A mechanism of Particle Swarm Optimization on motor patterns in the B4LC system

QI LIU, JIE ZHAO, STEFFEN SCHUETZ, KARSTEN BERNS

Robotics Research Lab, University of Kaiserslautern,
Kaiserslautern, 67655, Germany

This paper presents an optimization mechanism for feed-forward control units termed as motor patterns in the Bio-inspired Behavior-Based Bipedal Locomotion Control (B4LC) system. In specific locomotion phases, motor patterns *ActiveHipSwing* and *LegPropel* produce torques at the hip and ankle joints respectively. Biped is activated to swing leg ahead and push body forward by corresponding motor patterns, in which the parameters determine the profile of generated torques according to the sigmoid function. To obtain efficient and stable locomotion control, we employ Particle Swarm Optimization (PSO) method to tune motor patterns' parameters by formulating locomotion stability, energy consumption and walking speed as fitness functions. The optimization procedure takes place on a 3-dimensional simulated bipedal robot. Simulation results prove that suggested approach reinforces the walking behavior of biped with respect to stability, velocity control as well as enhances the energy efficiency significantly.

*Keywords: Particle Swarm Optimization; Motor pattern; Locomotion control

1. Introduction and related work

As humans have possessed advanced skills in locomotion, the optimization of bipedal robots becomes an intensive issue that they are expected to achieve human-like behaviors with high efficiency and robust stability.

According to the employed locomotion control methods, the bipedal optimization approaches can be divided into two categories, which includes the model-based and model-free optimization. The model-based optimization is a computational approach that generates optimal motions or joint force profiles based on complex robot dynamic models.[1] In those issues, the classical control methods are normally used to plan the trajectory and mo-

2

tion. The optimization problems regarded as nonlinear programming problems are solved by utilizing efficient algorithms, *e. g.* sequential quadratic programming methods and augmented Lagrangian methods.[2]

Nevertheless, the challenge for model-based approaches is to establish complex dynamical models, and the bipeds developed correspondingly can hardly imitate the human behaviors in terms of stability and energy efficiency. Therefore, model-free optimization approaches have been proposed. The robot controlled by biologically-inspired methods can easily adapt to searching the parameters of these controller with stochastic optimization approaches. Shafii *et al.* has introduced a system that Particle Swarm Optimization (PSO) is used to optimize the angular trajectories of a stable biped walking.[3] Yang *et al.* has presented a method that stable gait generation in bipedal locomotion by using Genetic Algorithm (GA) optimization based Fourier Series Formulation.[4] This approach has the advantage over the classical control method on achieving human-like motions.

Inspired by humans locomotion, Luksch has addressed a behavior-based control framework, in which the bipedal control structure is organized in hierarchical levels by combining feed-forward and feedback control units, namely motor patterns and reflexes respectively.[5] However, due to the interaction of behaviors, the derivations of parameters incur intensive variations on the whole system. Because of this complex interconnectivity, the performance of bipedal robot in terms of stability, energy consumption are hardly optimized compared to humans behaviors. Since motor patterns play significant roles in stabilizing locomotion and generating propulsive torques for joints, such as swinging the leg forward, propelling the body and accepting the body weight, we will optimize the parameters of motor patterns based on PSO method in this work.

2. The control structure and motor pattern

As described in Fig. 1, the control structure defined by the flow of stimulation, inhibition and modulation among six classes of control units is organized in hierarchical levels.[5] The local control behaviors motor patterns generate torque commands to the corresponding joints in feed-forward manner. A sigmoid function is applied as basis for motor patterns, as depicted in Eq. 1:

$$\hat{\tau} = A \cdot \begin{cases} \frac{1}{2} + \frac{1}{2} \, sin(\, \pi(\frac{t}{T_1} - \frac{1}{2}) \,) & 0 \leq t < T_1 \\ 1 & T_1 \leq t < T_2 \\ \frac{1}{2} - \frac{1}{2} \, sin(\, \pi(\frac{t-T_2}{T_3-T_2} - \frac{1}{2}) \,) & T_2 \leq t < T_3 \end{cases} \qquad (1)$$

where A, T_1, T_2 and T_3 are the parameters of motor patterns representing the maximum torque of torque command, the starting time of maximum torque, the ending of maximum torque and the total time of torque command respectively. The values of parameters are defined experimentally by comparing the kinematic data of bipedal robot and humans. As shown in Fig. 1, the modification of the parameters results in a variety of torque trajectories.
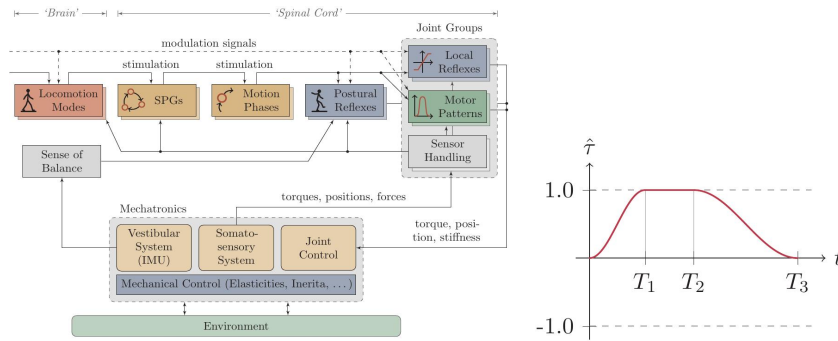

