Biped Gait Optimization Using Spline Function Based Probability Model

Lingyun Hu[1,2], Changjiu Zhou[2], Zengqi Sun[1]

[1]Department of Computer Science and Technology, Tsinghua University, Beijing, 100084, P.R. China
[2]School of Electrical and Electronic Engineering, Singapore Polytechnic, 500 Dover Road, 139651, Singapore
Email: huly02@mails.tsinghua.edu.cn, ZhouCJ@sp.edu.sg, szq-dcs@mail.tsinghua.edu.cn

Abstract— A new Estimation of Distribution Algorithm (EDA) with spline kernel function (EDA_S) is proposed to optimize biped gait for a nine-link humanoid robot. Gait synthesis of the biped locomotion is firstly formulated as a multi-constraint optimization problem with consideration of two objectives, including zero-moment point (ZMP) for dynamically stable locomotion and driving torque for energy efficiency. The parameters to be optimized are joint coordinates at transition poses between three successive phases. Rather than searching in joint angle permutation space directly, the proposed method approximates the probability distribution by Catmull-Rom (CR) cubic spline function and updates them with gradient descent learning strategy. The effectiveness of EDA_S for biped gait optimization has been successfully tested on the simulated model of a humanoid soccer robot. It is shown that the flexible kernel with the updating rule is able to remarkably accelerate the learning speed with comparison to the traditional EDA.

Index Terms—Biped robot, gait optimization, probability distribution, EDA, spline function.

I. INTRODUCTION

Researches related to humanoid robot have made notable progress in design, control and systemization of biped machine. However, gait generation and optimization still remains a challenge for such a high-order highly-coupled nonlinear dynamical system. To create valid biped walking, various criteria have been proposed such as stability and energy efficiency.

Zero Moment Point (ZMP) is one of the mostly used stability criteria for biped gaits. Recent humanoid robots using ZMP control strategies have demonstrated impressive biped walking [1, 2]. However, stability criterion is not good enough to guarantee the feasibility of biped gaits. Research of energy consumption in biped walking provides another useful criterion for gait optimization. In 1990, McGeer presented the possibility of passive dynamic walking [3]. Followed with him, several studies have explored energy efficient biped walking [4].

With consideration of the two criteria, many researches focus on intelligent paradigms including connectionist theory [5], fuzzy logic [6] and Genetic Algorithm (GA) [7] to optimize biped gaits. However, these heuristic computation algorithms depend greatly on a large number of associated parameters. Experiences are required to choose the suitable values for fast resolution.

To get the solution with less parameter, a new type of population-based searching algorithm named Estimation of Distribution Algorithms (EDAs) is adopted in this paper [8]. EDAs try to predict the population movements in the search space as well as to avoid needing too many parameters. Being one type of EDAs for combinatorial optimization, $PBIL_c$, the extension of the Boolean PBIL (Population-Based Incremental Learning) to continuous space, performs successfully on numerous benchmark and real world problems with comparison to a variety of standard GAs and hill-climbing algorithms [9]. It is used in this paper to optimize the generated biped gaits under energy and stability criteria with the assumption that selected variables are independent.

However, Gaussian function is not able to precisely describe complex multi-peak distribution. So the proposed algorithm EDA_S updates traditional EDA with non-parametric technique, namely spline functions [10], to address the probability distribution of joint coordinates. The essence of spline-based probability model is to represent an arbitrary continuously probability function with a discrete sum of weighted and shifted Catmull-Rom (CR) basis functions. Sampling points are updated according to gradient descent method. In this paper, we demonstrated that CR spline function, as a kind of sub-optimization way of cubic spline, can approximate arbitrary continuous function with arbitrarily high quality too. This will speed up the modification of distribution density model and consequently accelerate the joint permutation searching. Also the quality of the proposed method is only affected by a few parameters. This has been tested by the application in biped gait optimization.

II. PROBLEM BACKGROUND

Gait pattern of a nine-link biped robot is presented in this section to produce optimization parameters. Followed with it is the mathematic model for constraints definition.

A. Biped gait pattern

Gait model of a N_l links planar biped robot consisting of a torso, two hips and two identical legs with ankles and knees is analyzed. Biped walking is assumed to limit in sagittal plane and on a level surface, consisting the swing and double support phases as shown in Fig. 1.

