Vol. 43 No. 4 IPSJ Journal Apr. 2002

Regular Paper

A New Gait Optimization Approach Based on Genetic Algorithm for Walking Biped Robots and a Neural Network Implementation

Genci Capi,†1 Yasuo Nasu,†2 Leonard Barolli,†3
Kazuhisa Mitobe,†2 Mitsuhiro Yamano†2 and Kenro Takeda†4

In this paper, a Genetic Algorithm gait synthesis method for biped robots is proposed. The gait synthesis during walking is analyzed based on minimum consumed energy and minimum torque change. Except gait synthesis optimization, we also consider the stability, minimum torque change cost function, and the real time implementation. The stability is verified through the Zero Moment Point concept. For the real time implementation, a Radial Basis Function Neural Network, that is taught based upon Genetic Algorithm results, is considered. In this paper, we present the Neural Network results where the input variable is the step length. Simulation results and experiments show that proposed method has a good performance.

1. Introduction

The humanoid robot shape is very similar with that of humans. For this reason, they can substitute humans in home works, such as helping disabled persons, working in hazardous environments, etc. Autonomous humanoid robots are well fitted in these working places. But, they are inevitably restricted to a limited amount of energy supply. It would therefore be advantageous to consider the minimum energy consumption, when cyclic movements like walking are involved.

From the viewpoint of energy consumption, one factor that has a great influence is the gait synthesis. In most of the previous papers related to biped robots [1),2)], the angle trajectories of the leg part are prescribed based on data taken from humans. The upper body motion is calculated in order to have the Zero Moment Point (ZMP) inside the sole region. Some efforts have been placed to analyze the effect of gait synthesis on consumed energy. In Refs. 3), 4), the minimum consumed energy gait synthesis during walking is discussed. The body mass is considered concentrated on the hip of biped robot [3)]. In Ref. 4), the body link is restricted to the vertical position, the body forward velocity is considered to be constant and the tip motion of swing leg is constrained to follow sinusoidal functions. The effect of walking velocity and step length on the consumed energy is discussed in Ref. 5), by using a variational technique to minimize the cost function. However, in all these approaches related to optimal gait of biped robots, the stability and real time implementation are not considered.

In this paper, we present a Genetic Algorithm (GA) gait synthesis method for biped robots during walking based on Consumed Energy (CE) and Torque Change (TC). GA has been known to be robust for search and optimization problems [6)]. It has been used to solve difficult problems with objective functions that do not possess properties such as continuity, differentiability, etc. When solving for optimal gaits, some constrains must be considered. GA, in difference from other optimization methods, makes easy handling the constraints by using the penalty function vector, which transforms a constrained problem to an unconstrained one. In our work, the most important constraint is the stability, which is verified through the ZMP concept. In this paper, by comparing our method with Refs. 3), 4), we show that stability must be considered during the optimal gait generation. Simulations and experiments with the “Bonten-Maru I” humanoid robot show that GA generates a stable and smooth motion.

For real time implementation of their work, in Ref. 3), the authors suggest creating a database of pre-computed optimal gaits. This can generate the angle trajectories only for the step lengths and step times, which are included in the database. In order to cover the whole interval of pre-computed optimal gaits, we con-

†1 ATR Human Information Science Laboratories
†2 Faculty of Engineering, Yamagata University
†3 Faculty of Engineering, Saitama Institute of Technology (SIT)
†4 Tohoku RICOH Company

sider teaching a Radial Basis Function Neural Network (RBFNN) based on the GA results. We also tried back-propagation and perceptron Neural Networks, but the RBFNN gave the best results and the mean square error (mse) was minimal. The RBFNN gives good results for the approximation problems. It is capable of implementing arbitrary nonlinear transformations of the input space. When applied to supervised learning with linear models (that is if the basis functions are fixed in position and size), the weights can be derived optimally by solving a set of equations. This is a lot faster than training the network. In Ref. 7), the real time implementation for going up-stairs is considered. While, in this paper, the RBFNN results for walking are presented, where the input variable of RBFNN is the step length. Simulations show good results generated by RBFNN in a very short time.

The paper is organized as follows. Section 2 deals with the gait and body motion. In Section 3, the problem formulation and proposed method are discussed. Boundary conditions and GA variables are treated in Section 4. Simulation and experimental results are given in Section 5. A RBFNN real time implementation is presented in Section 6. Finally, conclusions are given in Section 7.

2. ZMP and Biped Model

During walking, the humanoid robot arms will be fixed on the chest. Therefore, it can be considered as a five-link biped robot in the saggital plane, as shown in **Fig. 1**. In the figure, m_i and θ_i are defined as mass and absolute angle of link i. The biped robot motion is considered to be composed from a single support phase and an instantaneous double support phase. The friction force between the robot's feet and the ground is considered to be great enough to prevent sliding. During the single support phase, the ZMP must be within the sole length, so the contact between the foot and the ground will remain. The ZMP is the point on the walking ground surface at which the horizontal components of the resultant moment generated by active forces and moments are equal to zero. In this paper, we calculate the ZMP position by considering the link mass concentrated at one point [2], as follows:

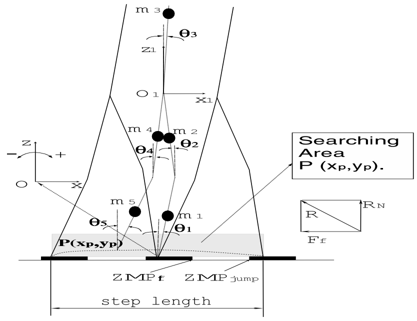


Fig. 1 Five-link biped robot.

