Object carrying of hexapod robots with integrated mechanism of leg and arm

Hexapod robots to carry objects: theoretical and experimental verification

Hua Deng[1, 2], Guiyang Xin[1, 2, 3], Guoliang Zhong[1, 2, *], Michael Mistry[3]
1 School of Mechanical and Electrical Engineering, Central South University, Changsha 410083, Hunan, China
2 State Key Laboratory of High-Performance Complex Manufacturing, Central South University, Changsha 410083, Hunan, China
3 School of Informatics, The University of Edinburgh, Edinburgh EH8 9AB, United Kingdom
*Corresponding Author: Guoliang Zhong, E-mail: zhong001985@csu.edu.cn

Abstract: Hexapod robots are easy to realize walking in complicated environments and have highly redundant property. It is well worth taking advantage of hexapod robots' versatility, such as using legs to do manipulation or carry objects. In this paper, several methods are proposed to deal with issues of carrying objects by transforming one or two legs as arms while walking with the other legs for hexapod robots. Firstly, we suggest practical gaits for one-legged carrying and two-legged carrying respectively. Secondly, problems existing in gait planning, for instance how to estimate the mass of object and how to calculate joint motion according to desired center of gravity (COG) trajectory, are solved by dynamic analysis and kinematic method based on COG Jacobian. Finally, the effective of our methods to implement carrying objects is demonstrated by two successful experiments of carrying a bottle of water and a box.
Keywords: Hexapod robots; gait planning; COG Jacobian; carrying objects

1. Introduction

Legged robots have really been a research hotspot in recent decades because of urgent requirements from some special scenarios like rescue response at nuclear power stations. The DARPA Robotics Challenge, which is partially stimulated by Fukushima nuclear disaster, putted legged robots into spotlight [1-4]. Many kinds of robots, such as humanoids, quadrupeds [5], hexapods [6] and wheel-legged integrated robots [7, 8], tried to accomplish several tasks like going through rough terrain or picking up tools to drill holes, etc. For humanoid robots, the biggest challenge is to keep balance when they are walking either with objects in hands or without objects, which stems from their narrow supporting polygon. There is only one robot that belongs to WPI-CMU team finished all the missions during DRC Finals without falling and being "rescued" [9]. For quadruped robots, since they cannot use three or two legs to walk, they have to mount an extra arm to pick up objects and carry them to other place [10, 11]. We think hexapod robots are more practical and flexible compared with humanoids and quadrupeds when considering balance capability and versatility. In this paper, we will try to propose several methods to deal with the issue of carrying objects for hexapod robots by transforming legs as arms. There are several choices for hexapod robots to transform legs as arms to do manipulation as shown in Fig. 1.

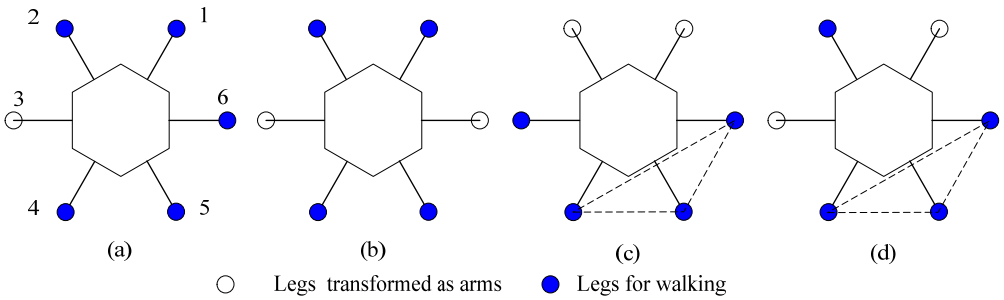

Fig. 1 Three possible choices and one impossible choice for hexapod robots to carry objects without extra manipulators

Some researchers have studied the subject of using hexapod robots to move objects. Kalouche et al. [12] still applied an extra arm to grip things on a hexapod robot. In [13], a hexapod robot developed for humanitarian demining activities is endowed with a 5-d.o.f. manipulator in front, which handles a sensor head to scan the terrain. Halvorsen [14] designed an ant inspired hexapod robot with a gripper mounted on the head like an ant with big mandibles. Lewinger et al. also created a small ant inspired hexapod robot and simply introduced the performance of their BILL-Ant-p robot in [15]. In 1995, Koyachi et al. [16, 17] proposed the concept of Integrated Limb Mechanism (ILM) which tries to integrate the functions of leg and arm into one limb in order to omit additional manipulator. Later, they developed two prototypes of "MELMANTIS-1" and "MELMANTIS-2" [18, 19]. However, the robots could only grip while stationary and cannot walk while carrying external objects because the COG would be beyond of supporting triangle when only three legs remaining as supporting legs meanwhile one is swing and the other two legs are gripping objects as shown in Fig. 1(c). Chen et al. [20, 21] tried to apply ILM to a modular and reconfigurable robot "MiniQuad" which didn't realize the purpose of carrying objects using two fore legs due to COG problem as well. Booysen et al. [22] suggested sliding the body of the robot in order to move COG into the supporting triangle before lift a rear leg. We think that is a good idea. But they didn't analyze how to move COG to desired point through the motion of joint. However, only when the COG is in the supporting triangle, the robot can walk using static gait successfully. Therefore, we should give an analytical relationship between the motion of COG and the motion of joint in order to guarantee the desired motion of COG while moving joints.