- Swing Phase

The swing phase starts from the swing foot toeing off the ground and ends when the heel touches the ground. The swing foot reaches the highest point at T_a and after T_b lands on the
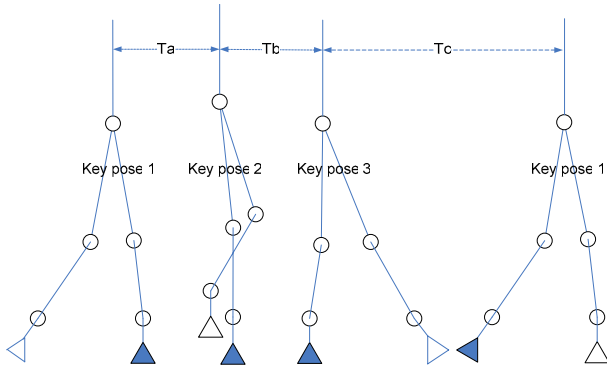
Fig. 1. Gait pattern of one cycle.

Fig. 2. Mechanical model of the robot.

ground. The transition moment between swing phase and double support phase is labeled as $T_a + T_b$.

This phase is modeled as a continuous forward motion in which, the support leg plays as a pivot and the other leg swings forward. Let q be the coordinate sets describing the link locomotion in joint space with respect to a world reference frame. $\dot{q}$ is the angular velocity of the mechanism. With above assumptions, the mechanism dynamic model of biped robots can be expressed by similar dynamic Newton-Euler equation like manipulators as Equation (1).

$$\hat{M}(q)\ddot{q} + \hat{C}(q,\dot{q}) + \hat{G}(q) = \hat{\tau} \qquad (1)$$

Where $\hat{M}(q) = \{p_{ij}\cos(q_j - q_i)\}_{N_l \times N_l}$ is the square matrix describing full inertia, $\hat{C}(q) = \{-p_{ij}\sin(q_j - q_i)\}_{N_l \times N_l}\dot{q}$ and $\hat{G}(q) = \text{diag}\{-f_i \sin q_i\}_{N_l \times 1}$ are the vector of centrifugal, Coriolis and gravitational moments acting at N_l mechanism joints respectively, $\hat{\tau}$ is the vector of driving moments at each joint. $J^T(q,\ddot{q})_{6 \times N_l}$ is the corresponding Jacobi matrix of the system. Parameters r_{ij}, s_{ij} and h_i are constants derived by Lagrange's equation. Inequality geometric constraints as shown in the next section are used to avoid collisions between ground and the swing foot.

- Double Support Phase

The double support phase begins at the moment that the front heel touching the ground and ends when the rear toeing sways off at $T_a + T_b + T_c$ (as shown in Fig. 1). End configuration of this phase initiates the swing phase of the next gait cycle.

Landing impact happens at the beginning of double support phase. The impulsive forces may cause an instantaneous velocity change in the generalized coordinates while the positions remain continuous. It can be described by Equation (2).

$$\hat{M}(q)\ddot{q} + \hat{C}(q,\dot{q}) + \hat{G}(q) = \hat{\tau} + \partial F_e \qquad (2)$$

Where F_e is the integral of contact impulse at the impact instant. By supposing that the actuators are not impulsion, the velocity right after the impact $\dot{q}^+$ can be represented as $\dot{q}^+ = \Delta(\dot{q}^-)$, where $\dot{q}^-$ is the velocity right before the impact and Δ is a continuous function with unique solution.
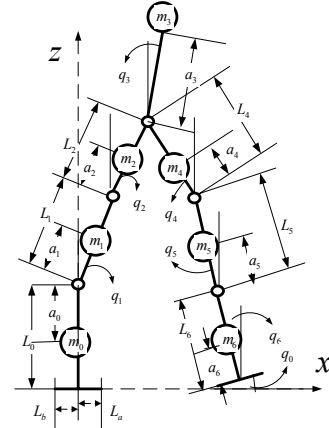
After the impact moment, the mechanical kinematic chain in double support phase can be considered as a manipulator with end position constraints. The dynamical model is given in Equation (3).

$$\hat{M}(q)\ddot{q} + \hat{C}(q,\dot{q}) + \hat{G}(q) = \hat{\tau} + J^T(q,\ddot{q})F \qquad (3)$$

Where F is the vector of external forces and moments acting at the landing foot of the robot.

