J Intell Robot Syst (2012) 67:255–270
DOI 10.1007/s10846-012-9661-1

Smooth transition between different gaits of a hexapod robot via a central pattern generators algorithm

Weihai Chen · Guanjiao Ren · Jianbin Zhang · Jianhua Wang

Received: 18 July 2011 / Accepted: 16 February 2012 / Published online: 7 March 2012

Abstract This paper focuses on the topic of smooth gait transition of a hexapod robot by a proposed central pattern generator (CPG) algorithm. Through analyzing the movement characteristics of the real insects, it is easy to generate kinds of gait patterns and achieve their smooth transition if we employ a series of oscillations with adjustable phase lag. Based on this concept, a CPG model is proposed, which is constructed by an isochronous oscillators and several first-order low-pass filters. As an application, a hexapod robot and its locomotion control are introduced by converting the CPG signal to robot's joint space. Simulation and real world experiment are completed to demonstrate the validity of the proposed CPG model. Through measuring the position of the body center and the distance between footpoints and ground, the smooth gait transition can be achieved so that the effectiveness of the proposed method is verified.

Keywords CPG · Multi-legged robot · Gait · Smooth transition · Phase lag

This work is supported by National Natural Science Foundation of China (NSFC) under the research project 61175108 and National 863 Program of China under the research project 2011AA040902. This work is also supported by the Innovation Foundation of BUAA for PhD Graduates.

Electronic supplementary material The online version of this article (doi:10.1007/s10846-012-9661-1) contains supplementary material, which is available to authorized users.

W. H. Chen · G. J. Ren (✉) · J. H. Wang
School of Automation Science and Electrical Engineering, Beijing University of Aeronautics and Astronautics, Beijing 100191, China
e-mail: sqrgj@163.com

W. H. Chen
e-mail: whchenbuaa@126.com

J. H. Wang
e-mail: jhwangbuaa@126.com

J. B. Zhang
School of Mechanical Engineering and Automation, Beijing University of Aeronautics and Astronautics, Beijing 100191, China
e-mail: jbzhangbuaa@126.com

1 Introduction

Legged locomotion is very common in nature world. For instance, human beings have two legs, mammals walk with four legs, and insects perform their locomotion by employing six legs. The most important advantage of legged movement, compared with wheeled or tracked movement, is the adaptability to various kinds of terrains. However, the coordination and control of the legs are difficult issues to solve. Due to this reason, much eyes of the scholars are attracted to find a better solution of this problem, see [1–3].

The traditional method is to scheme out the trajectory of the footpoint and calculate the inverse kinematics of the leg mechanism [4–6]. But this algorithm has some disadvantages such as large computational complexity and demanding of precise mechanical model. When a leg gets stuck by an obstacle, the robot has to stop the leg, detect the current position and orientation, and recalculate the trajectory. It wastes a lot computational resource.

Animals will not do this. In recent years, applying the mechanism in animal's central neural system to control robot's locomotion is becoming more and more popular. It is found that, animal's locomotion is controlled by a series of central oscillation generated by the spinal cord for the vertebrate, or by the ganglion for the invertebrate, which is called central pattern generator (CPG) [7–9]. The CPG model is at the earliest proposed by Cohen [10] in 1980s through the research on the dissection of a lamprey spinal cord. There is now very clear evidence that rhythms are generated centrally without requiring sensory information. This is how the word "central" comes from.

After that, many researchers apply this algorithm into the bio-inspired robot control. For example, Kimura uses the Matsuoka neuron models [11], with two neurons mutually inhibiting, to generate the oscillation and to control the walking of a quadruped robot [12–14]. The CPG is applied to the musculo-skeletal system by Taga to perform a bipedal locomotion [15]. Arena provides a multi-templates approach of cellular nonlinear networks (MTA-CNN) for the implementation of CPG to the hexapod movement [16–18]. Inagaki constructs a wave CPG model to control his hexapod robot [19, 20]. Manoonpong [21] achieves some high level behaviors in his robot such as reflex and escape through the sensor-driven neural control. What has to be mentioned is the impressive work accomplished by Ijspeert [22]. Nonlinear oscillator is used for the neural circuit and several oscillators are coupled together to construct the CPG model. His salamander robot can walk like a quadruped animal on the land and swim like a snake in the water.

However, as to this research area, most of the previous works doesn't take gait transition into account, or they only consider the gait patterns separately, that is, to consider different gaits as different patterns. For instance, Arena uses template to define gait. Every gait pattern is related to a pre-defined template, so his hexapod robot can only perform some pre-defined gait pattern. The quadruped robot designed by Kimura can move in walk, trot, pace and gallop gait, but the implementation of each gait need adjusting the oscillator's parameters, which are nonlinear and irregular to follow. Manoonpong's hexapod robot [21] can walk omnidirectionally but with only tripod gait. The latest work done by him can perform different kinds of gaits by chaos control [23], but the model is complicated. The author gets inspired from Ijspeert's salamander robot. It can switch from walking to swimming when entering the water by just progressively increasing one drive signal. The limb oscillators saturate and stop swinging so that the body is propelled by the twist of trunk. As Ijspeert mentioned in the review paper [24], CPG-based gaits transition remains an open topic. Often simple electrical stimulation of a particular region of the brain stem in animals can induce dramatic gait changes, so, smooth gait transition of the robot via regulating very few parameters should be paid more attention, which merits our study in this paper.

Insects are selected as the imitating model to control the robot's locomotion. The reason we choose insects is because they are so primitive that the neural system seems relative simple and the stability of the six-legged locomotion is better than four-legged. We hold that the description of different gaits should be formally uniform in essence and shouldn't be considered separately. The common point of the previous CPG models is that, they are all actually kinds of nonlinear dynamical system. Our model is also constructed by a dynamical system with limit cycle attractor. Different from others' previous works, it is relative simple to understand, which is constructed by an isochronous oscillators and several first-order low-pass filters. The filters are actually an extension of the central oscillation, just like the wave spreading. Different gaits generation are controlled only by one parameter – phase lag – which resembles the electrical stimulation at the brain stem. In addition, some seemingly complex gaits, of which the phase lags are actually not the

typical angle, are easily generated. This algorithm can simply achieve smooth transition between different gaits, compared with the previous works.

The article is organized as follows. Firstly we analyze the gaits of the real insects and find the relationship between gait and CPG. Secondly, the specification of our CPG model is described in detail. After that, the mechanism of the walking machine is stated and the definitions of the links and joints are given; meanwhile, how to convert the CPG signal to the angle value used to control the joints is addressed. Then, simulation and experiment are accomplished where three gaits and the smooth transition between them are illustrated, followed by the conclusions.

2 Gait analysis for insects

Animals are capable of using kinds of gaits. For instance, cat can easily vary its gait from walk to trot to gallop. Insects can perform wave gait, tetrapod gait, transition gait, tripod gait and so on. Moreover, when the insects move not so fast and not so slowly, it can show some gaits between the typical patterns: sometimes one leg transfers while sometimes two, and these statuses always mix together. It is difficult to define and describe all the gaits that insects always use.

The apparent complexity of insect walking behavior has stimulated considerable researches. Scholars have established sophisticated models to describe it, in which the most famous two are proposed by Hughs [25] and Wilson [26]. The two models are the same in essence, while the Wilson's model is more concise [27]. It satisfies the following five rules:

1. A wave of recoveries runs from rear to front (and no leg recovers until the one behind is place under a supporting position).
2. Contralateral legs of the same segment alternate in phase.
3. Recovery time is constant.
4. Frequency varies (drive time decreased as frequency increases).
5. The intervals between steps of the hind leg and the middle leg and between the middle leg and foreleg are constant, while the interval between the foreleg and hind leg steps varies inversely with frequency.

The first rule means the rear leg, rather than the front leg, recovers first. It is decided by the stability and more details can be found in [27]. Rule 2 means, for instance, the right front (RF) leg has a 180° delay from the left front (LF) leg and vice versa. This rule is crucial for the CPG method in robot locomotion control, which will be discussed later. Rule 4 means the moving velocity can be changed from regulating frequency. Even though rule 3 and rule 5 are truth for real insects, the recovery time and the interval between steps could be changed in robotic control to achieve speed regulating.