Fig. 1.   The left side of figure represents the hierarchical organization of control units, while the right side shows that torque trajectories produced by motor pattern.[5]

3. Particle swarm optimization algorithm

Inspired by the studies of fish and bird flocks, Kennedy and Eberhart first introduced the algorithm of PSO.[6] The movements of a particle is guided in the direction of its own best known position and the best know position of the entire swarm. The velocity of a particle in one iteration can be derived, as described in Eq. 2.

$$v_p = \omega v_p + c_1 \cdot Rand_1(\ B_p - x_p\ ) + c_2 \cdot Rand_2(\ B_g - x_p\ ) \qquad (2)$$

where v_p and x_p represent the velocity and the position of a particle respectively. B_p is the known best position of a particle in the past iterations, while B_g the known best position of entire swarm in the past iterations. c_1 and c_2 mean the parameters of the acceleration constants respectively. $Rand_1$ and $Rand_2$ are the random values ranged within $[0, 1]$. The inertia weight ω controls the impact of the previous velocities on the current velocity. The larger inertia weight tends to favor global searching, while the smaller one leads to the local searching strategy.

The fitness functions summarizing the distance of a given solution approaching to the objectives are calculated during an iteration. The best position of particles B_p and the best position of the entire swarm B_g are

4

updated by comparing the fitness values in each iteration. The position of a particle x_p is updated according to Eq. 3.

$$x_p = x_p + v_p \qquad (3)$$

4. Implementation of optimization in B4LC system

The optimization unit is built in a way that it can be integrated into an existing behavior-based control system. A behavior control unit termed as Optimization Module is inserted into the system by connecting the stimulated Motion Phase and the corresponding motor patterns, as shown in Fig. 2.

Activated by the stimulation signals from Motion Phase, the Optimization Module feeds generated outputs into each motor pattern as parameters T_1, T_2 and A. The cycling time of motor pattern T_3 is kept constant as introduced in.[5]
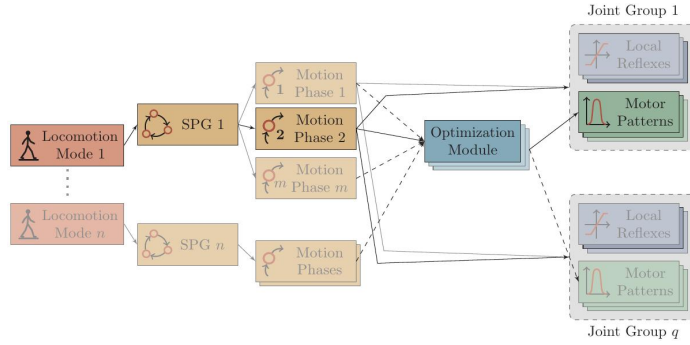

Fig. 2.   Optimization modules in the B4LC control system.

The parameters of motor pattern *Leg Propel* at the ankle joints and the motor pattern at the hip joints *Active Hip Swing* are optimized. A 6-dimensional vector $\vec{P}_{i,t}$ that consists of the parameters of *Leg Propel* $[T_{a1}, T_{a2}, A_a]$ and *Active Hip Swing* $[T_{h1}, T_{h2}, A_h]$ stands for the position of a particle, where i is the number of particles and t represents iteration times. The particle position are initialized randomly in the searching space. To increase the learning efficiency, the searching space are constrained in a range based on the experimental studies of bipedal locomotion, as depicted in Eq. 4.

$$
\begin{aligned}
T_{a1} \in [0.0 ~,~ 0.6] \qquad T_{a2} \in [0.6 ~,~ 0.9] \qquad A_a \in [0.0 ~,~ 1.0] \\
T_{h1} \in [0.0 ~,~ 0.4] \qquad T_{h2} \in [0.4 ~,~ 0.7] \qquad A_h \in [0.0 ~,~ 1.0]
\end{aligned}
\qquad (4)
$$