Before discussing the issue of how to calculate joint motion according COG movement, we think there is a convenient choice that is just using one leg to do manipulation as shown in Fig. 1(a). Heppner et al. [23, 24] has created a hexapod robot called as LAURON which can grip objects to a storage on the back of the robot through a gripper mounted on a leg and can realize the purpose of walking using six legs while carrying objects. In order to grip objects into the storage on the back, the leg with the gripper is designed exclusively to have a large reachable workspace. Actually, we can use the other five legs to implement walking while one leg remains leaving ground with gripping objects. That has been demonstrated by researchers who do research on fault tolerant gaits [25-27]. We also designed a gait to implement this method to carry objects. But it should be noted that the gripper could not pick up large objects limited by the size of gripper. So we think it is worth solving the problem existing in four-legged walking with two-legged grasping as mentioned above.

For the issue of obtaining joint motion according to desired COG trajectory, we can resort to

some methods proposed for humanoid robots. Humanoid robots have narrow supporting polygons. As a consequence, researchers need to plan the motions of humanoid robots very carefully. Unfortunately, most of the existing literature about locomotion of humanoid robots employs a method to treat the body motion as COG's motion approximately [28-30]. The reason for this is that the body of humanoid robot is much heavier than their legs. So this simplification method is feasible for majority of humanoid robots. Although some robots can walk based on this inaccurate method, most of walking robots do not have light legs, especially for our PH-Robot that was designed with heave parallel mechanism legs [31]. An accurate calculation method to solve this problem is to use optimization method as proposed by Kuindersma et al. [32, 33]. They firstly treat zero moment point (ZMP) dynamics as a Linear-quadratic regulator (LQR) problem, and then employ quadratic program (QP) to obtain desired joint motion constrained by whole body dynamics, actuation limits, and contact constraints. However, inspired by the concept of COG Jacobian [34, 35], we think we can get the analytical solution of this problem. In this paper, we will describe a method based on COG Jacobian that can derive the joint motion according to desired COG motion of our robot while enforcing the desired end-point motion of each leg.

On the other hand, it is worth pointing that joint motion range will limit the static stability margin for any multi-legged robots. In other words, the size of supporting polygon will affect robot's ability of carrying objects, i.e., how heavy the robot can lift and carry. The main contributions of this paper can be summarized as:

1) Complete and practical gaits for one-legged carrying and two-legged carrying for hexapod robots are proposed.
2) A novel method based on COG Jacobian is created to deal with the problem of establishing analytical solution for the relationship between joint motion and COG motion.

The remainder of this paper is organized as follows: Gait for one-legged carrying is analyzed in Section 2. Gait for two-legged grasping with four-legged walking is analyzed in Section 3. Problems of mass estimation and inverse computation of joint motion are solved theoretically in Section 4. Experiments are presented and analyzed in Section 5. Section 6 concludes the paper.

2. Carry objects using one leg

One of the critical issues for robot walking is how to keep balance. There are two types of gait can be chose for multi-legged robots, i.e., static gait or dynamic gait (trot gait). Trot gait only can be adopted by robots that have mammal shape like BigDog. However, for hexapod robots, it is difficult for them to run just as insect in nature usually climb rather than run. Static gait means robots need to keep COG in supporting polygon consistently. For one-legged grasping, we have two choices as shown in Fig. 1(a) and (b) respectively. What is interesting is that the choice of Fig. 1(b) can be understood either as one-legged grasping or two-legged grasping as shown in Fig. 2. The supporting polygon constructed by the rest four legs is a rectangle that is the type of quadruped robots. In other words, it is easy to walk as most of quadruped robots do [36, 37]. Moreover, if the swing speed of leg is fast enough, the robot can run using trot gait as well.

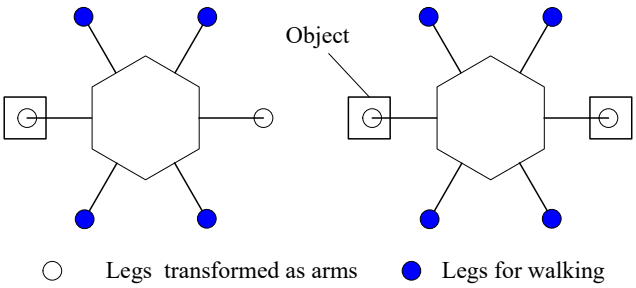

Fig. 2 Lifting two legs to become a mammal type of the other four legs

As for the first choice in Fig. 1, the stability margin is increased seemingly because of the increasing number of supporting legs. In fact, it depends on the definition of stability margin, if we define stability margin as the minimal value of a certain criterion. For example, we often use the minimal distance from COG to supporting polygon to evaluate stability. According to this criterion, the forms of Fig. 1(a) and (b) have the same stability margin as shown in Fig. 3. However, the increasing leg, i.e., leg 6 in Fig. 3, improves the stability of the right side of the robot. From this point of view, the increasing number of supporting legs indeed can improve stability of the robot.

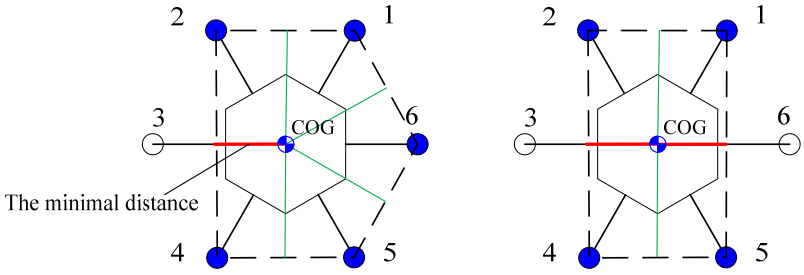

Fig. 3 Stability margin inlustration of two types of gait

Firstly, let's do gait planning for one-legged carrying method. Generally, for any kind of gait planning, the purpose is try to design a movement sequence to move every leg and robot body to desired walking direction. That doesn't mean that the increasing of number of supporting legs needs more procedure to move every leg since we can move two or three legs at one time rather than one-by-one. What we need to design is a kind of gait that can let robot go to anywhere with objects. In terms of a hexapod robot enforcing circular symmetric layout, crab gait is a good choice which doesn't need turning gait to adjust travel direction. Fig. 4-Fig. 7 are feasible crab gaits for one-legged carrying method. These gaits can bring hexapod robots to anywhere because robots can move along two axes with respect to world coordinate frame. Furthermore, it is not necessary to move body or swing legs strictly along orthogonal axes. Therefore, crab gait can reach any point in world reference frame.