Through the five rules, the various kinds of insect gaits can be described by the gait timing diagrams, as shown in Fig. 1. We can define duty factor β as the fraction of the cycle time that each foot is on the ground. In the figure, black areas indicate transfer (recover) phase and white areas indicate drive (support) phase. The wave recovers from rear to front; therefore, we can consider the wave spreads from right hind (RH) leg to right middle (RM) leg, then right front (RF), left hind (LH), left middle (LM) and left front (LF). Figure 1 shows the four typical gaits with the increasing of walking speed, in which the duty factors satisfy $\beta = \frac{5}{6}$, $\beta = \frac{3}{4}$, $\beta = \frac{2}{3}$ and $\beta = \frac{1}{2}$. We name them as wave gait, tetrapod gait, transition gait and tripod gait, respectively.

If we use the traditional method to control the hexapod robot's locomotion, we have to plan out the footpoint trajectory and the velocity for transfer phase and support phase, respectively [28]. Inverse kinematics has to be calculated after that. If the gait is changed, all the data have to be recalculated again. Moreover, there are many other not so regular gait patterns, which could not be planned out.

When we resort to the CPG method, the locomotion control problem and gait transition can be simplified. For the insects, the ganglia can generate a series of oscillations spontaneously. For the robot, the neural oscillations, which inhibit mutually, can control the motors directly without computing the kinematics. Take $\beta = \frac{5}{6}$ for example, if

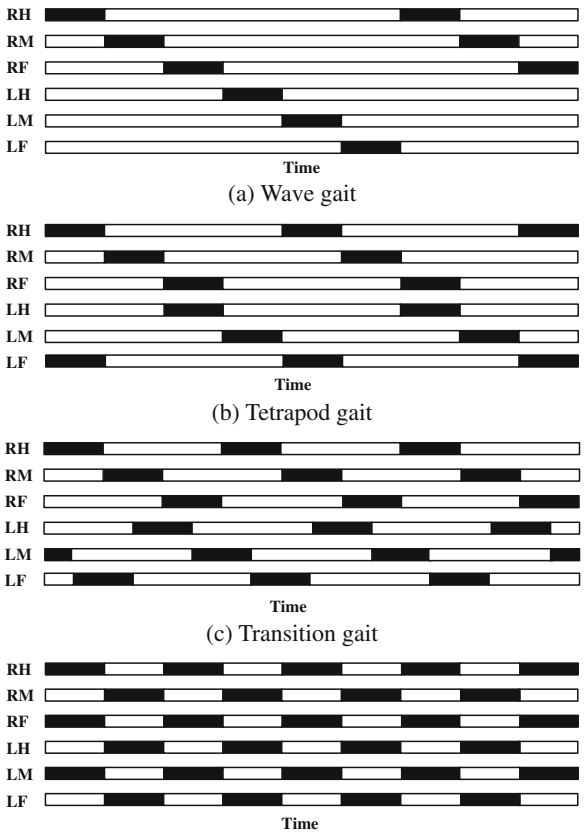


Fig. 1 Four typical gait diagrams arranged by the walking speed, from slow to fast

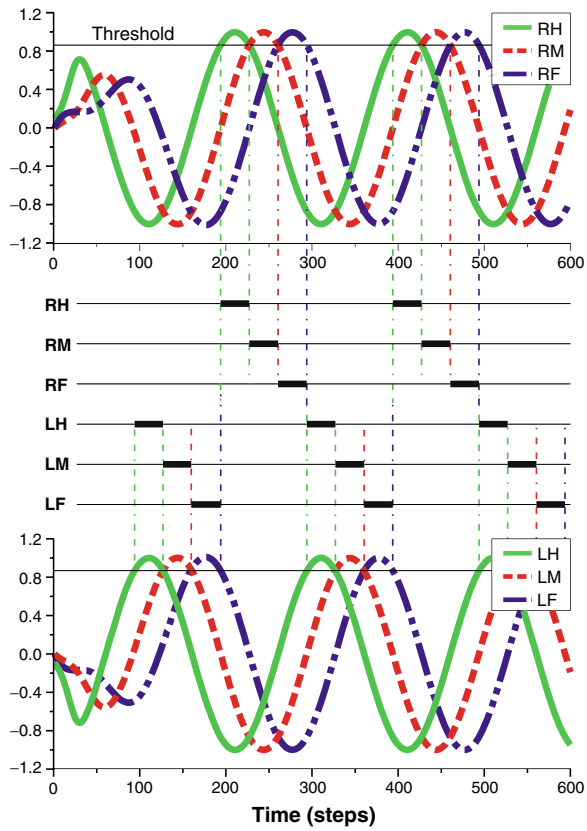


Fig. 2 The wave gait implementation, $\beta = \frac{5}{6}$. The *top figure* indicates the CPG signals, which can be used to control the three right legs. The signals of *bottom figure* is antiphase to the top figure's, which can be used to control the corresponding left legs. The *middle figure* is the timing diagrams of wave gait

three sine-like waves with 60° phase lag between each of them are obtained, the typical wave gait can be shown. As shown in Fig. 2, suppose the wave exceeds a threshold, this leg enters a transfer phase. When the wave value falls down the threshold, the leg begins its support phase again. The three waves are applied to control the three legs at the same side, from rear to front. According to the rule 2 of the Wilson's model discussed above, the other three legs, which are at the other side, are antiphase to the corresponding right legs or have a 180° lag, respectively.

As illustrated in Fig. 3, a tetrapod gait pattern like Fig. 1b can be obtained using the same approach. So are the other two gait patterns.

That is to say, different gaits are decided just by the phase lag between the three oscillation and the corresponding thresholds. The relationship between the phase lag, duty factor and the threshold forms Table 1. If the phase lag is less than 60°, there is at most one leg transferring at any time. When the phase leg is between 60° and 120°, the transfer leg number is one or two. The situation that three legs swing simultaneously emerges if the phase leg is greater than 120°. The 180° indicates the tripod gait. There are some reasons to believe that tripod gait is the fastest gait for insects, because the insects are too primitive to maintain their body in dynamic balance. Therefore, we don't need to discuss the dynamic balance gait, i.e., the phase lag is greater than 180°. If we want to generate a kind of gait not in the above four, we just need to choose a appropriate phase lag.

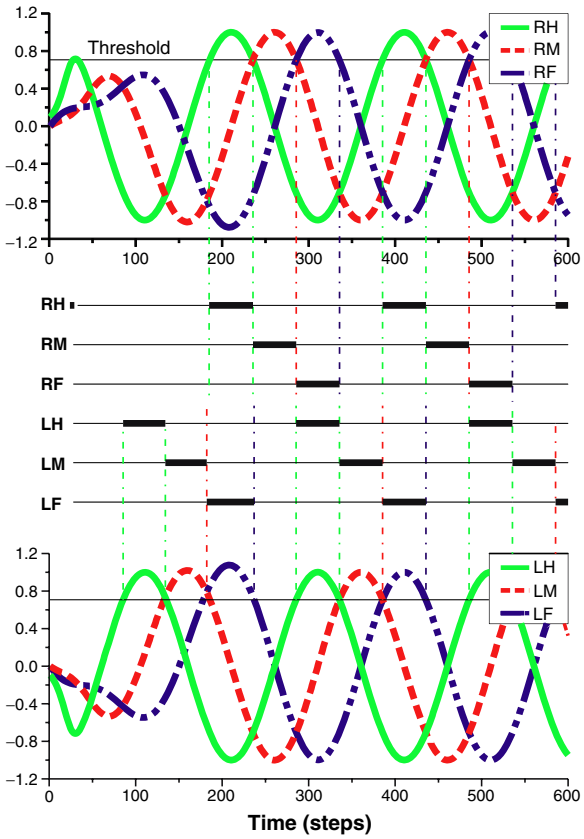


Fig. 3 This figure shows how the tetrapod gait ($\beta = \frac{3}{4}$) is generated from the series of waves which have a 90° phase lag

Through the above analysis, we can draw the following conclusions:

1. Three waves are enough to generate different gait patterns. The control signal of the other three legs can be achieved via reversing the waves of the contralateral legs.
2. The phase lag between the three waves determines the different gait patterns. With the increasing of the phase lag from 0 to 180, number of the transfer legs increases.