$$X_{ZMP} = \frac{\sum_{i=1}^{5} m_i(\ddot{z}_i + g_z)x_i}{\sum_{i=1}^{5} m_i(\ddot{z}_i + g_z)} - \frac{\sum_{i=1}^{5} m_i \ddot{x}_i z_i}{\sum_{i=1}^{5} m_i(\ddot{z}_i + g_z)}, \qquad (1)$$

where m_i is the mass of particle i, x_i and z_i are the coordinates of the mass particle i with respect to the OXZ coordinate system, $\ddot{x}_i$ and $\ddot{z}_i$ are the accelerations of the mass particle i with respect to the OXZ coordinate system and g_z is the gravitation acceleration. To have a stable walking motion, when the swing foot touches the ground, the ZMP must jump in its sole. The body link acceleration is considered to realize it. To have an easier relative motion of the body, the coordinate system from the ankle joint of supporting leg is moved transitionally to the waist of the robot ($O_1X_1Z_1$). Referring to the new coordinate system, the ZMP position is written as:

$$\bar{X}_{ZMP} = \frac{\sum_{i=1}^{5} m_i(\ddot{\bar{z}}_i + \ddot{z}_w + g_z)\bar{x}_i}{\sum_{i=1}^{5} m_i(\ddot{\bar{z}}_i + \ddot{z}_w + g_z)} - \frac{\sum_{i=1}^{5} m_i(\ddot{\bar{x}}_i + \ddot{x}_w)(\bar{z}_i + z_w)}{\sum_{i=1}^{5} m_i(\ddot{\bar{z}}_i + \ddot{z}_w + g_z)}, \quad (2)$$

where x_w and z_w are the coordinates of the waist with respect to the OXZ coordinate system, $\bar{x}_i$ and $\bar{z}_i$ are the coordinates of mass particle i with respect to the $O_1X_1Z_1$ coordinate system, $\ddot{\bar{x}}_i$ and $\ddot{\bar{z}}_i$ are the accelerations of mass particle i with respect to the $O_1X_1Z_1$ coordinate system.

Based on Eq. (2), if the positions $\bar{x}_i$, $\bar{z}_i$, and accelerations $\ddot{\bar{x}}_i$, $\ddot{\bar{z}}_i$ of the leg part (i = 1, 2, 4, 5), the body angle θ_3, and the body angular velocity $\dot{\theta}_3$, are known, then because $\ddot{\bar{x}}_3$ and $\ddot{\bar{z}}_3$ are functions of $l_3, \theta_3, \dot{\theta}_3, \ddot{\theta}_3$, it is easy to calculate the body angular acceleration based on the ZMP position. Let (0) and (f) be the indexes at the beginning and at the end of the step, respectively. At the beginning of the step, $\ddot{\theta}_{30}$ causes the ZMP to be in the ZMP_{jump} position. At the end of the step, the angular acceleration, $\ddot{\theta}_{3f}$, is calculated in order to have the ZMP at the ZMP_f position. Thus, the difference between $\ddot{\theta}_{3f}$ and $\ddot{\theta}_{30}$ is minimal. Also, the torque necessary to change the acceleration of the body link will be minimal.

3. Problem Formulation and Proposed Method

3.1 Problem Formulation

The problem consists of finding the joint angle trajectories, to connect the first and last posture of biped robot for which the CE or TC is minimal. It can be assumed that the energy to control the position of robot is proportional to the integration of square of torque with respect to time. Because the joints of manipulator are driven by torque, then the unit of torque, N_m, is equal to the unit of energy, joule. So, the cost function, J, can be defined as the following expression:

$$J = \frac{1}{2}\left(\int_0^{t_f} \tau^T \tau dt + \Delta\tau_{jump}^2 \Delta t + \int_0^{t_f} C dt\right) \quad (3)$$

where: t_f is the step time, τ is the torque vector, and $\Delta\tau_{jump}$ and Δt are the addition torque applied to the body link to cause the ZMP to jump and its duration time, and C is the constraint function, given as follows:

$$C = \begin{cases} 0 & \text{if the constraints are satisfied,} \\ c_i & \text{if the constraints are not satisfied,} \end{cases}$$

c denotes the penalty function vector.

We consider the following constraints for our system.

- The motion to be stable or the ZMP to be within the sole length.
- The distance between the hip and ankle joint of swing leg must not be longer then the length of extended leg.
- The swing foot must not touch the ground prematurely.

The results generated for minimum CE cost function, are compared with the angle trajectories that minimize the rate of change of the torque[8]. The cost function is as follows:

$$J_{torque\ change} = \frac{1}{2}\left(\int_0^{t_f} \left(\frac{d\tau}{dt}\right)^T \left(\frac{d\tau}{dt}\right) dt\right) + \frac{1}{2}\left(\left(\frac{\Delta\tau_{jump}}{\Delta t}\right)^2 + \int_0^{t_f} C dt\right). \quad (4)$$

3.2 Proposed Method

The proposed method is based on the GA. Therefore, in following, a brief introduction of GA is given. The GA is a search algorithm based on the mechanics of natural selection and population genetics. The search mechanism is based on the interaction between individuals and the natural environment. GA comprises a set of individuals (the population) and a set of biologically inspired operators (the genetic operators). The individuals have genes, which are the potential solutions for the problem. The genetic operators are crossover and mutation. GA generates a sequence of populations by using genetic operators among individuals. Only the most suited individuals in a population can survive and generate offspring, thus transmitting their biological heredity to the new generation. The main steps of GA are:

(1) Supply a population P_0 of N individuals and respective function values;
(2) $i \leftarrow 1$;
(3) $P_i' \leftarrow selection\_function\ (P_{i-1})$;
(4) $P_i \leftarrow reproduction\_function\ (P_i')$;
(5) Evaluate (P_i);
(6) $i \leftarrow i + 1$;
(7) Repeat step 3 until termination;
(8) Print out the best solution.