Comparing these gaits, one-legged grasping with four-legged walking gaits as shown in Fig. 6 and Fig. 7 have to slide body to supporting polygon before swing leg lifts. Therefore, it needs more procedures, the numbers of which are six and eight for walking longitudinally and laterally respectively, to complete one gait cycle. By contrast, one-legged grasping with five-legged walking gaits only need four and six steps within one gait cycle as shown in Fig. 4 and Fig. 5

because these two gaits don't resort to siding COG to get static stability. However, it should be noted that the mass of object can affect the COG position after be picked up. As shown in Fig. 4, the effect of object may make the COG out of supporting polygon in step 2 and step 3. If so, this gait has to include sliding movement, too. Therefore, we should to estimation the mass of external objects and calculate the new COG after grasping objects. Then we can judge whether a sliding motion is needed. Another insure is how to move COG to the desired position, which has been motioned on Section 1. We will solve these insures in Section 4. Before that, let's give a feasible gait for two-legged grasping with four-legged walking.

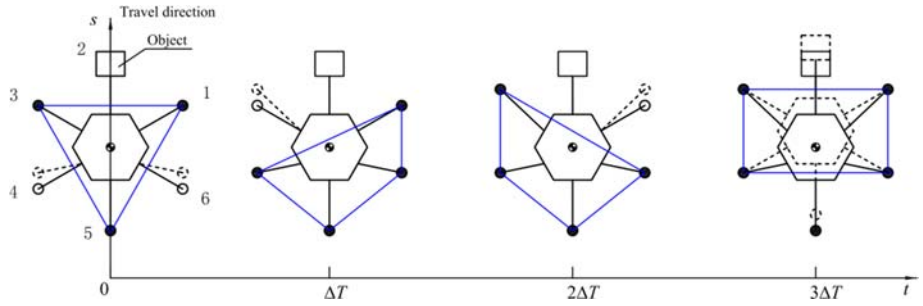

Fig. 4 Longitudinally walking gait for one-legged grasping with five-legged walking