Table 1 The volume of fast workspace

Gait	Phase lag (deg)	Duty factor	Threshold
Wave	60	5/6	0.8660
Tetrapod	90	3/4	0.7071
Transition	120	2/3	0.5000
Tripod	180	1/2	0.0000

3. The threshold corresponds to the phase lag.
4. There are two approaches to accelerate the robot. One is increasing the frequency of oscillation. The other approach is enlarging the amplitude in order to generate a larger stride for the robot.
5. We need to establish a CPG model, which contains the following features: It oscillates spontaneously; it contains three series of wave; the phase lag, frequency and amplitude can be adjusted independently.

3 The central pattern generator model

3.1 The central oscillation

There are a lot of models to generate the central oscillation as Section 1 introduced. The common point of the different models is that they all have a limit cycle shown in phase portrait, which is asymptotically stable. Wherever the initial value of the ODE (Ordinary Differential Equations) is, the solution would converge to the limit cycle eventually. The integral curves of the system appear to oscillate. Here we choose a simple one, isochronous oscillator [29], as CPG. It is expressed in Eq. 1

$$\begin{cases} \dot{\varphi} = \omega \bmod 2\pi \\ \dot{r} = r\left(\mu - r^2\right) \end{cases} \tag{1}$$

where φ indicates the phase and r indicates the amplitude. ω represents the frequency of the oscillation. $\dot{\varphi}$ equals to ω means the phase will change at the rate of ω. When $r^2 > \mu$, $\dot{r} < 0$, so r decreases. On the contrary, when $r^2 < \mu$, $\dot{r} > 0$, so r increases. Therefore, $r = \sqrt{\mu}$ is a fixed point.

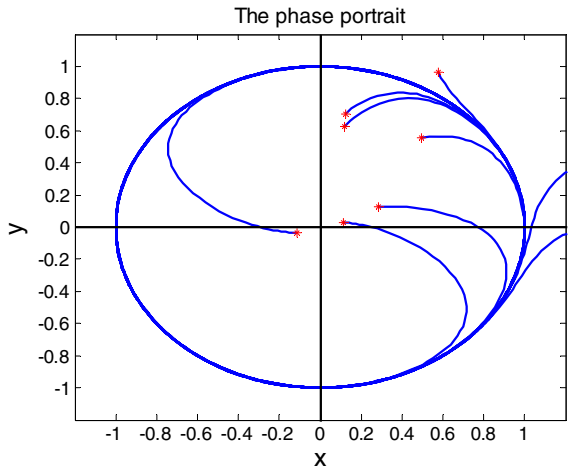
It is obviously that the above two equations are decoupled, so that the phase and amplitude can be controlled independently, which satisfies our requirement discussed in the last section. Actually, Eq. 1 is defined not on the Euclidean $\mathbb{R}^2$ space but on $\mathbb{S}^1 \times \mathbb{R}^1$. It is a description of polar coordinate. The description of the oscillator in Cartesian coordinate is expressed in Eq. 2.

$$\begin{cases} \dot{x} = \left(\mu - x^2 - y^2\right) x + \omega y \\ \dot{y} = \left(\mu - x^2 - y^2\right) y - \omega x \end{cases} \tag{2}$$

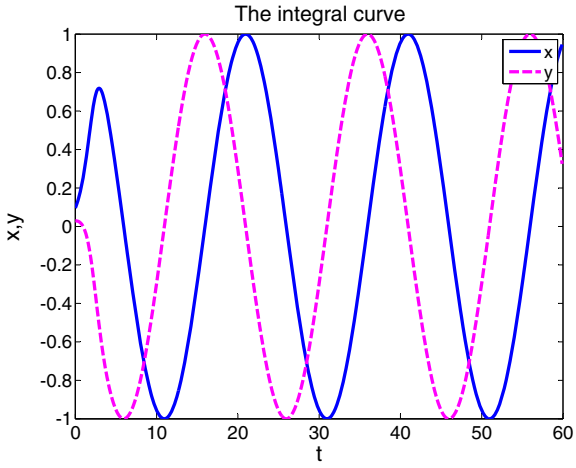
It is a typical Hopf oscillator. The phase portrait and integral curve are shown in Fig. 4.

3.2 The controllable phase lag

Through the above analysis, the amplitude and frequency can be controlled independently by the isochronous oscillator. In this subsection, we introduce the approach to control the phase lag.



(a) The phase portrait



(b) The integral curve

Fig. 4 The property of isochronous oscillator. (**a**) The *red star* means the initial value. It is obvious that no matter what initial value it is, it will converge to the limit cycle. (**b**) It shows the system can oscillate stably. At this figure, μ is 1 and $\omega = -\frac{\pi}{10}$

First-order low-pass filters are employed to accomplish this requirement. The transfer function of first-order low-pass filter is presented in Eq. 3

$$U(s) = \frac{1}{\tau s + 1} \tag{3}$$

where, τ is a time constant. The frequency domain expression is presented in Eq. 4.

$$U(j\omega) = \frac{1}{j\tau\omega + 1} = \frac{1}{\sqrt{\tau^2\omega^2 + 1}} \angle - \arctan(\tau\omega) \tag{4}$$

There is a reduction of the gain. We have to compensate it. The low-pass filter with gain can be written as Eq. 5

$$U(s) = \frac{k}{\tau s + 1} \tag{5}$$

where k equals $\sqrt{\tau^2\omega^2 + 1}$. In this way, the amplitude of the input can hold, while the time delay of the component can be controlled by adjusting the time constant τ. Furthermore, the frequency ω need to be negative in order to make the phase following the input, rather than going ahead of it.

One first-order low-pass filter can perform $(-90°, 90°)$ phase lag but exact 90° cannot be reached; hence, three filters are employed to achieve $[0°, 180°]$ phase lag. In order to make the format uniform, we rewrite the three first-order low-pass filter to ODE style, which is expressed in Eq. 6.

$$\begin{cases} \dot{x}_1 = (-x_1 + k u_{in})/\tau \\ \dot{x}_2 = (-x_2 + k x_1)/\tau \\ \dot{x}_3 = (-x_3 + k x_2)/\tau \end{cases} \tag{6}$$

Therefore, the controllable phase lag can be obtained. For example, when $\omega = -\frac{\pi}{10}$, the robot need a 180° phase lag to perform tripod gait, i.e., each filter has to generate a 60° delay. Suppose we set τ to $(\tan(-\frac{\pi}{3})/\omega)$, which is about 5.513, tripod gait can be generated.

3.3 Model discussion

Our CPG model is simple compared with other researchers' models, such as in [16] and [30], which are neural oscillators or each parameter has physical meaning, e.g. membrane potential or synaptic

weight. The reason we do not select a model founded from real insects' neuron as the CPG unit is, the robot would not be totally the same as real insects, especially in the actuators. The robot is always driven by motors or hydraulic devices or pneumatic artificial muscles, while insects are driven by real muscles, which have higher energy efficiency. Moreover, the robot always employs six identical legs, but each leg of insect is different. Our model extracts the essential characteristics of the neurons, such as spontaneous oscillating, disturbance avoiding, etc., which is more suitable for the robot control. The authors' viewpoint is similar to the one in [31]: a strict bio-mimicry strategy is still not appropriate to the application to robot.

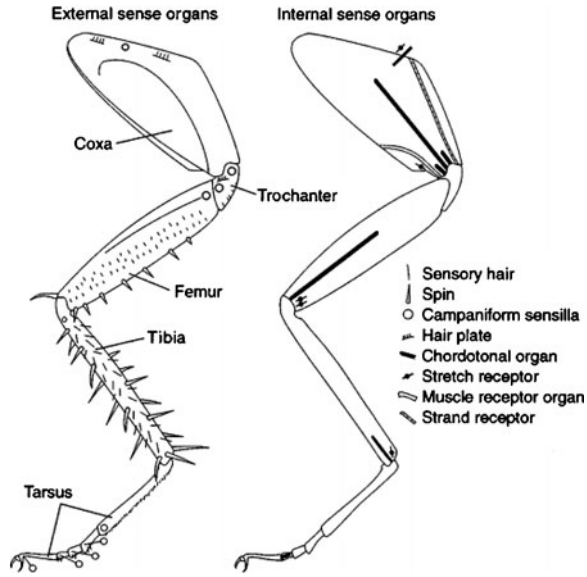


Fig. 5 Structure of the rear leg of a cockroach [7]

4 Conversion of CPG signal to angular position

From the above analysis, we can obtain the central oscillations of different gait patterns. So far, the signal still cannot be applied to control the robot or motors directly. They have to be converted to joint space.

