, which provides good generalization.

B. Leg bending

According to described procedure for collecting training data set simulation was performed which lasted 100s. In this period 178 different initial postures were generated and same number of leg bending were performed. Training set has been formed on the basis of data from every sampling period (in this case it was 1 ms) and it was collected 100 000 input-output pairs. As a training set for SVM was selected just 10% of all collected data[1]. Penalty parameter C and ε zone of insensitivity has been selected as 100 and 0.1. Those parameters are specific for each particular task and values were obtained by trial-and-error till satisfactory response was obtained.

After SVM training has been completed performance testing was performed by simulation. On the basis of specified leg bending intensity, movement duration, current ankle velocity and current leg posture and joints angular velocities (at knee and hip) driving torques are obtained from SVM and applied on robot. Torques drive system to new state, new torques are obtained, and procedure repeats till movement is completed. In

[1] There are three reasons to select just 10% of available data. First one is that for training set quadratic programming problem have to be solved. This requires large memory resources and training is time consuming. The second reason is that by selecting such size of training set problem of over fitting is avoided. Third reason is that such training set size was proven to be enough large for successful training for such kind of problems [17,18].



a) b)

Fig. 7. Humanoid robot performing leg bending: a) bending intensity 41%, parameter S is 45% b) bending intensity 92%, parameter S is is 70%

Fig. 7 are shown two examples of leg bending performed using trained SVM. Starting posture (angles at hip and knee) are selected to correspond approximately to posture of swing leg after deployment from ground during walking. Bending intensity and movement duration are defined randomly.

In example sketched in Fig 7.a) initial angles intensities (hip rotation about *x* and *y* axis, and knee joint rotation about *y* axis) were 0^o, 40^o and 10^o, respectively. Bending intensity was 41%, and motion duration (parameter S) was 45%. As a consequence, movement was performed for t_{prim}=0.69s. Distance between target ankle position (defined by bending intensity) and achieved position was 6.3 cm.

In second example (sketched in Fig. 7.b)) initial angles intensities were 0^o, 30^o i 15^o, respectively. Bending intensity was 92%, and motion duration (parameter S) was 70%. Motion was performed for t_{prim}=0.46s. Distance between target ankle position (defined by bending intensity) and achieved position was 2.7cm.

C. Walk generation

To perform complete walk movement of the leg in swing phase leg bending have to be continued by leg stretching. Also appropriate primitives have to be realized in supporting leg. In Fig. 8 is shown the stick diagram of walk composed using learned primitives by applying driving torques obtained from SVM.

At the beginning, robot stands still with both feet in contact with the ground. When walk starts, first a right leg bending is performed followed by leg stretching. Trunk is moved forward and robot is inclined forward. When right leg touches ground, the double support phase starts. After that the left leg bending is performed followed by leg stretching. Two half step simulation lasted for 2.3 s.

V. CONCLUSION

In this paper a novel approach of on-line synthesize of complex motions from basic movements (called primitives) is presented. Basic characteristic of proposed method is that, once trained, system is able to perform primitive from different starting points with different motion parameters. Also, new primitive can be added when previous is finished (or even before movement is over) without time consuming calculations. This enable human-like motion and on-line motion synthesis of humanoids in unstructured environment.

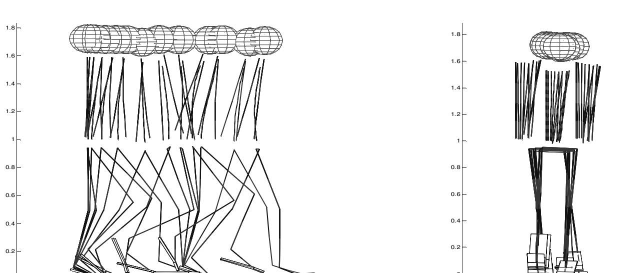


Fig. 8. Humanoid robot walk composed from primitives.