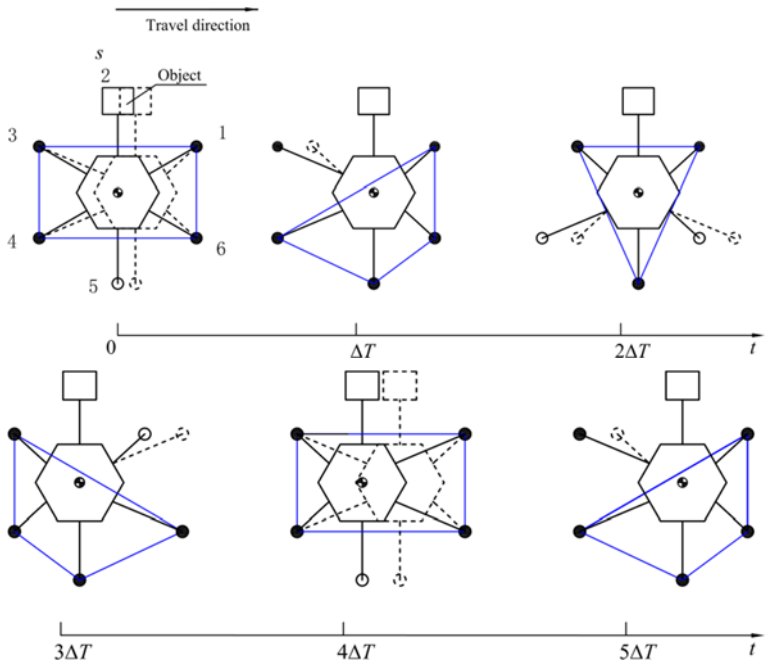

Fig. 5 Laterally walking gait for one-legged grasping with five-legged walking

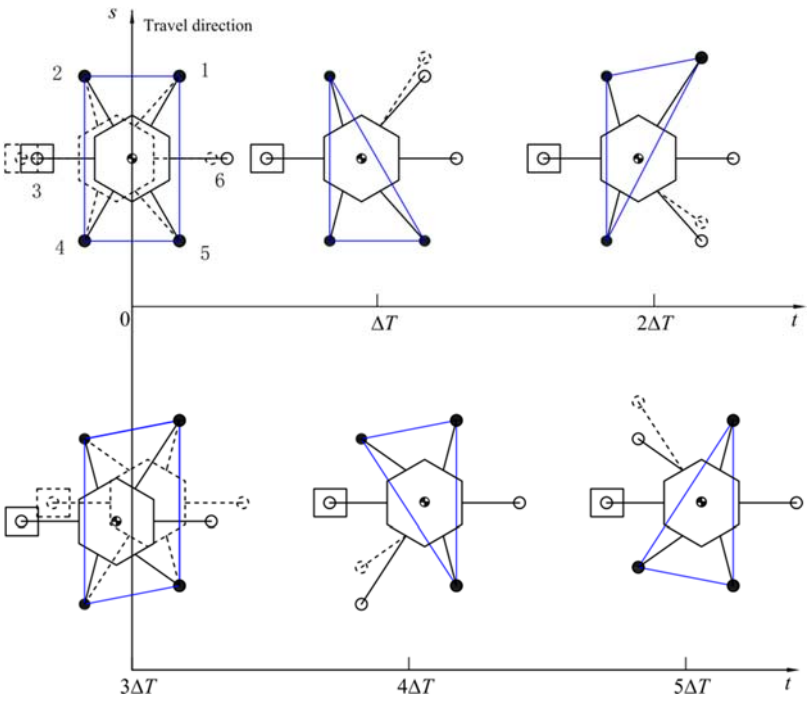

Fig. 6 Longitudinally walking gait for one-legged grasping with four-legged walking

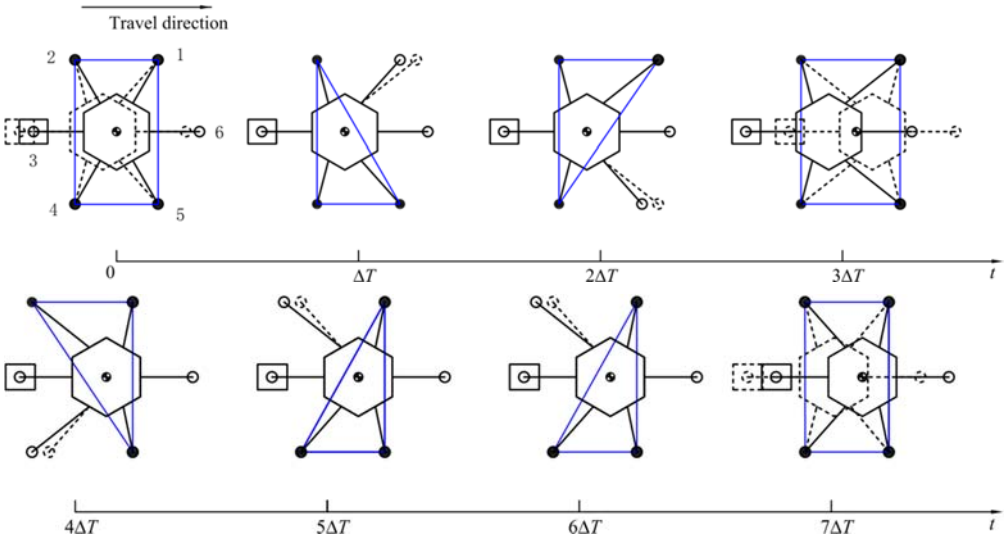

Fig. 7 Laterally walking gait for one-legged grasping with four-legged walking

3. Carrying objects using two legs

For two-legged grasping, we have three choices as shown in Fig. 1(b), (c) and (d). The type of Fig. 1(b) has been analyzed in Section 2. Whereas, the type of Fig. 1(d) is useless for most of the time. We will focus on the implementation for the type of Fig. 1(c).

Here, for description convenience, we will take our PH-Robot as an instance to analyze gait planning for two-legged grasping with four-legged walking. As shown in Fig. 8, the problem is how lift leg 3 and 6 due to the COG projection on the ground is out of the supporting triangle. So a sliding motion is necessary to shift COG into supporting triangle before lifting leg 3 and 6. A complete gait from initial state to grasping object and to walking is presented in Fig. 9. Note that

there are several points where the stability margin is small in this gait. That is due to the small workspace of our robot's leg mechanism. If the robot has serial leg mechanism, the stability margin will be improved by extend the supporting triangle. The complexity of this gait is we combine swing motion with body sliding motion together in order to reduce the number of procedure in one gait cycle. We can see that after the last procedure the robot will have the same configuration of step 7. Therefore, the gait can keep going through a cycle from step 7 to step 12 until robot comes to the target location. Of cause, this gait also suffers the problems of effect on COG by grasped object and how to switch COG into supporting polygon as mentioned in Section 2.

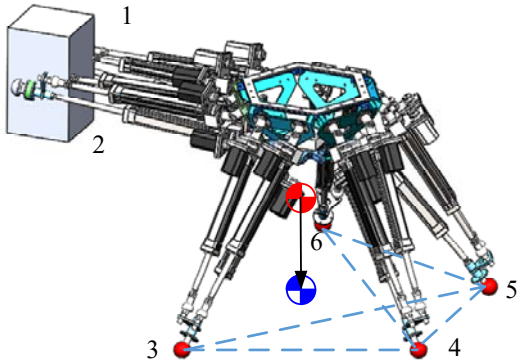

Fig. 8 Inllustration of COG projection beyond the supporting triangle while lifting leg 3 or 6