The block diagram of proposed method is presented in **Fig. 2**. Based on initial conditions and range of searching variables, an initial population is generated. Every angle trajectory is

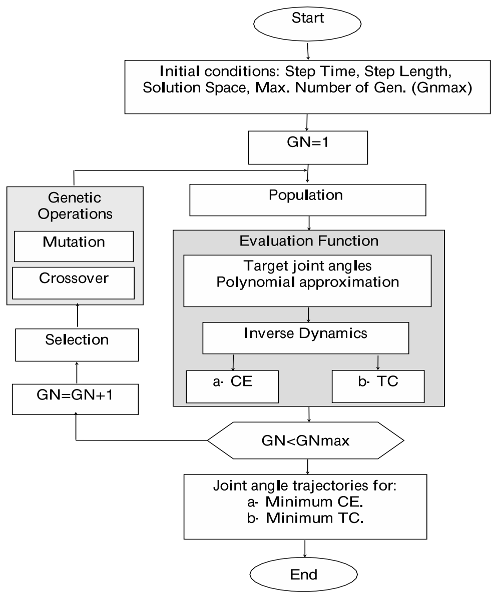


Fig. 2 Block diagram of the proposed method.

presented as a polynomial of time. Its degree is determined based on the number of angle trajectory constraints and the coefficients are calculated to satisfy these constraints. The torque vector is calculated from the inverse dynamics of five-link biped robot [9] as follows:

$$J(\theta)\ddot{\theta} + X(\theta)\dot{\theta}^2 + Y\dot{\theta} + Z(\theta) = \tau \qquad (5)$$

where J (θ) = [5 × 5] is the mass matrix, X (θ) = [5 × 5] is the matrix of centrifugal coefficients, Y = [5 × 5] is the matrix of Coriolis coefficients, Z (θ) = [5 × 1] is the vector of gravity terms, τ = [5 × 1] is the generalized torque vector, and θ, $\dot{\theta}$, $\ddot{\theta}$ are [5 × 1] vectors of joint angles, joint angular velocities and joint angular accelerations, respectively.

According to Eqs. (3) and (4), the cost function is calculated for minimum CE and minimum TC, respectively. The value of cost function is attached to every individual of the population. GA moves from generation to generation, selecting parents and producing offspring until the termination criterion (maximum number of generations GN_{max}) is met. Based on GA results, the gait synthesis is generated for minimum CE and minimum TC, respectively.

4. Boundary Conditions and GA Variables

To have a continuous periodic motion, the biped robot posture is considered to be the same at the beginning and at the end of the step. Therefore, the following relations must be satisfied:

$$\theta_{10} = \theta_{5f}, \theta_{20} = \theta_{4f}, \theta_{1f} = \theta_{50}, \theta_{2f} = \theta_{40}, \\ \theta_{30} = \theta_{3f}. \qquad (6)$$

In order to find the best posture at the beginning of the step, the optimum values of θ_{10}, θ_{20} and θ_{30} must be determined by GA. For a given step length, it is easy to calculate θ_{40} and θ_{50}, based on the biped robot inverse kinematics. When referring to Fig. 1, it is clear that links 1, 2, 4 at the beginning of the step and links 2, 4, 5 at the end of the step, change the direction of rotation. Therefore, we can write:

$$\dot{\theta}_{10} = \dot{\theta}_{20} = \dot{\theta}_{40} = \dot{\theta}_{2f} = \dot{\theta}_{4f} = \dot{\theta}_{5f} = 0. \qquad (7)$$

The angular velocity of link 1 at the end of the step and link 5 at the beginning of the step is considered to be the same. This can be written in the form $\dot{\theta}_{1f} = \dot{\theta}_{50}$. In order to find the best value of angular velocity, we consider it as one variable of GA, because the rotation direction of these links does not change. GA will determine the optimal value of the body link angular velocity, which is considered to be the same at the beginning and at the end of the step. The following relations are considered for the angular acceleration:

$$\ddot{\theta}_{10} = \ddot{\theta}_{5f}, \ddot{\theta}_{20} = \ddot{\theta}_{4f}, \ddot{\theta}_{1f} = \ddot{\theta}_{50}, \ddot{\theta}_{2f} = \ddot{\theta}_{40}. (8)$$

In this way, during the instantaneous double support phase, we don't need to apply an extra torque to change the link angular accelerations. To find the upper body angle trajectory, an intermediate angle, θ_{3p}, and its passing time, t_3, are considered as GA variables. To determine the angle trajectories of swing leg, the coordinates of an intermediate point, $P(x_p, z_p)$, and their passing time, t_p, are also considered as GA variables. The searching area for this point is shown in Fig. 1. Based on the number of constraints, the degree of time polynomial for θ_1, θ_2, θ_3, θ_4 and θ_5 are 3, 3, 7, 6 and 6, respectively.

5. Simulation Results

In the simulations, we use the parameters of the "Bonten-Maru I" humanoid robot, which is developed in our laboratory. The robot is shown in **Fig. 3** and the parameter values are presented in **Table 1**. The "Bonten-Maru I" is 1.2 m high, each leg has 6 degrees of freedom and is composed by three segments: upper leg,

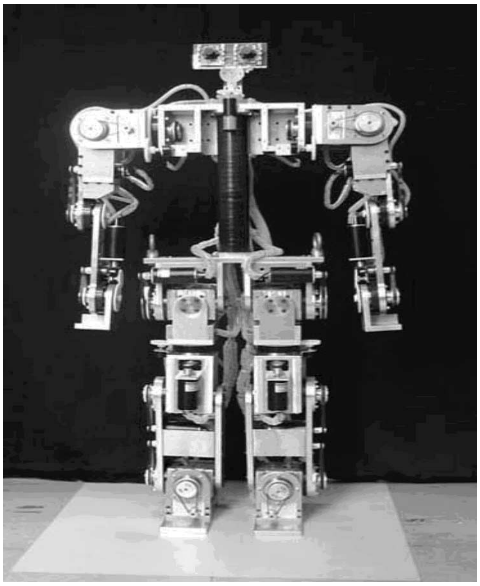


Fig. 3 "Bonten-Maru I".

Table 1 "Bonten-Maru I" parameters.