It can be seen from Fig. 1, totally three key poses are chosen in one cycle. Phases between these key poses are approximated by third order spline functions. Such strategy guarantees the second order derivatives at every point, especially at connection points. Consequently, actuating torques is continuous at knots.

Ratio between swing and double support periods is a function of walking velocity [2]. To be mentioned in simulation as an example, swing phase is set to be twenty percent of one walking cycle according to [2].

B. Mathematical model

The mathematical model of the biped locomotion proposed in [11] is modified in this paper. The spatial model defined in internal coordinate state space as shown in Fig. 2 will be used to synthesize the dynamic control of the biped mechanism and to verify the simulation research result. Definition of angular coordinates and disposition of the masses are also indicated in Fig. 2. The supporting ankle, noted as l_0 is adopted as the basic link of the mechanism. Since only symmetric gaits are considered, the same model can be used for different supporting leg by re-labeling coordinates.

III. CONSTRAINTS

Natural principles for biped gait control criteria should include 1) anthropomorphic characteristics of a gait; 2) dynamic balance during the locomotion; 3) energy cost at robot joints; 4) impact between swing foot and the ground at the contact moment. Since dynamical model of the biped robot considers only the last criterion, other three one will be realized through constraints design.

Constraints are divided into two categories, namely geometric and state constraints in this section according to the

definition space.

A. Geometric constraints

Geometric constraints (GCs) ensure the feasibility of biped gaits from the point of physical structure. Since one gait cycle can be regarded as a pose sequence that is generated by interpolating key poses, GCs will be constructed only at key poses. Two geometric constraints are designed as follows.

GC1) Positions limitations of knots as shown in Equation (4).

$$A_p \leq p(\phi) \leq B_p \tag{4}$$

Where $p = [p_i(t)]^T$, $A_p = [A_{p_i}(t)]^T$, $B_p = [B_{p_i}(t)]^T$, $p_i(t) = [x_i(t), y_i(t), z_i(t)]$ denotes center position of the ith link at t moment, $i = 1, 2, ..., N_l$, $A_{p_i}(t)$ and $B_{p_i}(t)$ are lower limit and upper limit of $p_i(t)$.

GC2) Structure limitations of joint angles as shown in Equation (5).

$$A_q \leq q(\phi) \leq B_q \tag{5}$$

Where $q = [q_i(t)]^T$ stands for the joint angle at time t, $i = 0, 1, ..., N_l - 1$, $A_q = [A_{q_i}(t)]^T$, $B_q = [B_{q_i}(t)]^T$, $A_{q_i}(t)$ and $B_{q_i}(t)$ are lower limit and upper limit of $q_i(t)$.

B. State constraints

Three more kinds of state constraints include force, velocity and stability constraints are also taken into consideration.

- **Force constraints**

During double support phase, the leg force must satisfy the force constraint **FC1** as

$$\sum_{i=1}^{N_L} m_i \ddot{p}_i = f_R + f_L + \sum_{i=1}^{N_L} m_i g . \tag{8}$$

Where f_R and f_L are foot force at right and left legs respectively. m_i, $\ddot{p}_i$ are the mass and the acceleration of the ith link. Since only the sum $f_R + f_L$ is known during the double support phase, another force constraint **FC2** (as shown in Equation (9)) is taken into assumption as the internal force f_d in the closed loop structure must be minimized.

$$f_d = \min\{F(f_R, f_L)\} \tag{9}$$

Where F is the function to calculate the internal force.

- **Velocity constraints**

Velocity constraints (**VC**) are considered to guarantee motion smoothness with respect to mechanical limitations of the biped system. VC can be simply written as

$$A_{\dot{q}} \leq \dot{q}(\phi) \leq B_{\dot{q}} . \tag{10}$$

Where $A_{\dot{q}} = [A_{\dot{q}_i}(t)]^T$ and $B_{\dot{q}} = [B_{\dot{q}_i}(t)]^T$ are lower and upper boundaries of the velocity $\dot{q}_i(t)$.