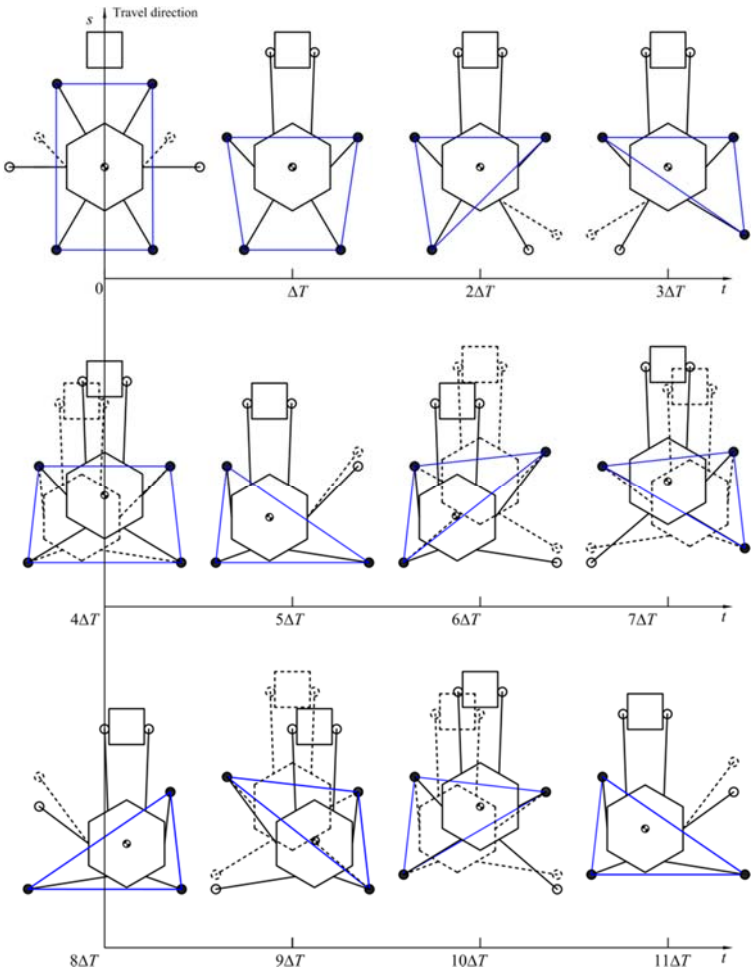

Fig. 9 Gait for two-legged grasping with four-legged walking

4. Theories

In Section 2 and 3, the movement sequence of body and legs, a.k.a. gait, is presented. In this section will mathematically solve the problems of how to decide the range of COG sliding movement and how to calculate joint motion according to pre-defined COG trajectory in order to guarantee the precise execution of COG movement.

4.1 Mass estimation of grasped objects

After robot picking up an object, that object will become a part of the robot. Strictly speaking, the object is fixed on the end-effector of leg. If we can get the mass of that object, we can calculate the new mass and COG position of the robot. Then we can do gait planning, judge whether a sliding motion is needed in some gaits and decide the range of sliding motion. There are two ways that can be utilized to get the mass of object. One is to use force/torque sensors that are mounted on angles of legs to directly measure the gravity force of grasped object. The other method is to use dynamic model to estimate the gravity force of external object. The dynamic model of one leg has been established in our previous paper [38]. In terms of legs, they can be treated as fixed-base systems when the robot body is static or moving slowly. So we can use dynamic model of leg to

estimate external forces acting on the end-effector. According to the dynamic model of one leg, the following equations can be obtained,

$$\mathbf{H}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) + \mathbf{F}(\mathbf{q},\dot{\mathbf{q}}) = \boldsymbol{\tau} + \mathbf{J}_x^T\mathbf{F}_x \tag{1}$$

where $\mathbf{F}_x = [\mathbf{f}^T \ \mathbf{M}^T]^T = [\mathbf{f}^T \ (\mathbf{p}\times\mathbf{f})^T]^T \in \mathbb{R}^{6\times1}$ is the wrench added on end-effector due to carrying object, $\mathbf{J}_x \in \mathbb{R}^{6\times3}$, $\mathbf{F}(\mathbf{q},\dot{\mathbf{q}}) \in \mathbb{R}^{3\times1}$ is the part of joint friction. We need to calculate $\mathbf{f} \in \mathbb{R}^{3\times1}$ and $\mathbf{p} \in \mathbb{R}^{3\times1}$. Note that $\mathbf{J}_x$ is not a square matrix, which means we cannot get a unique solution for $\mathbf{F}_x$. The Moore-Penrose pseudoinverse is employed as follows,

$$\mathbf{F}_x = \mathbf{J}_x^{T+}(\mathbf{H}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) + \mathbf{F}(\mathbf{q},\dot{\mathbf{q}}) - \boldsymbol{\tau}) \tag{2}$$

After getting $\mathbf{F}_x$, the position of center of mass of object can be derived by

$$\mathbf{p} = ([\mathbf{f}]_\times^T)^{-1}\mathbf{M} \tag{3}$$

where $[\bullet]_\times$ stands for the skew-symmetric matrix. For two-legged grasping, we can use the same way to calculate the acting force and moment for each leg. Then we can combine them to get the magnitude of object mass and mass position.

Note that this method is much dependent on the accuracy of dynamic model and the measurement results of motor torques. However, the dynamic model we have got in [38] is precise enough for model-based control like computed-torque control since there is PD feedback loop to compensate the accuracy of feedforward part and disturbance. But when we apply this model to measurement, the result is usually not satisfied because of the amplification of $\mathbf{J}_x^{T+}$. Parallel mechanisms have a characteristic that is a small move of end-effector needs a large move of actuated joints whereas a small torque of actuated joints can generate a large output force of end-effector. Mathematically, it due to the condition number of $\mathbf{J}_x$. Meanwhile, if the motor torques are measured by motor current, the results usually are very noisy. The practice on our PH-Robot is also not very well. As a result, we prefer to use force/torque sensors to estimate the mass of object as well as the position of center of mass.