4.1 Mechanism design of the robot

In order to explore the performance of CPG control in physical systems, a walking machine is designed out to accomplish simulation and experiment. The leg of real insect [7] contains the following segments: Coxa, Trochanter, Femur, Tibia and Tarsus (see Fig. 5). To imitate the leg structure of the real insect and satisfy the engineering requirement, our hexapod walking machine consists of six identical legs. Each leg has three joints (three DOFs), as shown in Fig. 6. We define them in step with Manoonpong [32]: the thoraco-coxal (TC-) joint enables forward and backward movements, the coxa-trochateral (CTr-) joint enables elevation and depression of the leg, and the femur-tibia (FTi-) joint enables extension and flexion of the tibia. It can be seen that the morphology of this multi-jointed leg is modeled on the basis of a stick insect or cockroach leg but tarsus segments are ignored. Every joint is a rotary DOF, and the footpoint can be regarded as a 3-DOF spherical joint. The mechanical structure of the robot is shown in Fig. 7, and the details of the leg are shown in Fig. 8.

4.2 Angle value of TC- joint

The leg movement in one cycle can be divided into four processes, as shown in Fig. 9. Process 1, 2 indicate the transfer phase and Process 3, 4 indicate the support phase. When this leg is at its transfer phase, it swings to the anterior extreme position (AEP) then it begins its support phase and moves backward. Because of the counter force (frictional force), the body is propelled to move forward until the leg reaching its posterior extreme position (PEP).

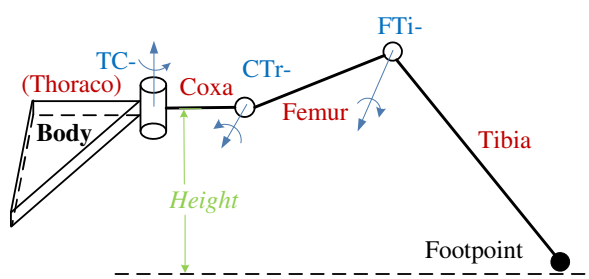


Fig. 6 The model of leg mechanism

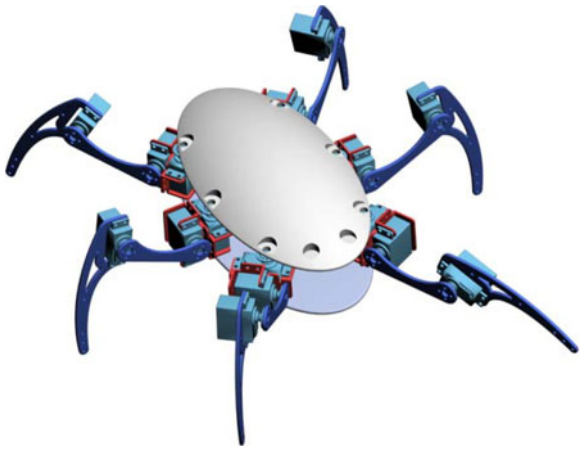


Fig. 7 The prototype of our hexapod robot

For the TC- joint, it is at positive max value when the footpoint of the leg is at AEP and at negative max value when leg is at PEP. The periods or frequencies for transfer phase and for support phase are not always the same in different gait pattern, although they together are related to the global frequency ω. The TC- joint angles must satisfy Fig. 10. When the leg is in Process 1, the joint angle recovers from the negative max value to zero. Then, it increases to the positive max value at Process 2. The joint angle decreased at Process 3 and 4 so as to perform the support phase.

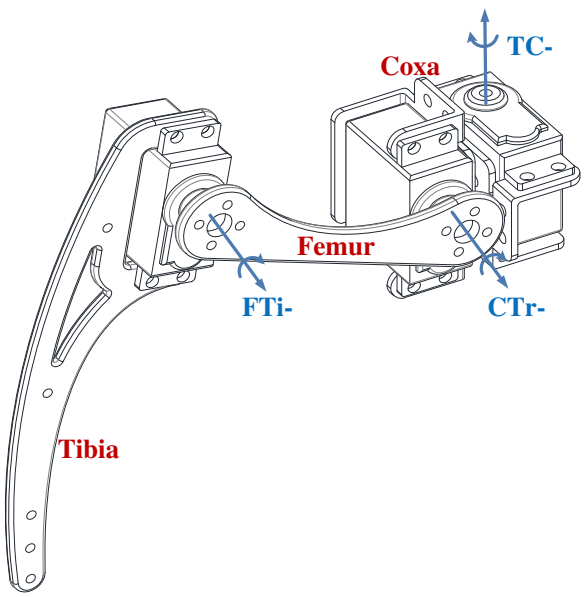


Fig. 8 The leg mechanism and the corresponding joints and links

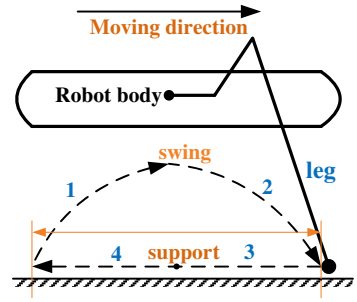


Fig. 9 The illustration of one leg movement cycle

Now, we come to the question that, how to convert the CPG signal to the angle value of TC-joint?

Firstly, let us focus on the transfer phase. It can only spend $(1-\beta)$ of one CPG cycle for the leg to finish its transfer phase. Therefore, the period of the transfer phase satisfies Eq. 7

$$\frac{1}{2}T_t = (1-\beta)T \tag{7}$$

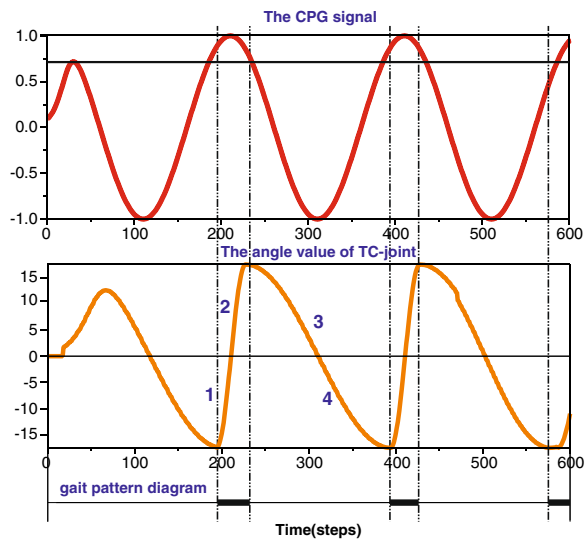


Fig. 10 The *top red wave* indicates the CPG signal generated by the central oscillation. The *middle orange wave* indicates the joint angle the TC- joint can reach. In one period, 1,2 is the transfer phase and 3,4 is the support phase, as shown in gait pattern diagram. In this figure, $\beta = \frac{3}{4}$, i.e. 90° phase delay

where T_t is the new period of the transfer phase (actually the transfer phase only occupies half of the period), and T is the original period of the CPG signal. The frequency of the transfer phase ω_t can be expressed in Eq. 8.

$$\omega_t = \frac{2\pi}{T_t} = \frac{2\pi}{2(1-\beta)T} = \frac{\omega}{2(1-\beta)} \tag{8}$$

The value of TC- joint should be at the negative max value when the leg enters the transfer phase. However, it equals the threshold value in original CPG. So the phase angle of joint value φ_t should be pulled ahead for some time, which is expressed in Eq. 9.

$$\varphi_t = \left(\frac{1}{2} - \frac{1-\beta}{2}\right) T = \frac{\beta}{2} T \tag{9}$$

The amplitude should be adjusted corresponding to the mechanical property in case of interference. Take $\beta = \frac{3}{4}$ for example, ω_t is 2ω and φ_t is $\frac{3}{8}T$. The oscillation is actually a series of discrete points; therefore, the new series of points, which are used to control the TC- joint angle, can be obtained by: (a) choosing a point $\frac{3}{8}T$ before the current point, (b) skipping one point and choosing another. The process is shown in Fig. 11.