To achieve a natural walking while maintaining stability, the fitness functions are formulated by considering optimization goals including robustness,

stability, and speed and energy consumption in form of:

$$F = k_1 f_1 + k_2 f_2 + k_3 f_3 + k_4 f_4 \tag{5}$$

where k_1, k_2, k_3 and k_4 are the weightings of the fitness functions, and

$$f_1 = s_w / s_{max} \tag{6}$$

$$f_2 = \sum_{i=0}^{T} (\Delta Xcom_r + \Delta Xcom_l)/s_w \tag{7}$$

$$f_3 = \sum_{i=0}^{T} |(V_{ref} - V_i)|^2 / T \tag{8}$$

$$f_4 = \sum_{i=0}^{T} \tau_i^h / T \tag{9}$$

f_1 is defined as the function that is linear to the walking steps s_w with respect to maximum walking steps s_{max} the robot can achieve. f_2 evaluates the stability of the robot by taking the error value of extrapolated Center of Mass ($Xcom$) into consideration, where $\Delta Xcom_r$ and $\Delta Xcom_l$ are the right and left error value at each time step respectively, and T is the maximum time step in one iteration. Furthermore, f_3 represents the deviation of the walking speed respecting reference, with V_i meaning the walking speed at time step i, and V_{ref} representing the reference equals 1.2m/s. In addition, f_4 contains the accumulated resistant torques generated by the reflex *Lock Hip*, which is utilized to stop the flexion of hip joints when swinging legs moved by *Active Hip Swing* to a desired angle. τ_i^h means the braking torques generated by *Lock Hip* at time step i.

5. Optimization process and setup

The optimization scenario is set up on the simulated robot with 21 degrees of freedom and 1.8m height.[7] The robot is designed to start the experiments with a stable walking on a flat ground. The parameters generated

Table 1.   The parameters before and after optimization.

parameters	before optimization	after optimization
Leg Propel T_1	0.5	0.3
Leg Propel T_2	0.7	0.75
Leg Propel A	0.65	0.47
Active Hip Swing T_1	0.2	0.09
Active Hip Swing T_2	0.56	0.65
Active Hip Swing A	0.7	0.41

by Optimization Module known as the position of a particle are fed into

6

Leg Propel and *Active Hip Swing*. When the pitch and roll angle of the robot exceed thresholds or a robust walking behavior emerges after successful walking in long time, a test of a particle is finished. Until all the particles in one iteration are tested, the best position of the entire swarm and the best position of each particle are updated. Consequently, the velocities and the positions of particles start a new iteration in accordance with updates.
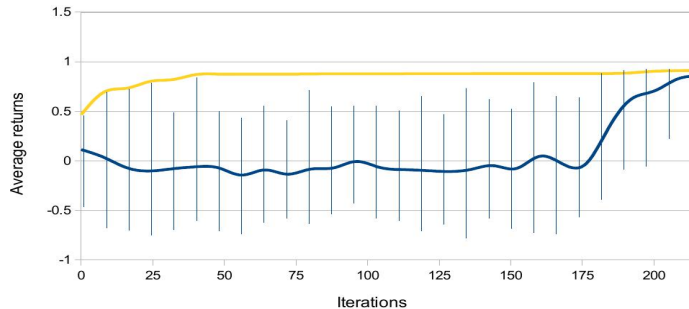

Fig. 3.   The figure shows the average returns of fitness values during optimization progress. The yellow line presents the average fitness value of the known best particles in the past iterations. The blue line indicates the average fitness value of particles in each iteration. The vertical lines stand for the maximum and minimum value of the fitness value in each iteration.

The average fitness values for each iteration are shown in Fig. 3. At the beginning, we use the high inertia weight, which is equal to 1, to perform global searching. After about $50th$ iteration, the fitness values of the best known particles are constant, and the average fitness value of the particles has no more significant improvement. To explicitly search the optimal parameters, we decrease the inertia weight after 160 iterations to perform local searching. The average fitness value in each iteration is sharply increased and navigates closer to the best known fitness value, and the robot is able to perform stable and robust walking. The parameters before and after optimization are shown in Table 1.

6. Simulation results

To compare the bipedal performance after and before optimization approach, we have implemented 30 times of walking experiments with optimized and non-optimized parameters respectively. The motor torques generated at the ankle and hip joint are illustrated in Fig. 4. The data is normalized to one gait cycle and averaged over 30 successive steps of walking on level ground. The torque profiles present smaller peek during activation of *Leg Propel* and *Active Hip Swing* after optimization. This leads to a

longer swing phase and shorter stance phase compared to the robot with non-optimized parameters.
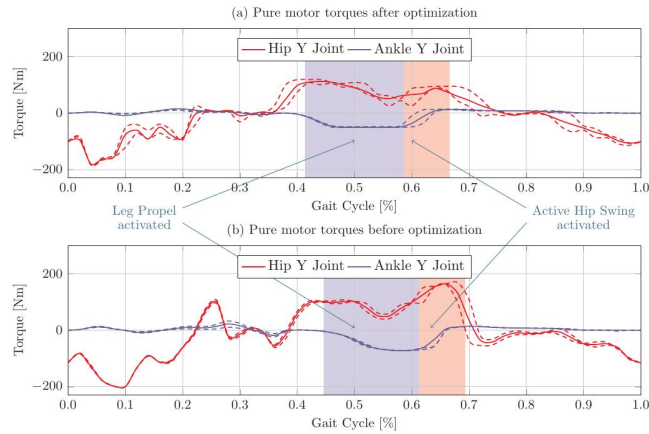