- **Stability constraint**

To achieve a dynamically stable gait, ZMP is usually used as a criterion [1, 2]. The ZMP is defined as the point on the ground at which the net moment of the inertial forces and the gravity forces has no component along the horizontal axes [12]. It can be represented as Equations (11) and (12).

$$x_{zmp} = \frac{\sum \{m_i(\ddot{z}_i + g)x_i - m_i \ddot{x}_i z_i - (I_i \ddot{\theta}_i)_y\}}{\sum m_i(\ddot{z}_i + g)} \tag{11}$$

$$y_{zmp} = \frac{\sum \{m_i(\ddot{z}_i + g)y_i - m_i \ddot{y}_i z_i + (I_i \ddot{\theta}_i)_x\}}{\sum m_i(\ddot{z}_i + g)} \tag{12}$$

To achieve dynamically stable gait, the ZMP constraints (**ZC**) should satisfy

$$p_{zmp}(\phi) \in S_{zmp}(A_{zmp}, B_{zmp}) . \tag{13}$$

Where $p_{zmp} = (x_{zmp}, y_{zmp}, 0)^T$, S_{zmp} is the support polygon defined by A_{zmp} and B_{zmp}, which are lower and upper boundaries of the stable region respectively.

IV. PROBLEM FORMULATION

The proposed constrained dynamical model with geometric and state constraints provides the searching space boundary. Based on the proposed framework, an optimization procedure will be developed to select a continuous feasible solution by minimizing an appropriate cost function. Since analytical property of the performance criterion affects the optimal searching in terms of either dynamics or stabilities. ZMP displacement Υ_1 and energy cost Υ_2 are chosen to form the cost function. Performance criterion is defined as:

Minimize $\Upsilon = \Upsilon_1 + \Upsilon_2$

Subject to

GC1 $A_p \leq p(\phi) \leq B_p$

GC2 $A_q \leq q(\phi) \leq B_q$

FC1 $\sum_{i=1}^{N_L} m_i p_i = f_R + f_L + \sum_{i=1}^{N_L} m_i g$

FC2 $f_d = \min\{F(f_R, f_L)\}$

VC $A_{\dot{q}} \leq \dot{q}(\phi) \leq B_{\dot{q}}$

ZC $p_{zmp}(\phi) \in S_{zmp}(A_{zmp}, B_{zmp})$

Where $\Upsilon_1 = \int_{t=0}^{Ta+Tb+Tc} \|p_{zmp}(t) - p_{zmp}^d(t)\|^2 dt$ estimates the Euclidean distance between the actual ZMP trajectory p_{zmp} and the desired one p_{zmp}^d. $\Upsilon_2 = \int_{t=0}^{Ta+Tb+Tc} \tau(t)dt$ represents the dynamic load during one gait cycle $[0, T_a + T_b + T_c]$.

Let N_p be the total number of discrete sample poses in one cycle. Thus the cost function can be defined as Equation (14).

$$\Upsilon(\bullet) = \beta_f f(\bullet) + \beta_g g(\bullet) = \sum_{i=1}^{N_p} (\beta_f \mathbb{N}(f_i(\bullet)) + \beta_g \mathbb{N}(g_i(\bullet)) \tag{14}$$

Where $f_i(\bullet) = \frac{1}{2}\|p_{zmp}^i - p_{zmp}^d\|^2$ calculates the Euclidean distance between the actual ZMP and the desired ZMP at the ith sampling index. $g_i(\bullet) = \sum_{j=1}^{N_t} \tau_{ij}$ summarizes the torque of all actuated joints at the ith sample index. Where N_t is the number

of driving torques. β_f and β_g are weights satisfying $\beta_f + \beta_g = 1$. $\mathbb{N}$ is the normalization operator.

It is proved that ZMP trajectory will satisfy third order spline interpolation function if joint trajectories are generated by the same function [13]. So the proposed algorithm improves only key states to minimize the desired cost function.

V. SPLINE FUNCTION BASED EDA

A new evolutionary algorithm named EDAs is adopted to solve the biped gait optimization problem. EDA is based on the probability distribution function where selection and recombination of building blocks in GAs are replaced by selection from probability models and modification of probability models [8].

As the input coordinates q_i (i=1, 2, …, N_l) of the N_l links in joint space are considered independent, their probability distribution functions satisfy $\Pi_{i=1}^{N_l} \text{Pro}_i(q_i)$, which is the product of N_l univariate distribution $\text{Pro}_i(q_i)$. To describe the multi-model distribution function for each joint coordinate exactly, one non-parametric model, spline function, is used with cubic CR basis in the proposed method EDA_S. The kernels will be updated with gradient descent method. By this means, EDA_S can characterize more complex distribution functions since every continuous function on a closed interval can be approximated uniformly to any prescribed accuracy by a polynomial [14].