Then, let us cope with the support phase. The analysis method is the same as the transfer phase. The frequency of the support phase ω_s is presented in Eq. 10.

$$\omega_s = \frac{\omega}{2\beta} \tag{10}$$

Compared with the original point series, the phase angle should be pulled ahead for φ_s, which is expressed in Eq. 11.

$$\varphi_s = \frac{1-\beta}{2} T \tag{11}$$

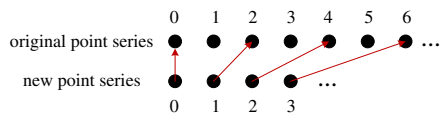


Fig. 11 This figure shows how to generate the new point series in transfer phase. The frequency of new point series is twice the original series. What need to be point out is, the new point series have been pulled ahead for $\frac{3}{8}T$

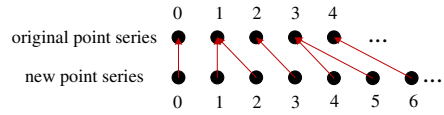


Fig. 12 This figure shows how to generate the new series of points in support phase. The frequency of new point series is $\frac{2}{3}$ the original series. The new series have been pulled ahead for $\frac{1}{8}T$

Assuming $\beta = \frac{3}{4}$, $\omega_s = \frac{2}{3}\omega$ and $\varphi_s = \frac{1}{8}T$. The start point of the new point series should be $\frac{1}{8}T$ before the original series. Because the new frequency is less than the original frequency, several new points share one original point in some situations, see Fig. 12. Then a series of wave like Fig. 10 shown is formulated, which can be applied to control the TC- joint.

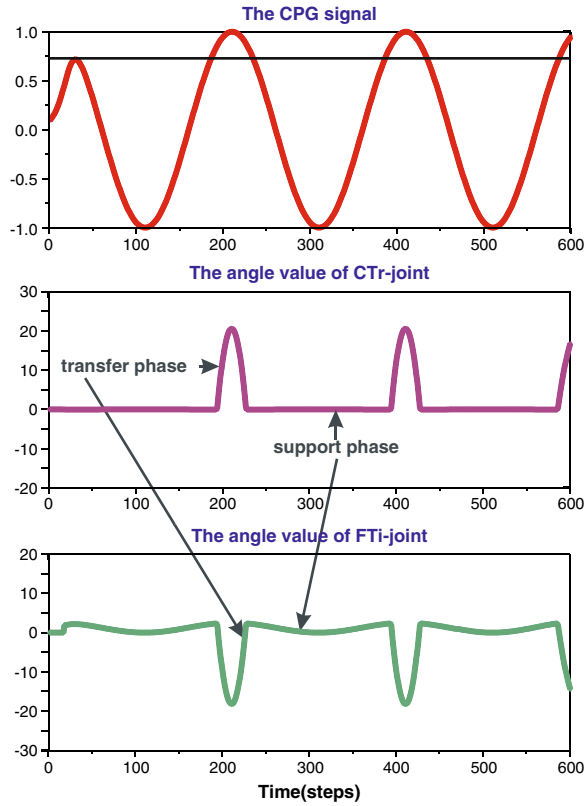


Fig. 13 The angle value of the CTr- joint and FTi- joint, which represent the real angle of the two joints. It is obviously the movement in transfer phase is intense than the support phase for this two joints

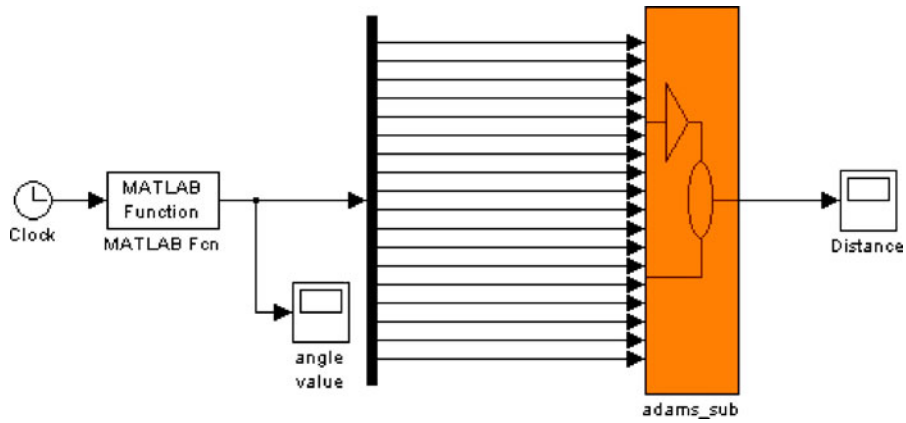


Fig. 14 The control system structure of the co-simulation shown in Simulinks. The *orange module* indicates the model in ADAMS. The MATLAB Fcn is used to generate the angle values of 18 motions. Two scopes are used to watch the joint angles and the body center distance respectively

4.3 Angle value of CTr- and FTi- joint

The approach to obtain the joint angles of CTr- and FTi- is the same as the approach of TC- joint. It can be divided into the steps below:

1. Consider the transfer phase and support phase separately;
2. Calculate the period or frequency for transfer phase and support phase;
3. Calculate the phase angle of the new series of points;
4. Adjust the amplitude due to the mechanical property, i.e., the length of the links. Examples of the actual value of the two joints are shown in Fig. 13.

5 Simulation and experiment results

5.1 Simulations

In order to verify the validity of our CPG model, the application of the algorithm described in the previous sections is firstly illustrated with the help of a simulation. The mechanical model, a hexapod robot, has been introduced in the previous section, where the lengths of the three links, coxa, femur and tibia, are 26 mm, 85 mm, and 134 mm, respectively. The narrowest part of the body is 106 mm, while the widest part, which is between the two middle legs, is 136 mm. The body length is 170 mm. The prototype is built in the CAD software — Solidworks, and then it is imported and assembled in the simulation software — ADAMS. All of the parts are designed by aluminum and the actuators are modeled by a standard servomotor module. All of the 18 joints are revolute