	Body	Lower Leg	Uper Leg	Lower Leg + Foot
Mass [kg]	12	2.93	3.89	4.09
Inertia [kg m^2]	0.19	0.014	0.002	0.017
Length [m]	0.3	0.2	0.204	0.284
CoM Dist. [m]	0.3	0.09	0.1	0.136

Table 2 Functions and parameters of GA.

Function Name	Parameters
Arithmetic Crossover	2
Heuristic Crossover	[2 3]
Simple Crossover	2
Uniform Mutation	4
Non-Uniform Mutation	[4 GN_{max} 3]
Multi-Non-Uniform Mutation	[6 GN_{max} 5]
Boundary Mutation	4
Normalized Geometric Selection	0.08

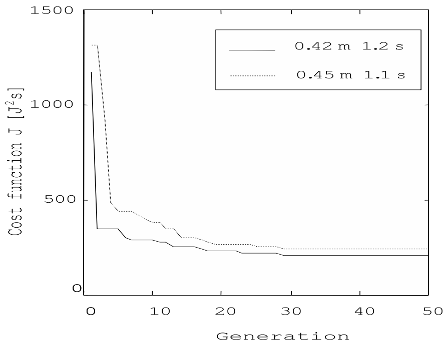


Fig. 4 Cost function J vs. generations.

lower leg and the foot. The foot length is 0.18 m. A DC servomotor actuates each joint. The control platform is based on Common Object Request Broker Architecture (CORBA), which allows an easy updating and addition of new modules.

For optimization of the cost function, a real-value GA was employed in conjunction with the selection, mutation and crossover operators [10]. Many experiments comparing the real value and binary GA have shown that the real value GA generates better results in terms of solution quality and CPU time. To ensure a good result of the optimization problem, the best GA parameters are determined by extensive simulations that we have performed, as shown in **Table 2**. The maximum number of generations is used as termination function. GA converges within 40 generations (see **Fig. 4**). In **Fig. 5** is presented the convergence of GA for different population sizes. The step length and step time have been 0.42 m and 1.2 s, respectively.

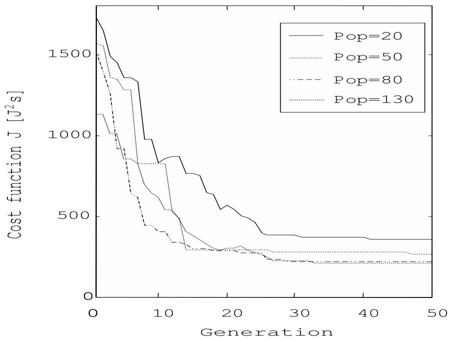


Fig. 5 Cost function J for different population sizes.

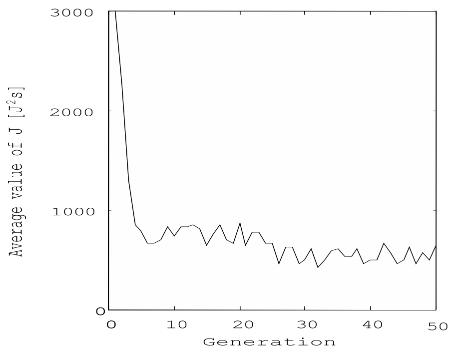


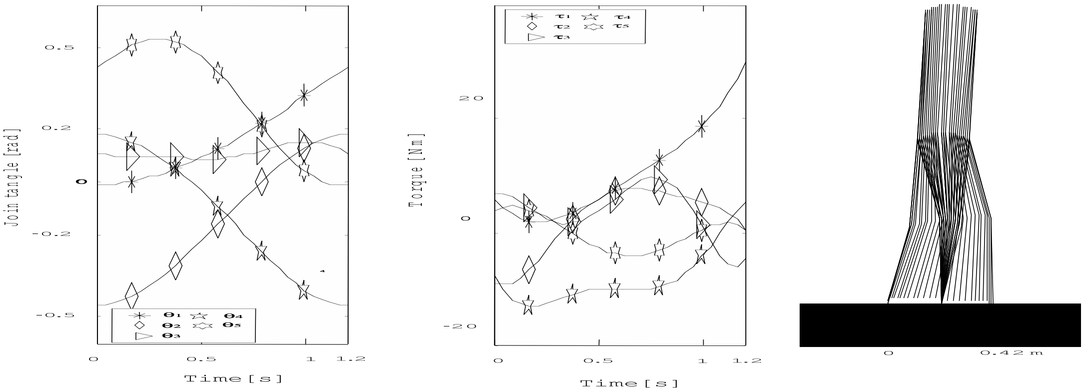
Fig. 6 Average of the cost function J vs. generations.

Table 3 Variable space and GA results.

GA Variables	Limits	CE	TC
θ_{10}	$-0.3 \sim 0.0$	-0.122	-0.0004
θ_{20}	$-0.7 \sim -0.3$	-0.455	-0.57
θ_{30}	$0.0 \sim 0.3$	0.1074	0.370
$\dot{\theta}_{1f}$	$0 \sim 2$	0.523	0.3995
$\dot{\theta}_{30}$	$-1 \sim 1$	-0.031	-0.11
$\dot{\theta}_{3p}$	$-0.1 \sim 0.2$	0.0840	0.370
t_3	$0.2 \sim 0.8$	0.5186	0.7612
x_p	$-0.2 \sim 0.2$	-0.135	-0.132
y_p	$0.01 \sim 0.04$	0.0163	0.017
t_p	$0.0 \sim 1.0$	0.441	0.432