A. Spline function based probability model

Different from traditional probability model, the proposed algorithm employs piecewise polynomial spline interpolation schemes, which is local adaptive and has continuous first order derivative. For given points $\{W_1 \dots W_{Nw}\}$, $W_r = [w_{x_r}, w_{y_r}]^T$, a general spline can be described by Equation (15).

$$F(u) = \bigcup_{i=1}^{N_w-3} F_i(u) \qquad (15)$$

Where $\bigcup$ is the concatenation operator on local spline basis functions. N_w is the number of points and

$$F_{y_r}(u) = \sum_{j=0}^{3} C_j(u) w_{y_{r+j}} . \qquad (16)$$

$F_{y_r}(u)$ ($0 \leq u \leq 1$) is the rth curve span function controlled by four points $[W_r, W_{r+1}, W_{r+2}, W_{r+3}]^T$ and $C_j(u)$ is the CR cubic spline basis, which is a sub-optimal and adaptive realization of the cubic spline function with the kernel $|x^3|$. We use CR spline, as shown in Equation (17), due to its low computation load and local characteristic to avoid the oscillatory in global adaptation. Cubic polynomials are chosen to achieve a better balance between the computational complexities and to guarantee both continuity and the existence of its derivatives.
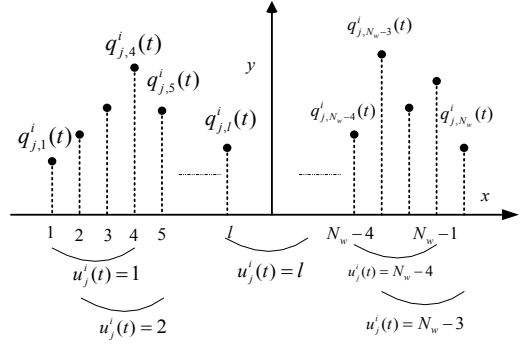

Fig. 3. Structure of spline functions.

$$C(u) = \begin{bmatrix} C_0(u) \\ C_1(u) \\ C_2(u) \\ C_3(u) \end{bmatrix}^T = \frac{1}{2} \begin{bmatrix} u^3 \\ u^2 \\ u \\ 1 \end{bmatrix}^T \times \begin{bmatrix} -1 & 3 & -3 & 1 \\ 2 & -5 & 4 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 0 \end{bmatrix} \qquad (17)$$

For input $W = [w_x, w_y]^T$, proper local parameters of u and r can be obtained by $W_x = F_{x_r}(u)$. To accelerate the computation of local parameters and to reduce free parameters, we uniformly sample the x-axis with fixed step length $\Delta w = w_{x_{r+1}} - w_{x_r}$. This turns $F_{x_r}(u)$ to a first degree polynomial as

$$F_{x_r}(u) = u\Delta w + w_{x_r} . \qquad (18)$$

It has been proved that function smoothness can be tuned by $\triangle w$, which is insensitive to generalization error in a suitable range [15]. So the proposed method sets it by manual tuning.

Supposing q_j^i is the input angle of joint j at the ith key pose in one gait cycle. Output of this model Pro_j^i can be calculated by Equations (19) to (21) (see Fig. 3). Where i=1, 2, …, N_{kp}, j=1, 2, …, N_l.

$$r_j^i = \left\lfloor \frac{q_j^i}{\Delta w} + \frac{N_w}{2} \right\rfloor \qquad (19)$$

$$u_j^i = \frac{q_j^i}{\Delta w} + \frac{N_w}{2} - r_j^i \qquad (20)$$

$$Pro_j^i = F_{j,y_r}^i(\cdot) = \sum_{m=0}^{3} w_{j,x_{(r_j^i+m)}}^i C_{j,m}^i(u_j^i) \qquad (21)$$

Equations (19) and (20) implement local parameter computation. $\lfloor \ \rfloor$ is the floor operator and the second term in Equation (19) is designed to guarantee u_j^i be always nonnegative.