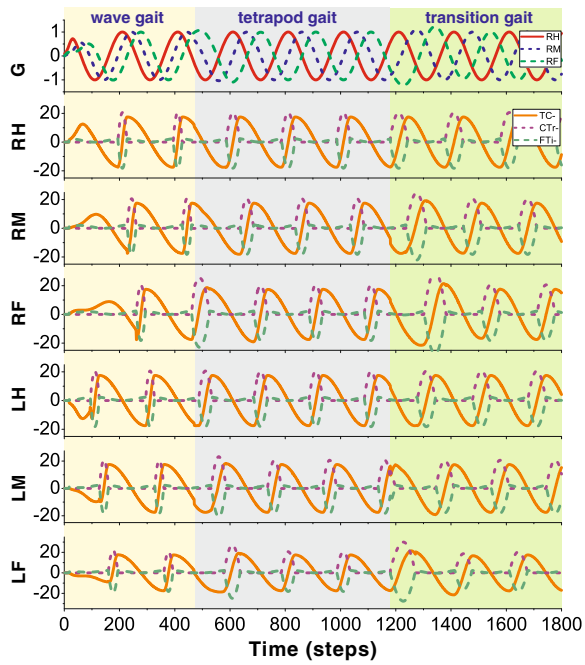
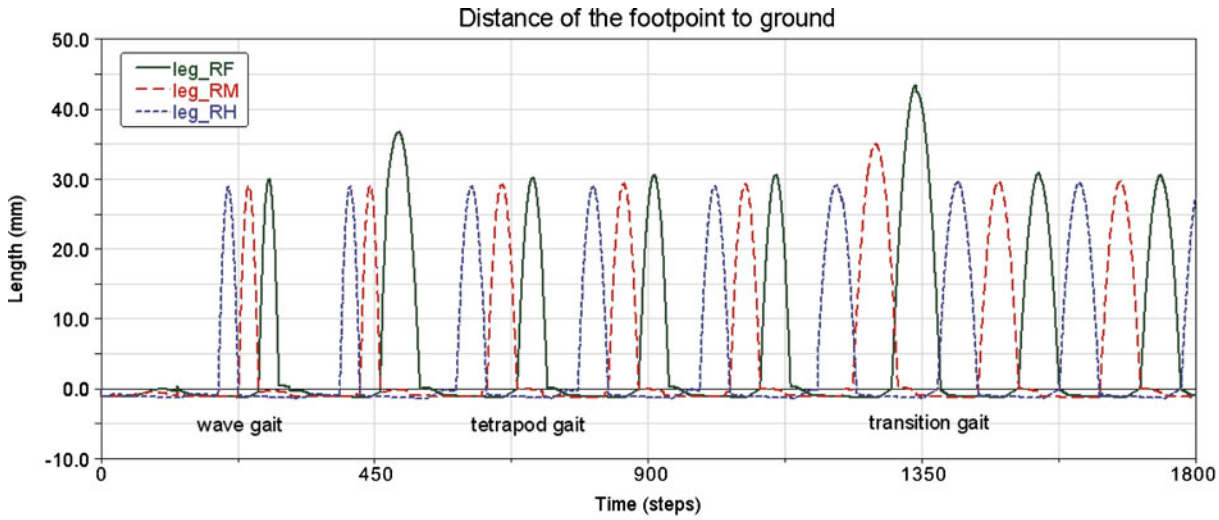
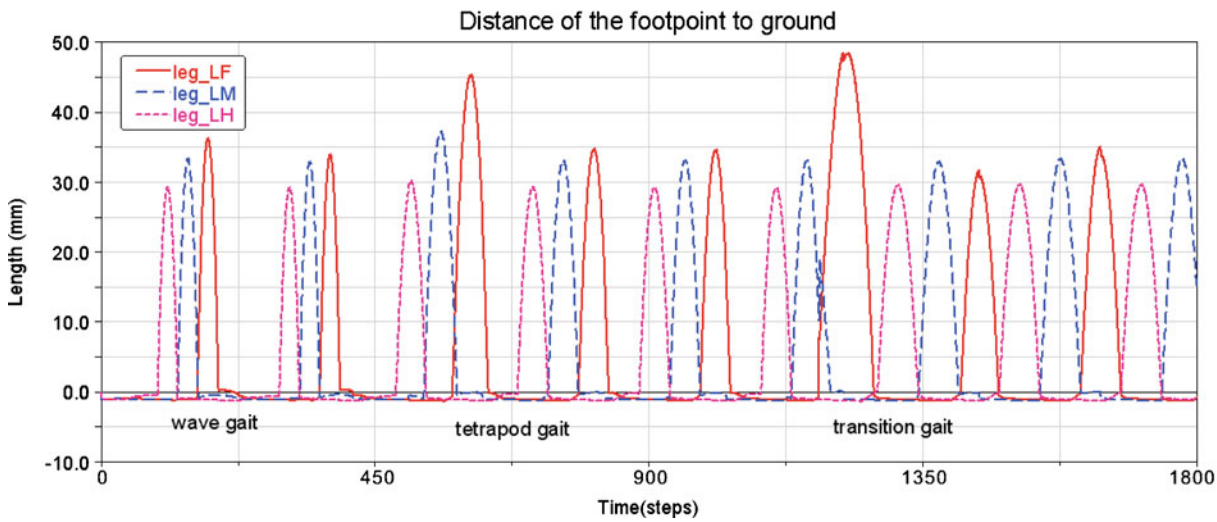


Fig. 15 The current angles values used to control the robot joint positions in the simulation and experiment. The first row of wave is the original CPG signal. From top to bottom, the waves stand for the different legs ordered by RH, RM, RF, LH, LM and LF. The *solid waves* indicate the angle value of TC- joints; the *dot lines* stand for the CTr- joints; the *dash lines* correspond to the FTi- joints



(a) The right three legs



(b) The left three legs

Fig. 16 The distance of the footpoints to the ground

pair. Every leg is set up a *contact* to the ground so that the friction can drive the body forward. The servo-motor is easy to accomplish position control. Thus, it can be easily transplanted to the physical prototype if the simulation passes.

The Matlab-ADAMS co-simulation is employed to demonstrate the correctness of the proposed algorithm. The control system structure is shown in Fig. 14. The positions, i.e. the joint angles, are calculated in Matlab and are founded as a m-function in Simulink. Motions which depend on the revolution joints are created on all of the joints. The motions use *state variable* values as the position control signal. 18 *plant inputs* are created and connected to the corresponding *state variable* to obtain values from Matlab. Moreover, we measure the distance of the body center in the ground coordinate as a *plant output* (although we don't use it as feedback in the control system). ADAMS/Control module is employed to export the mechanical model in ADAMS as a component in Simulink. Thus, the Matlab function calculated

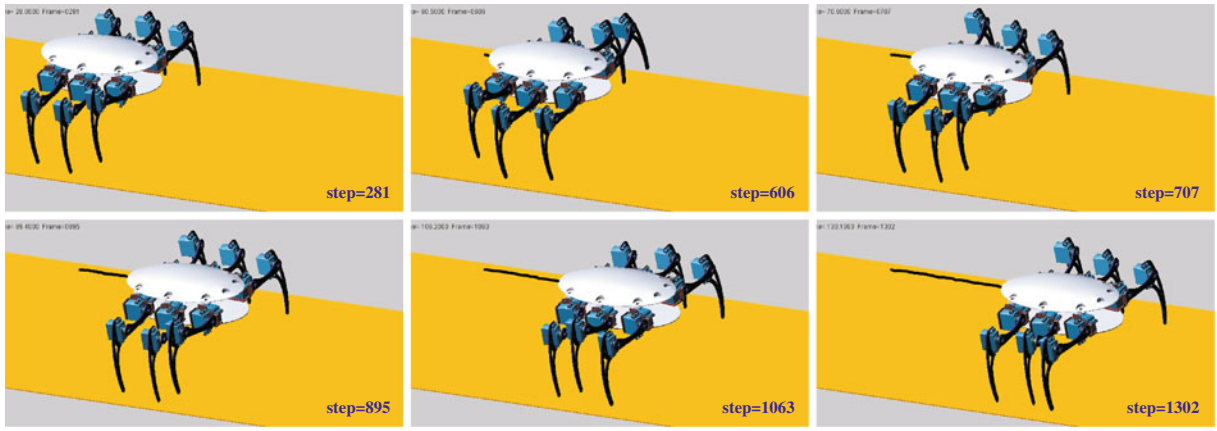


Fig. 17 The screenshot of the ADAMS dynamic simulation

the joint angles in real time and ADAMS displays the movement. We can also import the result files from Matlab to ADAMS to reproduce the simulation and watch the postprocessing results. In this way, the robot can perform the joint movement calculated in Matlab.

The three joint angles of the six legs are shown in Fig. 15, which are calculated by the algorithm stated previously. At first, the phase lag is set to 60°, and the wave gait emerges. Then, the phase lag is set to 90° at the time t = 47. The LF leg and RH leg are move simultaneously at this time, so are the LH and RF legs. At t = 118, the phase lag is set to 120° so as to generate the gait whose duty factor is $\beta = \frac{2}{3}$. With the increase of the phase lag, the fraction of the transfer phase to the whole walking cycle is enlarged.

In order to inspect the gait pattern satisfying our demand or not, we create some markers to measure the distance between the footpoint of each leg and ground. Figure 16 illustrates the simulation result. The legs are lifted up 30 mm to