4.2 Solving joint motion according to COG trajectory based on COG Jacobian

When the mass and its position of each link is known, the following equation can be used to calculate the COG position of the robot,

$$\mathbf{c} = \sum_{i=0}^{n}\sum_{k=1}^{n_i}\frac{m_{i,k}}{M}\mathbf{c}_{i,k} \tag{4}$$

where $\mathbf{c}$ is the COG position of the robot with respect to base frame, $\mathbf{c}_{i,k}$ means the COG position of *k*th link of *i*th limb with respect to base frame, whereas $i = 0$ means the limb is robot body, *M* stands for the total mass of the robot, $m_{i,k}$ is the mass of *k*th link of *i*th limb. However, for inverse calculation, the result of $\mathbf{c}_{i,k}$ according to $\mathbf{c}$ would be infinite. Hereafter, a method based on COG Jacobian will be formulated, which can obtain joint motion according to desired $\dot{\mathbf{c}}$ combined with $\dot{\mathbf{x}}_i$ (the desired end-effector motion of *i*th leg/limb).

Eq. (4) can be rewritten as,

$$\mathbf{c} = \frac{\sum_{i=0}^{n} m_i \mathbf{c}_i}{M} = \frac{\sum_{i=0}^{n} m_i (\mathbf{p}_0 + \mathbf{R}_0 {}^0\mathbf{c}_i)}{M} = \mathbf{p}_0 + \sum_{i=0}^{n} \mathbf{R}_0 (\sum_{k=1}^{n_i} \mu_{i,k} {}^0\mathbf{c}_{i,k}) \tag{5}$$

where ${}^0\mathbf{c}_i$ means the COG position of *i*th limb with respect to body frame, $\mathbf{p}_0$ is the position vector of origin of body frame with respect to base frame, $\mathbf{R}_0$ is the rotation matrix between body frame and base frame, m_i is the mass of *i*th limb, $\mu_{i,k} = m_{i,k} / M$. By differentiating Eq. (5), the following equation can be obtained,

$$\begin{aligned} \dot{\mathbf{c}} &= \dot{\mathbf{p}}_0 + \sum_{i=0}^{n} \dot{\mathbf{R}}_0 (\sum_{k=1}^{n_i} \mu_{i,k} {}^0\mathbf{c}_{i,k}) + \sum_{i=0}^{n} \mathbf{R}_0 (\sum_{k=1}^{n_i} \mu_{i,k} {}^0\dot{\mathbf{c}}_{i,k}) \\ &= \dot{\mathbf{p}}_0 + \sum_{i=0}^{n} (\boldsymbol{\omega}_0 \times \mathbf{R}_0)(\sum_{k=1}^{n_i} \mu_{i,k} {}^0\mathbf{c}_{i,k}) + \sum_{i=0}^{n} \mathbf{R}_0 (\sum_{k=1}^{n_i} \mu_{i,k} \frac{\partial {}^0\mathbf{c}_{i,k}}{\partial \mathbf{q}_i} \dot{\mathbf{q}}_i) \\ &= \dot{\mathbf{p}}_0 + \boldsymbol{\omega}_0 \times (\mathbf{c} - \mathbf{p}_0) + \sum_{i=0}^{n} \mathbf{R}_0 {}^0\mathbf{J}_{ci} \dot{\mathbf{q}}_i \end{aligned} \tag{6}$$

where $\boldsymbol{\omega}_0$ is the rotating velocity of robot body with respect to base frame, ${}^0\mathbf{J}_{ci} = \sum_{k=1}^{n_i} \mu_{i,k} \frac{\partial \mathbf{c}_{i,k}}{\partial \mathbf{q}_i}$ is called as the COG Jacobian for *i*th limb, $\mathbf{q}_i$ is the generalized coordinate of *i*th limb. When $i = 0$ that means the limb of body, there is no joint for this limb. Therefore, Eq. (6) can be rewritten as,

$$\dot{\mathbf{c}} = \dot{\mathbf{p}}_0 + \boldsymbol{\omega}_0 \times (\mathbf{c} - \mathbf{p}_0) + \sum_{i=1}^{n} \mathbf{J}_{ci} \dot{\mathbf{q}}_i \tag{7}$$

with the definition of $\mathbf{J}_{ci} = \mathbf{R}_0 {}^0\mathbf{J}_{ci}$.

For a legged robot, the velocity of the end-effector of *i*th leg can be obtained by

$$\dot{\mathbf{x}}_i = \boldsymbol{\Lambda}_i \dot{\mathbf{x}}_0 + \mathbf{J}_i \dot{\mathbf{q}}_i \tag{8}$$

where $\dot{\mathbf{x}}_0 = \begin{bmatrix} \dot{\mathbf{p}}_0^T & \boldsymbol{\omega}_0^T \end{bmatrix}^T \in \mathbb{R}^{6\times 1}$, $\dot{\mathbf{x}}_i = \begin{bmatrix} \dot{\mathbf{p}}_i^T & \boldsymbol{\omega}_i^T \end{bmatrix}^T \in \mathbb{R}^{6\times 1}$ $\boldsymbol{\Lambda}_i = \begin{bmatrix} \mathbf{I}_3 & -[\mathbf{R}_0 {}^0\mathbf{p}_i]_\times \\ \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix}$, $\mathbf{I}$ is an 3×3 identity matrix, $\mathbf{p}_i \in \mathbb{R}^{3\times 1}$ denotes the position vector of end-effector of *i*th limb with respect to body fame, $\mathbf{J}_i = [\mathbf{J}_{vi} \quad \mathbf{J}_{\omega i}]^T \in \mathbb{R}^{6\times 3}$ is the Jacobian matrix for *i*th leg with respect to base frame. It should be noted that the rotation and position of the end-effector for a 3-d.o.f. leg mechanism are not independent. In other words, when we assign the desired motion of end-effector position, the rotation is also determined. This relationship can be described as follows,

$$\dot{\mathbf{x}}_i = \begin{bmatrix} \dot{\mathbf{p}}_i \\ \boldsymbol{\omega}_i \end{bmatrix} = \mathbf{J}_i \dot{\mathbf{q}}_i = \begin{bmatrix} \mathbf{J}_{vi} \\ \mathbf{J}_{\omega i} \end{bmatrix} \dot{\mathbf{q}}_i \tag{9}$$

$$\boldsymbol{\omega}_i = \mathbf{J}_{\omega i} \mathbf{J}_{vi}^{-1} \dot{\mathbf{p}}_i \tag{10}$$