REFERENCES

[1] M. Vukobratović, "How to control the artificial anthropomorphic systems", *IEEE Trans. on System, Man and Cybernetics*, SMC-3, pp. 497-507, 1973

[2] M. Vukobratović, *Legged locomotion lystems and anthropomorphic mechanisms*, Mihajlo Pupin Institute, Belgrade 1975, also published in Japanese, Nikkan Shimbun Ltd. Tokyo, in Russian, "MIR", Moscow, 1976, in Chinese, Beijing 1983

[3] M. Vukobratović, B. Borovac, D. Surla, D. Stokić, *Biped locomotion – Dynamics, stability, control and application*, Springer-Verlag, Berlin, 1990

[4] M. Vukobratović, B. Borovac, "Zero-Moment Point- Thirty five years of its life", *Int. Jour. of Humanoid Robotics*, Vol. 1, No. 1, pp. 157-173, 2004

[5] A Realtime Pattern Generator for Biped Walking Shuuji Kajita, Fumio Kanehiro, Kenji Kaneko, Kiyoshi Fujiwara, Kazuhito Yokoi and Hirohisa Hirukawa

[6] Online Walking Control System for Humanoids with Short Cycle Pattern Generation K. Nishiwaki S. Kagami

[7] F.A. Mussa-Ivaldi, S.F. Gisztler, E. Bizzi, "Linear combinations of primitives in vertebrate motor control", *Proc. Natl. Acad. Sci. 91*, pp. 7534-7538, 1994

[8] K. Hauser, T. Bretl, J.C. Latombe, "Using motion primitives in probabilistic sample-based planning for humanoid robots" *Algorithmic foundation of robotics VII*, Vol. 47 Springer Berlin / Heidelberg , pp. 507-522, 2008

[9] J. Peters, S. Schaal, "Policy learning for motor skills*", Neural Information Processing*, Springer Berlin / Heidelberg , Vol. 4895, pp. 233-242, 2008

[10] J. Denk, G. Schmidt, "Synthesis of walking primitive databases for biped robots in 3D-environments", *Proc of the IEEE Int. Conf. on Robotics and Automation*, Vol. 1, pp. 1343–1349, Taipei, Taiwan, 2003

[11] D. Kulić, Y. Nakamura, "Incremental learning and memory consolidation of whole body motion patterns", *Int. Conf. on Epigenetic Robotics*, pp. 61–68, 2008

[12] M. Vukobratović, V. Potkonjak, B. Borovac, K. Babković, "Simulation model of general human and humanoid motion" *Multibody System Dynamics*, Vol. 17, No 1, pp 71-96, 2007

[13] M. Vukobratović, B. Borovac, M. Raković, V. Potkonjak, M. Milinović, "On Some Aspects of Humanoid Robots Gait Synthesis and Control at Small Disturbances", *Int. Jour. of Humanoid Robotics*, Vol. 5, No 1, pp 119-156, 2008

[14] P. Song, P. Kraus, V. Kumar, P. Dupont, "Analysis of Rigid-Body Dynamic Models for Simulation of Systems With Frictional Contacts", *Journal of Applied Mechanics*, Vol. 68, No 1, pp 118-128, 2001

[15] S. Goyal S, E. M. Pinson, F. W. Sinden, "Simulation of dynamics of interacting rigid bodies including friction I: General problem and contact model", *Engineering with Computers*, Vol. 10, No 3, pp 162-174, 1994

[16] B. Borovac, M. Nikolić, M. Raković M. "How to compensate for the disturbances that jeopardize dynamic balance of a humanoid robot" *International Journal of Humanoid Robotics*, In Print 2011.

[17] B. Borovac, M. Raković, M. Nikolić, "One Approach for On-line Generation of Motion of Humanoid Robots Using Primitives", *55th ETRAN*, Teslić, Republika Srpska, 2011

[18] M. Raković, M. Nikolić, B. Borovac, "Humanoid Robot Reaching Task Using Support Vector Machine", *Procc of Int. Conf. Eurobot 2011*, Prague, Czech Republic, 2011

[19] Kecman V. *"Learning and Soft Computing: Support Vector Machines, Neural Networks, and Fuzzy Logic Models"*, The MIT Press, Cambridge Massachusetts, 2001