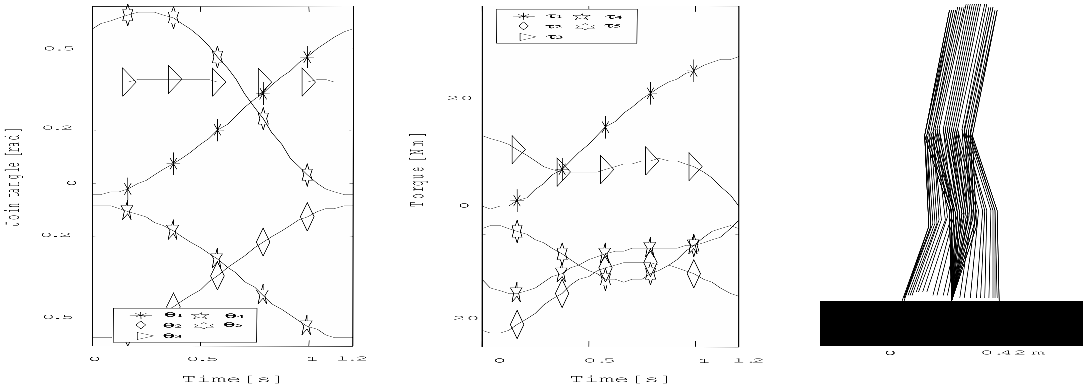
We see that for population size larger then 80, the solution quality doesn't change. The average of cost function J against the number of generations is shown in **Fig. 6**. The 33-th generation has the lowest value, which agrees with Fig. 4 and Fig. 5 results.

Based on the "Bonten-Maru I" parameters, the step length can vary up to 0.5 m. If the step length is smaller than 0.36 m, the ZMP can smoothly pass from one foot to the other during the instantaneous double support phase. The problem becomes more complex when the step length is larger then 0.36 m because the ZMP must jump to the new supporting foot. In the following, the optimal motion for step length



(a) (b) (c)

Fig. 7 GA results for minimum CE.



(a) (b) (c)

Fig. 8 GA results for minimum TC.

0.42 m and step time 1.2 s is analyzed. The GA results are shown in **Table 3**. The joint angle trajectories (θ_i), torque vector (τ_i) and optimal motions are shown in **Fig. 7** and **Fig. 8**, for minimum CE and minimum TC, respectively. As can be seen from Fig. 7(a) and Fig. 8(a), the boundary conditions are satisfied. Comparing Fig. 7(b) and Fig. 8(b), the torques change more smoothly when minimum TC is used as a cost function. The biped robot posture is straighter, similar to human walking, when minimum CE is used as cost function (Fig. 7(c) and Fig. 8(c)). The swing foot does not touch the ground prematurely, and the ZMP is always inside the sole length, as presented in **Fig. 9**. At the end of the step, the ZMP is at the position ZMP_f, as shown in Fig. 1. At the beginning of the step, the ZMP is not exactly at the position ZMP_{jump} because the foot's mass is not neglected. It should be noted that the mass of lower leg is different when it is in supporting leg or swing leg. The values of J cost function, calculated by Eq. (3) for minimum CE and minimum TC gait synthesis, are presented in **Fig. 10**. The minimum CE gait synthesis reduces the energy by about 30 % compared with minimum TC.

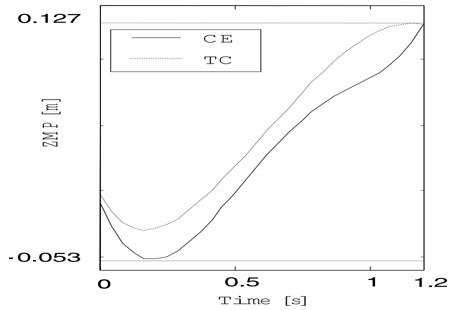


Fig. 9 ZMP trajectories.

In **Fig. 11** is presented a video capture of experiments with "Bonten-Maru I" humanoid robot. GA generates the optimal gait during walking. We see during the experiments that

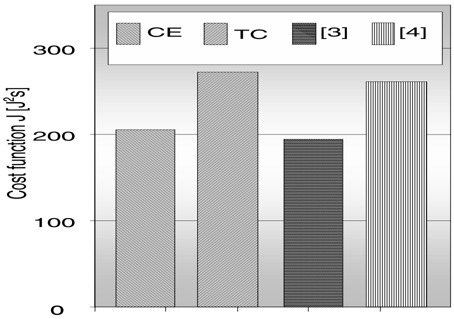


Fig. 10 Values of cost function J.

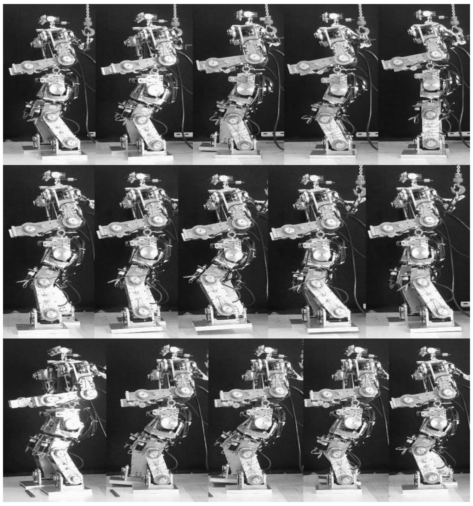


Fig. 11 Video capture of walking motion.

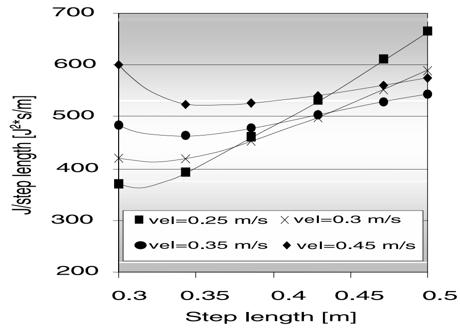


Fig. 12 Cost function J vs. the walking velocity.

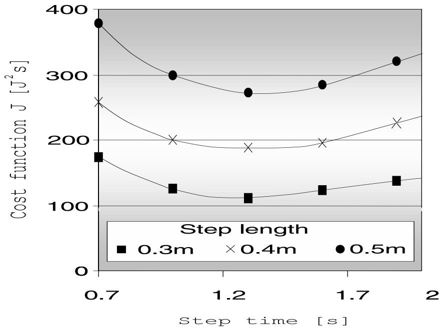


Fig. 13 Optimum step time for different step lengths.