The CR spline approximation error is a function of the sampling step $\triangle w$ and approximation order L. Since CR spline is a sub-optimal of the cubic spline function, it can also be understood as a subclass of Moms functions, which stands apart as the best achievable compromise between approximation quality and speed [16].

B. Gradient descent updating rule

Supposing that $D_j^i(t)$ is the desired output. Consider a representation of the cost function $C(q)$ using a Boltzmann distribution as

$$D_j^i(x) = \frac{1}{Z} e^{-\frac{\overline{c}(q)}{T}} = \frac{1}{Z} e^{-\frac{1}{Tc(q)}}. \qquad (22)$$

Where $\overline{c}(q) = 1/c(q)$, then

$$E_d(k) = \frac{1}{2}(D_j^i(k) - \mathrm{Pro}_j^i(k))^2. \qquad (23)$$

Equation (23) is the squared output error related to the kth iteration. According to gradient descent algorithm, the control points are updated with anti-gradient of error surface as

$$q_{j,(r_j^i+m)}^i(k+1) = q_{j,(r_j^i+m)}^i(k) - \mu_q \frac{\partial E_d(k)}{\partial q_{j,(r_j^i+m)}^i}(k). \qquad (24)$$

Where μ_q is learning rates for control points. m=0, …, 3, i=1, 2, …, N_{kp}, j=1, 2, …, N_l. By substituting (23) to (24), we have

$$q_{j,(r_j^i+m)}^i(k+1) = q_{j,(r_j^i+m)}^i(k) + \mu_q (D_j^i(k) - \mathrm{Pro}_j^i(k)) C_{i,m}^j(u_j^i(k)). \qquad (25)$$

C. EDA_S

The proposed algorithm EDA_S can be outlined as follows.
1. Give a multivariate distribution model $\mathrm{Pro}_j^i(1)$ represented by N_w random sample points $q_{j,l}^i$, l=1, 2, …, N_w, i=1, 2, …, N_{kp}, j=1, 2, …, N_l.
2. Generate a population of N_s individuals.
3. While termination condition is not met:
   a) Select the best N_b individuals from parent generation k's population according to cost function value Υ.
   b) Update $q_{j,l}^i(k)$ and $\mathrm{Pro}_j^i(k)$ with the N_b individuals according to Equations (24) and (21).

VI. EXPERIMENT RESULTS

To show the effectiveness of the proposed algorithm, we apply it on the simulation model of the humanoid robot named Robo-Erectus (RE) as shown in Fig.4 [17]. Totally 23 Degree of Freedoms (DOFs) are designed in RE as 6 per leg, 4 per arm and 1 for head, 2 for waist. They are driven by servo motor with maximum torque as 30kg×cm. One notebook with window XP operation system is set on RE for online control. (www.robo-erectus.org)

Based on the robot, a simplified model as shown in Fig. 2 is

Fig. 4. Physical model of robot Robo-Erectus. (60cm, 4.6kg, 23 DOFs)

constructed for simulation experiment. Basic structure parameters of the model are $N_l = 9$, $N_{kp} = 3$, $N_p = 10$, L_h=10 cm, L_0=4cm, $L_1 = L_2 = 8$cm, L_3=20cm, D_s=20cm, $M_0 = 0.1$kg, $M_1 = M_2$=0.2kg, M_3=1kg. Desired ZMP is set as the middle line in the stable region.

Other parameters used in EDA are designed as N_s=16, N_b=4, $\beta_f = \beta_g = 0.5$. Number of sample points N_w=20 is chosen empirically [15]. Terminal condition to stop learning is $\Upsilon(k) < 3$ for five continuous iterations or k>100.

To verify the effectiveness of EDA_S, an analytical experiment is taken on it with comparison to traditional EDA with Gaussian distribution kernel $G(\mu_i, \sigma_i)$ (EDA_G). Same number of candidate solutions is generated in EDA_G to select the best, the second best and the worst samples for the modification on the means μ_i and covariance σ_i [13]. Details of these two learning methods are given in Table I. Where α is the learning rate and $\tilde{\mu}_i^j$ is the average value of μ_i^j. $\Upsilon^i(k)$ records the cost function value at the kth iteration in the ith experiment.