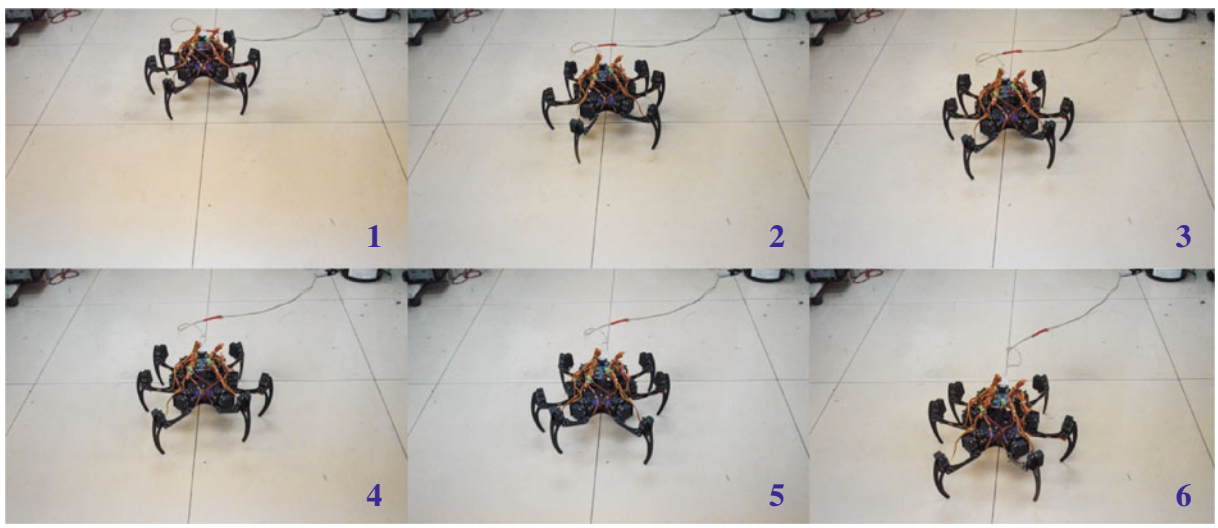


Fig. 18 The screenshot of the real prototype experiment

50 mm. At the first two walking cycles, the leg swings sequently and each leg's transfer time occupies one-sixth of the whole period. Each of the middle three walking cycles is divided into four steps. The legs swing for one step and recover for three steps. The LF and RH legs are in one group, and the RF and LH legs are in one group. The last three cycles stand for the gait that the swing phase occupied one-third of the walking period. At this gait pattern, the RH leg moves immediately after the RF leg touchdown, so are the left legs.

Figure 17 shows us the screenshots of the ADAMS dynamic simulation. The black line stands for the trajectory of the body center. It is almost a straight line with only a little undulation in the lateral direction and vertical direction. The robot moves almost at a constant velocity, since the frequency is maintained at a constant value, $\omega = \frac{\pi}{10}$. There are no great changes of the foot-points' distance to ground. The gait pattern varies fluently. The supplementary video I shows the gait transition clearly. If we adjust the phase lag slowly and don't set it to the typical value, such as 60, 90, etc., the gait transition would be smoother.

5.2 Physical prototype experiment

A physical prototype is built up to make the result more convincible. The actuators we used are standard analog servo-motor. A controlling board is equipped on the robot to convert the joint angle values from serial port to Pulse-Width Modulation (PWM) waves. The baudrate of the serial port is 115.2 kbit/s. The 18 joint angles are calculated at an upper PC (Now we use Matlab to send the instruction. Some embedded processor would be equipped to the robot in future to make it totally autonomous.) and sent to the controlling board in real time. Figure 18 shows the experimental screenshots. The supplementary video II shows the experiment record.

The result is nearly the same as the simulation except that the robot cannot walk so straight as in simulation. We deduce that it is because different legs have different friction from the ground, and then some legs will slip. Therefore, foot contact sensors are important to make the each leg's pressure equivalent in future works.

So we can draw a conclusion that our CPG algorithm can perform the smooth gait transition via adjusting only one control parameter: phase lag.

6 Discussion and comparison with other walking control technologies

This paper addressed a locomotion control algorithm for a hexapod robot inspired by the central pattern generator in the insect's neural system. Several successful walking machines, which have been introduced in Section 1 [12, 16, 19, 21–23], have proved the effectiveness of this kind of methodology via central oscillators. The common points (also advantages) of these works and our proposed algorithm include: the nonlinear oscillators show limit cycle behavior (i.e. stability), which can be used to resist perturbations; inverse kinematics do not need to be calculated therefore we don't need to take the complicated geometric parameters of legs into consideration; period orbits are produced by the ODE (Ordinary Differential Equations) to generate oscillations and so on. Buchli et al. [29] have discussed different oscillators and analyzed the similarities and differences between them in detail.

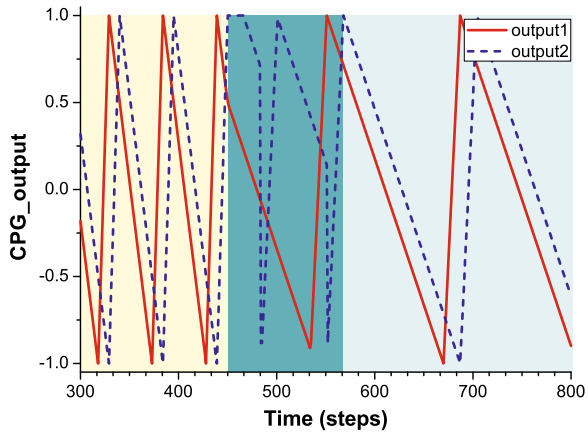
Nevertheless, our algorithm is not only a new model compared with others' work, but also more suitable for the hexapod robot's locomotion control, especially when we concern the instant of the gait varying.

As we all know, the more legs the robot owns, the more gait patterns it shows. At the same time, the more easily it becomes stable. For example, the two legs are in swing and support phase alternately in the bipedal locomotion in most cases. So, more attentions are paid on the balance maintaining when designing the humanoid robot. Although nonlinear oscillators are also applied, Uchitane et al. [33] employ some evolution strategies afterward to tune the parameters of CPG to avoid mutation. Righetti et al. [34] raise some strategies to shape the signal from the oscillator so that the infant robot performs a stable trot-like gait. They are all impressive works but the key point is not the gait transition.

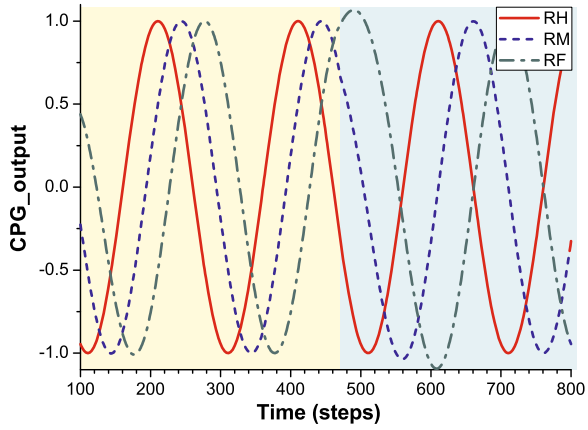
For quadruped robot, gait modes can be attributed to the following five types: walk, trot, pace, gallop and bound, which are not so many as the hexapod's locomotion. Similar to biped motion, dynamic balance issues should be taken into account. Just because of this, models are always complex enough to handle with some disturbances caused by the leg contacts to obstacles and body inclination while gait transition issue is always not so important. Fukuoka et al. [12] show a very impressive work that their quadruped robot "Tekken" can walking on irregular terrains adaptively. However, to change their gait modes, three coupled parameters (θ_{stance}, τ and ω_{21}) have to be changed interdependently and it will affect the stride and cyclic period (section 4.1 in literature [12]). Besides, the three parameters are adjusted manually, i.e., in an experimental attempt method.

For hexapod robot, we nearly don't need to consider the dynamic balance problem because the projected point of body center is always in the polygon formed by the support legs. However, the gait patterns become complex due to the increasing of legs. In the previous works, gaits are regarded as several different patterns discretely. For example, Manoonpong [23] uses several P (periods) to indicate gait patterns. Thus, the gait transition could not be smooth (see Fig. 19a. It spends more than 100 steps to vary the gait from tetrapod gait to wave gait.) While, in our algorithm, we nearly do not need time to stabilize the oscillation when changing gait (see Fig. 19b). Arena's hexapod robot [16] does not refer to the gait transition problems, either. They use templates to define gait patterns, which means the design ideas is from the already known gait to generate corresponding oscillator, rather than use a universal oscillating model to perform gaits. In general, our proposed model is better at coping with the gait smooth transition issue, compared with the previous works.

What's more, the stride, oscillating frequency and gait can be adjusted independently according to the three parameters, r, ω and phase lag τ, respectively, because the model is at first defined in polar coordinate. These kind of parameters are always coupled together in previous works.



(a) Manoonpong's work. The light yellow area indicates the tetrapod gait. The light blue area indicates the wave gait. The transient duration of the gait changing is shown in the green area, which means the unstable time.



(b) The application of proposed algorithm in this paper. The light yellow area indicates the wave gait while the light blue area indicates the tetrapod gait.

Fig. 19 The comparison of the gait transition instant of Manoonpong's method [23] and ours. The result indicates that there exists no transient process when gait changing of our method, only a little bit overstrike at about the 500 step (the *dot-dash* signal)

Besides, our model can perform some transient gait which is not so common seen but important in robot walking, such as $\tau = 75$.