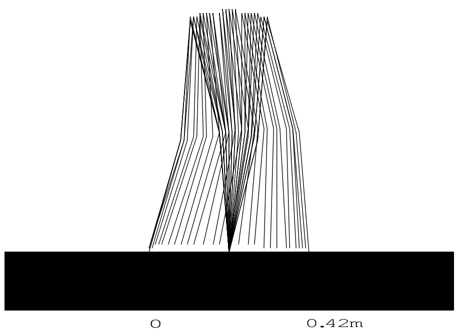


Fig. 14 Ref. 3) method optimal motion.

the optimal motion is very smooth. The walking is stable and the impact of the foot with the ground is very soft. The energy required for one meter walking against the step length is presented in **Fig. 12** for several walking velocities. In this case, the cost function is divided by step length. One result, which comes out from this figure, is that, as the walking velocity gets higher, the optimal step length gets larger; the curves become more tended and don't intersect with each other. The energy required when the biped is moving slowly with a large step length is high. This makes the curves of slow velocities to intersect with the others. This suggests that low walking velocity doesn't mean low CE. In addition to walking velocity, the step length must be also considered. In **Fig. 13** is presented the variation of cost function J versus the step time for the step lengths 0.3 m, 0.4 m and 0.5 m. It shows that each step length is optimal at one particular walking velocity.

5.1 Performance Evaluation

In the literature, the most recent algorithms for biped robot optimal gait generation are those presented in Refs. 3), 4). Here, we compare the results of our method with these two methods for the same step length and step time, respectively 0.42 m and 1.2 s. The criteria are:

(1) CE results;
(2) Stability.

In Ref. 3) the mass of body link is considered located at the hip, so it will be fixed in the vertical position. The optimal walking motion is presented in **Fig. 14**. The CE is 4.8 % lower compared with our optimal motion, as shown in Fig. 10. But, the calculated ZMP is not in-

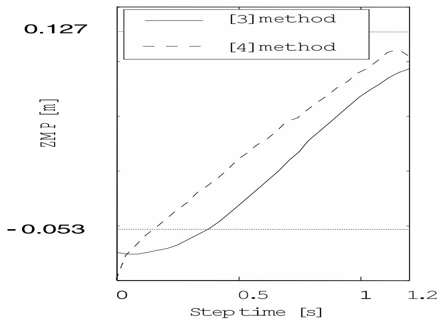


Fig. 15 ZMP trajectories.

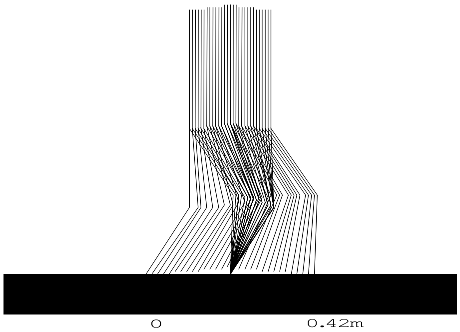


Fig. 16 Ref. 4) method optimal motion.

side the sole length, as can be seen in **Fig. 15**. This results in an unstable motion. As the step length becomes larger the calculated ZMP continues to move further out of the sole length.

In Ref. 4), the body is also fixed in the vertical position. The ankle joint of the swing leg and hip motions are produced based on sinusoidal functions. This motion has the advantage of soft impact. On the other hand, the irregularities of terrain must be very small. The optimal motion for the same foot clearance is presented in **Fig. 16**. The calculated ZMP is not inside the sole length at the beginning of the step (see Fig. 15. Because of additional constraints, the CE is larger compared with our optimal motion, as shown in Fig. 10. One important result here is that the stability must be considered when generating the optimal gait.

6. NN Implementation

In contrast to other optimization methods, GA needs more time to get the optimal solution. In our simulations, it needs about 10 minutes. However, in real time situations, based on the step length and step time, the angle trajectories must be generated in a very short time. In order to apply our method in real time, we considered teaching a RBFNN based on the GA

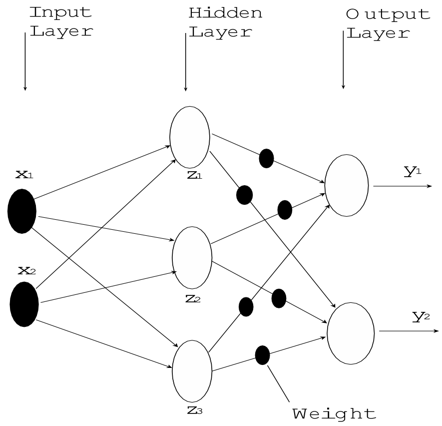


Fig. 17 RBFNN structure.

results. Our method employs the approximation abilities of a RBFNN [11]. When the biped robot has to walk with a determined velocity and step length, the RBFNN input variables would be the step length and step time. The output variables would be the same as the variables generated by GA. As we explained in Section 6, when the robot is not constrained to walk with a determined velocity for a given step length, the optimum velocity is the best from CE point of view. In this case, the RBFNN output will be the GA variables and the best step time. The RBFNN input will be only the step length. In this paper, we present the RBFNN simulation results, where as RBFNN input variable is used the step length.