Fig. 4.   The motor torques generated from hip joints around y-axis and ankle joints around y-axis using optimized and non-optimized parameters over the course of one gait cycle. The solid line presents the mean value over 30 successive steps. The dashed lines stands for the minimum and maximum values. The blue area and the red area indicate the activation of the Leg Propel and Active Hip Swing respectively.

Because of the improper torques generated, the average walking speed is not able to follow the reference. It can be seen from Fig. 5 that the robot is initialized to walk at the speed of 1.2m/s. However, as walking distance increases, the speed enhances to around 1.3m/s. This result in major increase of velocity deviation calculated in Eq. 8.
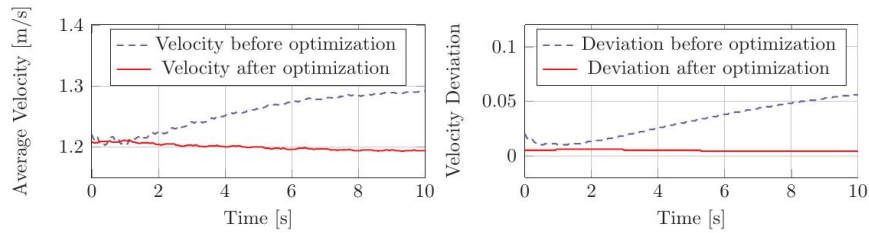

Fig. 5.   The left side of figure shows the pitch angle before and after optimization, while the right side presents the average walking speed and the speed deviations with optimized and non-optimized parameters.

In term of energy consumption, the result of locking-hip torques both after and before optimization progress are shown in Fig. 6. By using non-optimized parameters, the robot boosts the leg in a shorter duration to the desired position. The robot with optimized parameters generates less

torques to prevent over-swinging of the leg to achieve a suitable angle of attack at heel strike. One can observe that the angle of attack at hip joint dwindles since less energy is generated from Leg Swing and therefore the over-swing at hip joints appears to be insignificant compared to previous.
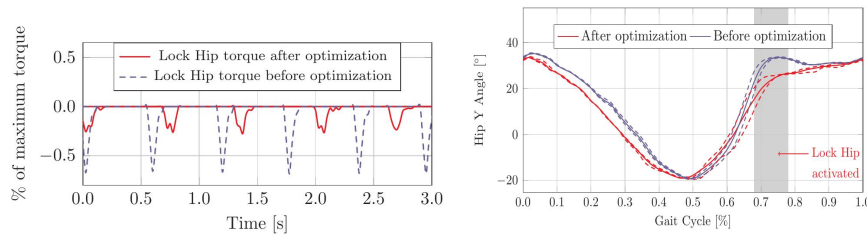

Fig. 6. The left side of figure indicates the percentages of the torque commands generated from Lock Hip reflex over the maximum hip torque; the right side of figure shows the hip angular trajectories around y-axis over the course of one gait cycle.

7. Conclusion

In this paper we described the development of a PSO-based optimization approach for motor patterns in the B4LC system. The simulated biped is optimized in terms of stability, energy consumption and walking velocity. The optimized parameters have been used to conduct the normal walking experiments. The locomotion performance with optimized parameters have been compared to that with non-optimized parameters.

References

1. M. Srinivasan, "Computer optimization of a minimal biped model discovers walking and running", Nature 439.7072 (2006): 72-75.
2. Y. Xiang, "Physics-based modeling and simulation of human walking: a review of optimization-based and other approaches", Structural and Multidisciplinary Optimization 42.1 (2010): 1-23.
3. N. Shafii, "Biped walking using coronal and sagittal movements based on truncated Fourier series", RoboCup 2010: Robot Soccer World Cup XIV. Springer Berlin Heidelberg, 2011. 324-335..
4. L. Yang, "A uniform biped gait generator with offline optimization and online adjustable parameters", Robotica 25.05 (2007): 549-565.
5. T. Luksch, "Human-like control of dynamically walking bipedal robots", Verlag Dr. Hut, 2010.
6. R. Eberhart, "A new optimizer using particle swarm theory", Proceedings of the sixth international symposium on micro machine and human science. Vol. 1. 1995.
7. J. Zhao, "Biologically motivated push recovery strategies for a 3D bipedal robot walking in complex environments", Robotics and Biomimetics (ROBIO), 2013 IEEE International Conference on. IEEE, 2013.