Fig 5. gives the value variation of the cost function by these two algorithms. It can be found that EDA_S stands out notably in term of convergent speed. Its cost function value reduces third of the initial value in less than 50 epoches while the cost function value of EDA_G does not show obvious descent tendency with the same learning rate. The avearage of ten times experiment results of EDA_S is also better than that of EDA_G (see Table I). The corresponding ZMP trajectory and joint torques of the learned gait is illustrated in Fig. 6 and Fig 7 with comparison to the one before learning. They regress to proper value in logical region as the learning goes on. To represent it more clearly, value variation of the two critera before normalization along the learning epoches is also exhibited.

It can be concluded that

TABLE I
DETIALS OF THE TWO ALGORITHMS AND TEN TIMES EXPERIMENT RESULTS

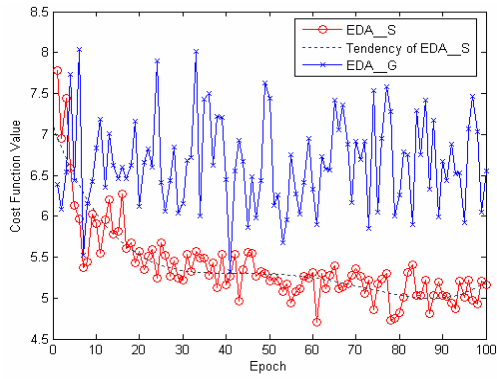|	EDA_S	EDA_G
Distribution model	Spline function	Gaussian Distribution Function $G(\mu_i, \sigma_i)$
Updating rules for model	Equations (21) and (25)	$\mu_i(k+1) = (1-\alpha)\mu_i(k) + \alpha(\mu_{i,best}(k) + \mu_{i,2ndbest}(k) - \mu_{i,worst}(k))$ $\sigma_i(k+1) = (1-\alpha)\sigma_i(k) + \alpha\sqrt{\frac{1}{N_s}\sum_{j=1}^{N_s}(\mu_i^j(k) - \tilde{\mu}_i^j(k))^2}$
Learning rate	0. 01	0.01
$\frac{1}{10}\sum_{i=1}^{10}\min(\Upsilon^i(1),...,\Upsilon^i(k))$	4.62	5.38

Fig. 5. Cost function value in the first 100 loops.

1) Spline functions characterize the probability distribution in a more previous way.

2) Gradient descent updating rule plays an important role in acclerating the final learning. It makes EDA_S learning continuely even the best one in the current population is not significantly greater than the rest individuals.

VII. CONCLUSION AND FUTURE WORK

This paper proposes a new biped gait optimization method EDA_S to achieve a dynamically stable and low energy cost walking by using EDA with spline kernel and gradient descent updating rule. A nine-link biped robot is firstly formulated as a nonlinear system with impulse effects, evolving in a subset of R^9 . In this framework, the biped gait optimization can be modeled as a multi-object optimization problem with multi-constraints. By modeling the probability distribution of joint angles with the non-parametric kernel, EDA_S transfers the searching in infinite dimensional joint space to probability distribution space constructed by CR cubic spline functions. The effectiveness of EDA_S for biped gait optimization has been successfully tested on a humanoid soccer robot named Robo-Erectus [17], which has won several awards in RoboCup competitions. It is shown that the EDA_S is able to speed up
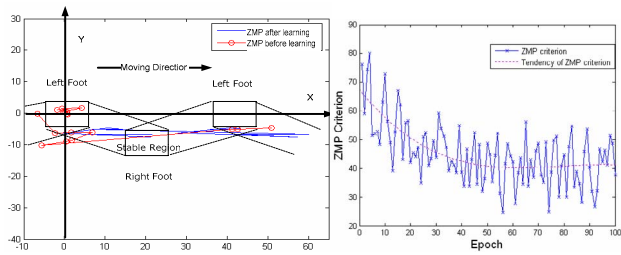

Fig. 6. Left figure shows ZMP trajectories before and after learning. Right figure is sum of distances between the real ZMP and Middle line of the stable region at sample points.
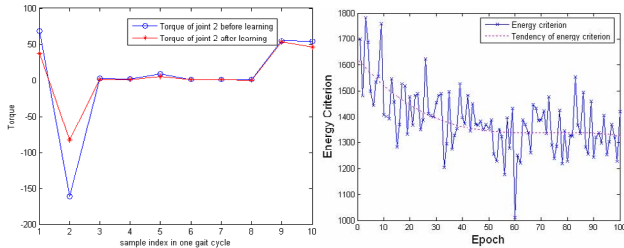

Fig. 7. Left figure is joint torques before and after learning. Right figure is sum of torque at actuated joints at sample points.

learning by interpolating distribution function with gradient descent learning algorithm.