6.1 RBFNN

The RBFNN involves three layers with entirely different roles, as shown in **Fig. 17**. The input layer connects the network to the environment. The second layer (hidden layer) applies a nonlinear transformation from input space to the hidden space, which is of high dimensionality. We use as nonlinear transfer function the Gaussian function, which is the most widely used. The Gaussian function is expressed as follows:

$$h_i(x) = exp(-\frac{\parallel x_i - c_i \parallel}{\sigma_i}), \qquad (9)$$

where: h_i is the i-th output of the neuron, x_i is the input vector, c_i and σ_i are the center and the width of the i-th RBF neuron. The width of Gaussian function is a positive constant that represents the standard deviation of the function. The output layer is linear, supplying the response of the network to the activation pattern (the signal applied to the input layer). Based on number of nodes in hidden layer, the

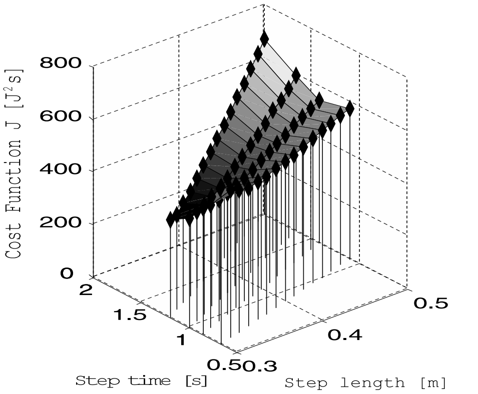


Fig. 18 Relation between cost function J, step length and step time.

RBFNN are divided in generalized and regularization RBF networks. In our simulations, we use a regularization RBF network.

6.2 RBFNN Results

The input layer of our RBFNN has one neuron and the output layer has 11 neurons. Because we use a regularization RBF network, the number of hidden neurons is the same with the number of training data. To teach the RBFNN, the step length varies from 0.3 m to 0.5 m and step time from 0.7 s to 2 s. Relation among the cost function J, step length and step time is presented in **Fig. 18**. Because in our RBFNN, the centers are the same with training data, determining the best value of the width σ is important in order to minimize the overall training error. The goal is to select the width σ in such way as to minimize the overlapping of nearest neighbors as well as maximize the generalization ability. The width selection depends on distance between two neighbor vectors. We consider the width σ the same for all neurons. In **Fig. 19** is shown the *mse* versus the width σ. The minimal value of *mse* is for $\sigma = 0.95$. The GA and RBFNN results for several different step lengths, which are different from the RBFNN training data, are presented in **Fig. 20**. The GA and RBFNN results are very close. The GA and RBFNN angle trajectories, for the step length 0.44 m, are shown in **Fig. 21**. The difference between the RBFNN and GA angle trajectories is very small. The ZMP trajectories for GA and RBFNN gait are all the time inside the sole length, as shown in **Fig. 22**. The RBFNN time to generate the solution is 50 ms, which is a good time for the real time implementation. The value of J cost function for RBFNN gait is only 3.7 % more

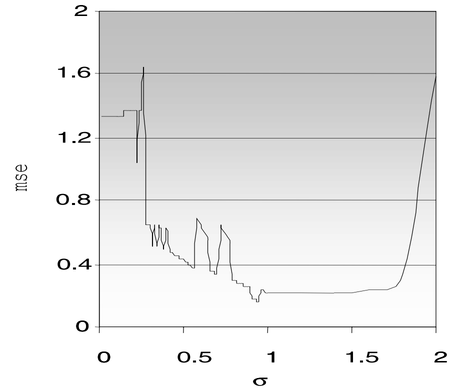


Fig. 19 mse vs. width.

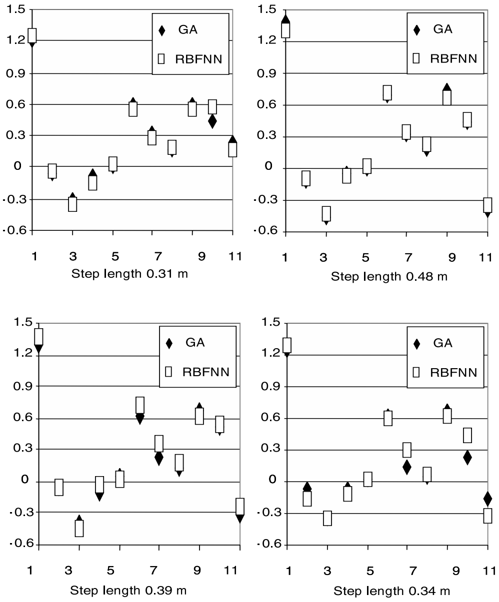


Fig. 20 GA and RBFNN results for different step lengths.

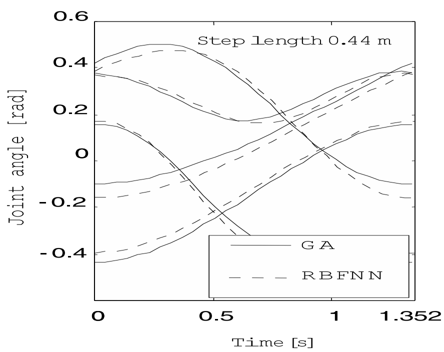


Fig. 21 GA and NN joint angle trajectories.

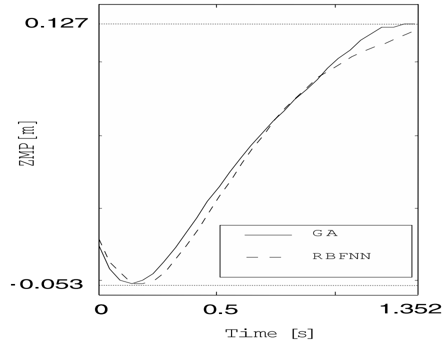


Fig. 22 *ZMP* trajectories.

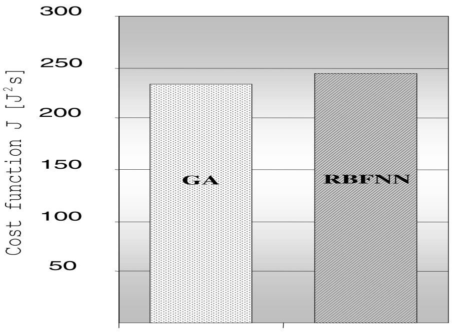


Fig. 23 Values of J cost function.

compared with GA gait, as shown in **Fig. 23**.