7 Conclusion and future works

In this paper, a CPG model constructed by isochronous oscillator and first-order low-pass

filter is proposed, after analyzing the insects' gaits. The isochronous oscillator has a limit cycle property and it's easy to be analyzed in polar coordinate. The other two oscillators are essentially the delay of the rhythmic signals. Therefore the gaits are easily controlled by one parameter — phase lag. The frequency and amplitude of the oscillation can also be controlled independently. After transforming the CPG signal to joint space, simulation and real world experiment are given with the postprocessed angular values, in which three gaits and their smooth transition are performed. Although only three gaits are tested, the other gaits which are hard to name can be produced by simply extend the algorithm (changing τ value). The experimental results demonstrate the validity of our CPG model.

Our future work will be the omnidirectional walking via this algorithm and walking on uneven terrains. Suppose there are some bumps or traps, the legs may be stuck, and lead to the instability of the robot. Therefore, some sensory information should be employed as feedback, and some signal shaping methods, such as that addressed in [34], may be useful for adjusting the control of the legs. Another interesting topic is to analyze the gait from the viewpoint of energy consumption. Which gait is more efficient? Which gait is more suitable for flat terrain and which gait is more suitable for rough terrain? It needs further research.

References

1. Quinn, R.D., Ritzmann, R.E.: Construction of a hexapod robot with cockroach kinematics benefits both robotics and biology. Connect. Sci. **10**(3,4), 239–254 (1998)
2. Go, Y., Yin, X., Bowling, A.: Navigability of multi-legged robots. IEEE/ASME Trans. Mechatron. **11**(1), 1–8 (2006)
3. Moore, E.Z., Campbell, D., Grimminger, F., Buehler, M.: Reliable stair climbing in the simple hexapod 'RHex'. In: Proceedings of IEEE International Conference on Robotics and Automation (ICRA), pp. 2222–2227 (2002)
4. Duan, X.J., Chen, W.H., Yu, S.Q., Liu, J.M.: Tripod gaits planning and kinematics analysis of a hexapod robot. In: Proceedings of IEEE International Conference on Control and Automation (ICCA), pp. 1850–1855 (2009)
5. Ren, G.J., Chen, W.H., Chen, B., Wang, J.H.: Antenna sensor based on PSD and application mobile robot. J. Beihang Univ. (in Chinese) **36**(5), 601–605 (2010)
6. Ren, G.J., Chen, W.H., Chen, B., Wang, J.H.: Mechanism design and analysis of cockroach robot based on double four-bar linkage. J. Mech. Eng. **47**(11), 14–22 (2011)
7. Delcomyn, F.: Walking robots and the central and peripheral control of locomotion in insects. Auton. Robots **7**(3), 259–270 (1999)
8. Hooper, S.L.: Central pattern generators. Curr. Biol. **10**(5), R176–R179 (2000)
9. Rossignol, S.: Locomotion and its recovery after spinal injury. Curr. Opin. Neurobiol. **10**(6), 708–716 (2000)
10. Cohen, A.H., Holmes, P.H., Rand, R.H.: The nature of the coupling between segmental oscillators of the lamprey spinal generator for locomotion: a mathematic model. J. Math. Biol. **13**, 345–369 (1982)
11. Matsuoka, K.: Mechanisms of frequency and pattern control in the neural rhythms generators. Biol. Cybern. **56**, 345–353 (1987)
12. Fukuoka, Y., Kimura, H., Cohen, A.H.: Adaptive dynamic walking of a quadruped robot on irregular terrain based on biological concepts. Int. J. Rob. Res. **22**(3–4), 187–202 (2003)
13. Kimura, H., Fukuoka, Y., Konaga, K., Hada, Y., Takase, K.: Towards 3D adaptive dynamic walking of a quadruped robot on irregular terrain by using neural system model. In: Proceedings of IEEE/RSJ International Conference of Intelligent Robots and Systems (IROS), pp. 2312–2317 (2001)
14. Kimura, H., Fukuoka, Y., Cohen, A.H.: Adaptive dynamic walking of a quadruped robot on natural ground based on biological concepts. Int. J. Rob. Res. **26**(5), 475–490 (2007)
15. Taga, G., Yamaguchi, Y., Shimizu, H.: Self-organized control of bipedal locomotion by neural oscillators in unpredictable environment. Biol. Cybern. **65**, 147–159 (1991)
16. Arena, P., Fortuna, L., Frasca, M.: Multi-template approach to realize central pattern generators for artficial locomotion control. Int. J. Circuit Theory Appl. **30**(4), 441–458 (2002)
17. Arena, P., Fortuna, L., Frasca, M.: Attitude control in walking hexapod robots: an analogic spatio-temporal approach. Int. J. Circuit Theory Appl. **30**(4), 349–362 (2002)
18. Arena, P., Fortuna, L., Frasca, M., Sicurella, G.: An adaptive, self-organizing dynamical system for hierarchical control of bio-inspired locomotion. IEEE Trans. Syst. Man Cybern. **34**(4), 1823–1837 (2004)
19. Inagaki, S., Yuasa, H., Arai, T.: CPG model for autonomous decentralized multi-legged robot systemgeneration and transition of oscillation patterns and dynamics of oscillators. Robot. Auton. Syst. **44**, 171–179 (2003)
20. Inagaki, S., Yuasab, H., Suzuki, T., Arai, T.: Wave CPG model for autonomous decentralized multi-legged

robot: gait generation and walking speed control. Robot. Auton. Syst. **54**, 118–126 (2006)
21. Manoonpong, P., Pasemann, F., Wörgötter, F.: Sensor-driven neural control for omnidirectional locomotion and versatile reactive behaviors of walking machines. Robot. Auton. Syst. **56**, 265–288 (2008)
22. Ijspeert, A.J., Crespi, A., Ryczko, D., Cabelguen, J.M.: From swimming to walking with a salamander robot driven by a spinal cord model. Science **315**(5817), 1416–1420 (2007)
23. Steingrube, S., Timme, M., Wörgötter, F., Manoonpong, P.: Self-organized adaptation of a simple neural circuit enables complex robot behaviour. Nat. Phys. **6**, 224–230 (2010)
24. Ijspeert, A.J.: Central pattern generators for locomotion control in animals and robots: a review. Neural Netw. **21**(4), 642–653 (2008)
25. Hughes, G.M.: Locomotion: terrestrial. In: Rockstein, M. (ed.) The Physiology of Insecta, pp. 227–254. Academic Press, New York (1965)
26. Wilson, D.M.: Insect walking. Annu. Rev. Entomol. **11**, 103–121 (1966)
27. Donner, M.D.: Real-time Control of Walking. Birkhauser, Boston (1987)
28. Kar, D.C.: Design of statically stable walking robot: a review. J. Robot. Syst. **20**(11), 671–686 (2003)
29. Buchli, J., Righetti, L., Ijspeert, A.J.: Engineering entrainment and adaptation in limit cycle systems—from biological inspiration to applications in robotics. Biol. Cybern. **95**(6), 645–664 (2006)
30. Kimura, H., Fukuoka, Y.: Adaptive dynamic walking of the quadruped on irregular terrain—autonomous adaptation using neural system model. In: Proceedings of IEEE International Conference on Robotics and Automation (ICRA), pp. 436–443 (2000)
31. Ritzmann, R.E., Quinn, R.D., Fischer, M.S.: Convergent evolution and locomotion through complex terrain by insects, vertebrates and robots. Arthropod Struct Develop. **33**, 361–379 (2004)
32. Manoonpong, P., Pasemann, F., Wörgötter, F.: Reactive neural control for phototaxis and obstacle avoidance behavior of walking machines. International Journal of Mechanical Systems Science and Engineering **1**(3), 172–177 (2007)
33. Uchitane, T., Hatanaka, T., Uosaki, K.: Evolution strategies for biped locomotion learning using nonlinear oscillators. In: Proceedings of SICE Annual Conference 2010, pp. 1458–1461 (2010)
34. Righetti, L., Ijspeert, A.J.: Design methodologies for central pattern generators: an application to crawling humanoids. In: Proceedings of Robotics: Science and Systems (RSS), pp. 191–198 (2006)