For the proposed spline-based EDAs for biped gait optimization, the basic assumption is that the joint angles are independent variables, which may not be valid for some more general applications. To find out more meaningful mapping between object function and state variables for biped gait generation and optimization, our next research will focus on the variable interrelationship in EDA's probability model for biped gait transition. We will also look at how to use computational learning approaches, e.g. fuzzy reinforcement learning to further optimize biped gaits [6].

REFERENCES

[1] C. Chevallereau, Y. Aoustin, "Optimal reference trajectories for walking and running of a biped robot," *Robotica*, vol. 19, pp. 557-569, 2001.
[2] Q. Huang, et al, "Planning walking patterns for a biped robot," *IEEE Trans. on Robotics and Automation*, vol. 17, no. 3, pp. 280-290, 2001.
[3] T. McGeer, "Passive dynamic walking," *Int. Journal of Robotics Research*, vol.9, no.2, pp. 62-82, 1990.
[4] F. Asano, M. Ymakita, K. Furuta, "Virtual passive dynamic walking and energy-based control laws," *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, vol. 2, pp. 1149-1134, 2000.
[5] J. Hu, J. Pratt, G. Pratt, "Stable adaptive control of a bipedal walking robot with CMAC neural networks," *Proc. IEEE Int. Conf. on Robotics and Automation*, Detroit, USA, pp. 1950-1956, 1999.
[6] C. Zhou, Q. Meng, "Dynamic balance of a biped robot using fuzzy reinforcement learning agents," *Fuzzy sets and Systems*, vol. 134, no. 1, pp. 169-187, 2003.
[7] G. Capi, S. et al, "Optimal trajectory generation for a prismatic joint biped robot using genetic algorithms," *Robotics and Autonomous Systems*, vol. 38, pp.119-128, 2002.
[8] P. Larranaga, J. A. Lozano, *Estimation of Distribution Algorithms. A New Tool for Evolutionary Computation*, Kluwer Academic Publishers, 2001.
[9] J. R. Greene, "Population-based incremental learning as a simple versatile tool for engineering optimization," *Proc. the first Int. Conf. on EC and Applications*, 1996, pp. 258-269.
[10] I. J. Schoenberg, "Contributions to the Problem of Approximation of Equidistant Data by Analytic Functions. Part A-On the Problem of Smoothing or Graduation. A First Class of Analytic Approximation Formulae", *Quarterly of Applied Mathematics*, vol. IV, no. 1, pp. 45-99, 1946.
[11] J.W. Grizzle, G. Abba, F. Plestan, "Asymptotically stable walking for biped robots: analysis via systems with impulse effects," *IEEE Trans. on Automatic Control*, vol. 46, No. 1, pp. 51-58, 2001.
[12] M. Vukobratovic, J. Stepanenko, "On the stability of anthropomorphic systems," *Math. Biosci.* vol. 15, no. 1, pp. 1-37, 1972.
[13] L. Hu, C. Zhou, Z. Sun, "Biped gait optimization using estimation of distribution algorithm," *Proc. of the 5th IEEE-RAS International Conference on Humanoid Robots*, Japan, pp. 283-289, 2005.
[14] P. Thévenaz, T. Blu, M. Unser, "Interpolation Revisited," *IEEE Trans. on Medical Imaging*, vol. 19, no. 7, pp. 739-758, 2000.
[15] L. Vecci, F. Piazza, A. Uncini, "Learning and approximation capabilities of adaptive spline activation function neural networks," *Neural Networks*. vol. 11, no. 2, pp. 259-270, 1998.
[16] T. Blu, M. Unser, "Quantitative Fourier analysis of approximation techniques: Part I—Interpolators and projectors," *IEEE Trans. on Signal Processing*, vol.47, no. 10, pp. 2783–2795, 1999.
[17] C. Zhou, P. K. Yue, "Robo-Erectus: a low cost autonomous humanoid soccer robot," *Advanced Robotics*, vol.18, no.7, pp. 717-720, 2004.