7. Conclusions

In this paper, a GA based approach for optimization of biped robot gait synthesis during walking is presented. The proposed method can be applied to generate the angle trajectories for other tasks performed by biped robots, such as overcoming obstacles, going down stairs, etc. By using GA as an optimization tool it is easier to include constraints and to add new variables to be optimized. To ensure a stable motion, because the time of double support phase is very short, the *ZMP* jumping is realized by accelerating the body link. By using a RBFNN, the real time implementation of proposed method is considered. In this paper, we presented the simulation results when as RBFNN input variable is used the step length. The simulations and experiments are performed using the parameters of "Bonten-Maru I" humanoid robot. Based on the simulation and experimental results, we conclude:

- each step length is optimal at a particular velocity;
- the stability must be considered when generating the optimal gait;
- the biped robot posture is straighter when minimum CE is used as cost function, which is similar to human's walking;
- the energy for CE is reduced 30 % compared with TC cost function;
- the optimal motion is smooth and stable;
- RBFNN gives good results for real time implementation.

References

1) Vukobratovic, M., Borovac, B., Surla, D. and Stokic, D.: *Biped Locomotion, Stability, Control and Application*, Springer-Verlag (1990).
2) Takanishi, A., Ishida, M., Yamazaki Y. and Kato, I.: A Control Method for Dynamic Biped Walking Under Unknown External Force, *Proc. IEEE Int. Workshop on Intelligent Robots and Systems*, pp.795–801 (1990).
3) Roussel, L., Canudas-de-Wit, C. and Goswami, A.: Generation of Energy Optimal Complete Gait Cycles for Biped Robots, *Proc. IEEE Int. Conf. on Robotics and Automation*, pp.2036–2041 (1998).
4) Silva, F.M. and Machado, J.A.T.: Energy Analysis During Biped Walking, *Proc. IEEE Int. Conf. on Robotics and Automation*, pp.59–64 (1999).
5) Channon, P.H., Pham, D.T. and Hopkins, S.H.: A Variational Approach to the Optimization of Gait for a Bipedal Robot, *Journal of Mechanical Engineering Science*, Vol.210, No.1, pp.177–186 (1996).
6) Goldberg, D.E.: *Genetic Algorithm in Search Optimization, and Machine Learning*, Addison Wesley (1989).
7) Capi, G., Nasu, Y., Barolli, L., Mitobe, K. and Takeda, K.: Application of Genetic Algorithms for Biped Robot Gait Synthesis Optimization During Walking and Going up-Stairs, *Advanced Robotics Journal*, Vol.15, No.6, pp.675–694 (2001).
8) Uno, Y., Kawato, M. and Suzuki, R.: Formulation and Control of Optimal Trajectory in Human Multijoint Arm Movement, *Biol. Cybernet.*, Vol.61, pp.89–101 (1989).
9) Mita, T., Yamaguchi, T., Kashiwase, T. and Kawase, T.: Realization of High Speed Biped Using Modern Control Theory, *Int. J. Control*, Vol.40, No.1, pp.107-119 (1984).
10) Michalewich, Z.: *Genetic Algorithms + Data Structures = Evaluation Programs*, Springer-Verlag (1994).
11) Haykin, S.: *Neural Networks a Comprehensive Foundation*, Prentice Hall International (1999).

(Received February 7, 2001)
(Accepted January 16, 2002)

Genci Capi received B.E. and Ph.D. degrees from Polytechnic University of Tirana and Yamagata University in 1993 and 2002, respectively. From December 1993 to September 1998, he worked as a Design Engineer at Research and Design Institute of Tirana. From April 2002, he is a Post Doctor Researcher at ATR Human Information Science Laboratories. His research interests include humanoid robots, mobile robots, genetic algorithms, and neural networks. He is a member of ASME and IEEE.

Yasuo Nasu received B.E. degree in Mechanical Engineering from Yamagata University in 1962, and M.E. and D.E. degrees in Precision Engineering from Osaka University in 1964, and 1976, respectively. From 1965 to 1977 he was a Lecturer, from 1977 to 1984 an Associate Professor and is currently a Professor at Department of Mechanical Systems Engineering, Yamagata University. His research interests include robotics manufacturing, control of walking and humanoid robots, and adaptive control systems theory. He is a member of RSJ.

Leonard Barolli received B.E. and Ph.D. degrees form Tirana University and Yamagata University in 1989 and 1997, respectively. From April 1997 to March 1999, he was a JSPS Post Doctor Researcher at Faculty of Engineering, and from April 1999 to March 2002 a Research Associate at Department of Public Policy and Social Studies, Yamagata University. From April 2002, he is an Assistant Professor at Department of Computer Science, Saitama Institute of Technology (SIT). His research interests include network traffic control, intelligent algorithms and agent-based systems. He is a member of SOFT and IPSJ.

Kazuhisa Mitobe received B.E. and M.E. degrees in Precision Engineering from Niigata University in 1986 and 1988, respectively, and Ph.D. degree from Kyoto University in 1996. From 1990 to 1998, he was a Research Associate and is currently an Associate Professor at Department of Mechanical Systems Engineering, Yamagata University. His research interest includes robotics manufacturing, control of walking robots, and adaptive control systems theory. He is a member of RSJ.

Mitsuhiro Yamano was born in 1972. He received the B.E. in mechatronics and precision engineering from Tohoku University in 1995, and M.E. and Ph.D. degrees in aeronautics and space engineering from the same university in 1997 and 2000, respectively. He is currently a Research Associate at Yamagata University. His research interests include flexible robots and humanoid robots.

Kenro Takeda received B.E., M.E. and Ph.D. degrees in Information Engineering from Yamagata University in 1994, 1996 and 2001, respectively. He is now working as programmer at Tohoku RICOH Company. His research interests include control of humanoid robots, parallel and distributed computing, and neural networks.