Eq. (8) yields,

$$\dot{\mathbf{x}}_0 = \boldsymbol{\Lambda}_i^{-1} (\dot{\mathbf{p}}_i - \mathbf{J}_i \dot{\mathbf{q}}_i) \tag{11}$$

Obviously, for any other leg *j*, there is the following equation,

$$\dot{\mathbf{x}}_0 = \Lambda_i^{-1} (\dot{\mathbf{p}}_i - \mathbf{J}_i \dot{\mathbf{q}}_i) = \Lambda_j^{-1} (\dot{\mathbf{p}}_j - \mathbf{J}_j \dot{\mathbf{q}}_j) \tag{12}$$

If we choose one supporting leg as a base limb, and for description convenience, we define the number of the base limb as $i = 1$. Then the joint motions of other limbs can be determined by the

motion of this base limb and their own end-effector motion as follows,

$$\dot{\mathbf{q}}_i = \mathbf{J}_i^+(\dot{\mathbf{p}}_i - \mathbf{\Lambda}_i\mathbf{\Lambda}_1^{-1}(\dot{\mathbf{p}}_1 - \mathbf{J}_1\dot{\mathbf{q}}_1)) \tag{13}$$

where $\dot{\mathbf{p}}_1 = 0$ because the base leg is a supporting leg. As a result, Eq. (13) can be rewritten as follows,

$$\dot{\mathbf{q}}_i = \mathbf{J}_i^+(\dot{\mathbf{p}}_i + \mathbf{\Lambda}_i\mathbf{\Lambda}_1^{-1}\mathbf{J}_1\dot{\mathbf{q}}_1) \tag{14}$$

Then substituting Eq. (14) into Eq. (7) yields the following equation,

$$\dot{\mathbf{c}} = \dot{\mathbf{p}}_0 + \boldsymbol{\omega}_0 \times (\mathbf{c} - \mathbf{p}_0) + \mathbf{J}_{c1}\dot{\mathbf{q}}_1 + \sum_{i=2}^{n}\mathbf{J}_{ci}\mathbf{J}_i^+\dot{\mathbf{p}}_i + \sum_{i=2}^{n}\mathbf{J}_{ci}\mathbf{J}_i^+\mathbf{\Lambda}_i\mathbf{\Lambda}_1^{-1}\mathbf{J}_1\dot{\mathbf{q}}_1 \tag{15}$$

According to Eq. (11), there is the following equation for the base limb,

$$\begin{bmatrix}\dot{\mathbf{p}}_0\\ \boldsymbol{\omega}_0\end{bmatrix} = \begin{bmatrix}\mathbf{I}_3 & [\mathbf{R}_0{}^0\mathbf{p}_1]_\times\\ \mathbf{0}_3 & \mathbf{I}_3\end{bmatrix}\left(\begin{bmatrix}\dot{\mathbf{p}}_1\\ \boldsymbol{\omega}_1\end{bmatrix} - \begin{bmatrix}\mathbf{J}_{v1}\\ \mathbf{J}_{\omega 1}\end{bmatrix}\dot{\mathbf{q}}_1\right) \tag{16}$$

Note that $\dot{\mathbf{x}}_1 = \mathbf{0}$ because the base limb is a supporting leg. Therefore, Eq. (16) can be rewritten as,

$$\begin{bmatrix}\dot{\mathbf{p}}_0\\ \boldsymbol{\omega}_0\end{bmatrix} = -\begin{bmatrix}\mathbf{I}_3 & [\mathbf{R}_0{}^0\mathbf{p}_1]_\times\\ \mathbf{0}_3 & \mathbf{I}_3\end{bmatrix}\begin{bmatrix}\mathbf{J}_{v1}\\ \mathbf{J}_{\omega 1}\end{bmatrix}\dot{\mathbf{q}}_1 = \begin{bmatrix}-(\mathbf{J}_{v1} + [\mathbf{R}_0{}^0\mathbf{p}_1]_\times\mathbf{J}_{\omega 1})\dot{\mathbf{q}}_1\\ -\mathbf{J}_{\omega 1}\dot{\mathbf{q}}_1\end{bmatrix} \tag{17}$$

Substituting Eq. (17) into Eq. (15) yields,

$$\dot{\mathbf{c}} = -(\mathbf{J}_{v1} + [\mathbf{R}_0{}^0\mathbf{p}_1]_\times\mathbf{J}_{\omega 1})\dot{\mathbf{q}}_1 + (\mathbf{c} - \mathbf{p}_0)\times\mathbf{J}_{\omega 1}\dot{\mathbf{q}}_1 + \mathbf{J}_{c1}\dot{\mathbf{q}}_1 + \sum_{i=2}^{n}\mathbf{J}_{ci}\mathbf{J}_i^+\dot{\mathbf{p}}_i + \sum_{i=2}^{n}\mathbf{J}_{ci}\mathbf{J}_i^+\mathbf{\Lambda}_i\mathbf{\Lambda}_1^{-1}\mathbf{J}_1\dot{\mathbf{q}}_1 \tag{18}$$

Using $\mathbf{p}_0 = \mathbf{p}_1 - \mathbf{R}_0{}^0\mathbf{p}_1$ to replace $\mathbf{p}_0$ in Eq. (18) yields,

$$\dot{\mathbf{c}} = -\mathbf{J}_{v1}\dot{\mathbf{q}}_1 + (\mathbf{c} - \mathbf{p}_1)\times\mathbf{J}_{\omega 1}\dot{\mathbf{q}}_1 + \mathbf{J}_{c1}\dot{\mathbf{q}}_1 + \sum_{i=2}^{n}\mathbf{J}_{ci}\mathbf{J}_i^+\dot{\mathbf{p}}_i + \sum_{i=2}^{n}\mathbf{J}_{ci}\mathbf{J}_i^+\mathbf{\Lambda}_i\mathbf{\Lambda}_1^{-1}\mathbf{J}_1\dot{\mathbf{q}}_1 \tag{19}$$

As a result, the relationship between $\dot{\mathbf{c}}$ and $\dot{\mathbf{q}}_1$ can be obtained from Eq. (19) as follows,

$$\dot{\mathbf{q}}_1 = (-\mathbf{J}_{v1} + (\mathbf{c} - \mathbf{p}_1)\times\mathbf{J}_{\omega 1} + \mathbf{J}_{c1} + \sum_{i=2}^{n}\mathbf{J}_{ci}\mathbf{J}_i^+\mathbf{\Lambda}_i\mathbf{\Lambda}_1^{-1}\mathbf{J}_1)^{-1}(\dot{\mathbf{c}} - \sum_{i=2}^{n}\mathbf{J}_{ci}\mathbf{J}_i^+\dot{\mathbf{p}}_i) \tag{20}$$

After getting $\dot{\mathbf{q}}_1$ from Eq. (20), we can get the joint motions of other limbs including either supporting legs or swing legs from Eq. (14). The benefit of this method is that it can guarantee desired COG trajectory as well as the desired end-effector trajectory of each leg. That is perfectly consistent with gait and trajectory planning. In the procedure of gait and trajectory planning, we plan the desired COG trajectory and the end-effector trajectories of swing legs. Then we can use Eq. (20) and (14) to get the motions of all active joints. Finally, if we use joint space control to track these joint motions, the desired COG trajectory and end-effector trajectories can be implemented. The complete control frame is shown in Fig. 10. It should be noted that in order to gain a precise trajectory tracking performance, the joint controller is also a crucial element in this control frame to guarantee walking successfully. Actually, we have proposed two kinds of good methods for joint control in our earlier works [39, 40].

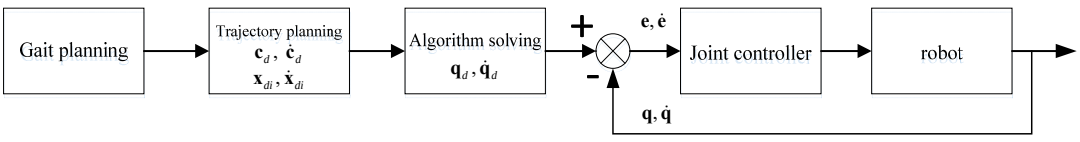

Fig. 10 Control block diagram for two-legged grasping with four-legged walking gait

5. Experimental verification

In this section, we will present two experiments for one-legged carrying and two-legged carrying respectively. For one-legged carrying, the gaits shown in Fig. 4 and Fig. 5 were applied successfully to moving a battle of water as shown in Fig. 11. In term of two-legged carrying, the gait shown in Fig. 9 was implemented successfully to move an empty box as shown in Fig. 12. According to the gaits shown in Fig. 4 and Fig. 5, there is no need for COG trajectory planning. In the first experiment, we did not do COG trajectory planning because the mass of object, a battle of water, is relatively too small. Therefore, the robot walked laterally based on the procedures shown in Fig. 5 and then walked forward based on Fig. 4.

By contrast, the gait of two-legged carrying must include COG sliding motion. Fig. 13 shows the flow chart of the second experiment. As mentioned before, the gait will cycle from step 7 to step 12 until arriving the destination point, which is explicitly described by the flow chart. Additionally, this cycle period matches the pictures from the sixth to the twelfth in Fig. 12. It can be seen that robot has the same posture in the sixth picture and twelfth picture.

The COG trajectory in every procedure of the gait is defined by the following function,

$$\begin{cases} c_x = S_x t / T - S_x \sin(2\pi t / T) / (2\pi) \\ c_y = S_y t / T - S_y \sin(2\pi t / T) / (2\pi) \qquad t \in [0,\ T] \\ c_z = H_c \end{cases} \tag{21}$$

where T is the interval time of COG motion in one procedure, S_x and S_y can define the motion distance, H_c is a constant. This function has the property of zero velocities and accelerations at the beginning and end of the motion, which is beneficial to a smooth motion and control continuity. In this experiment, we set T as 2s and S_x and S_y depend on different procedures of the gait. Fig. 14 shows the COG trajectory from the beginning to the end of the second cycle step. Fig. 15 shows the displacements of the four legs for walking during these experiments. They are derived from the method proposed in Section 4.2 with the desired COG trajectory defined by Eq. (21). We can see that at the beginning and end of each step the motion of each actuator has zeros velocities due to the feature of Eq. (21). Moreover, the maximum displacement has become close to 150mm that is the limit range of our actuator. As a result, this also limits the payload capability of our robot (in this experiment the object is a box with the mass about 2.5Kg). If we want to carry heavier objects using this robot, we need to extend actuator's range as shown in [41] or resort to the reconfiguration method proposed our previous paper [42].

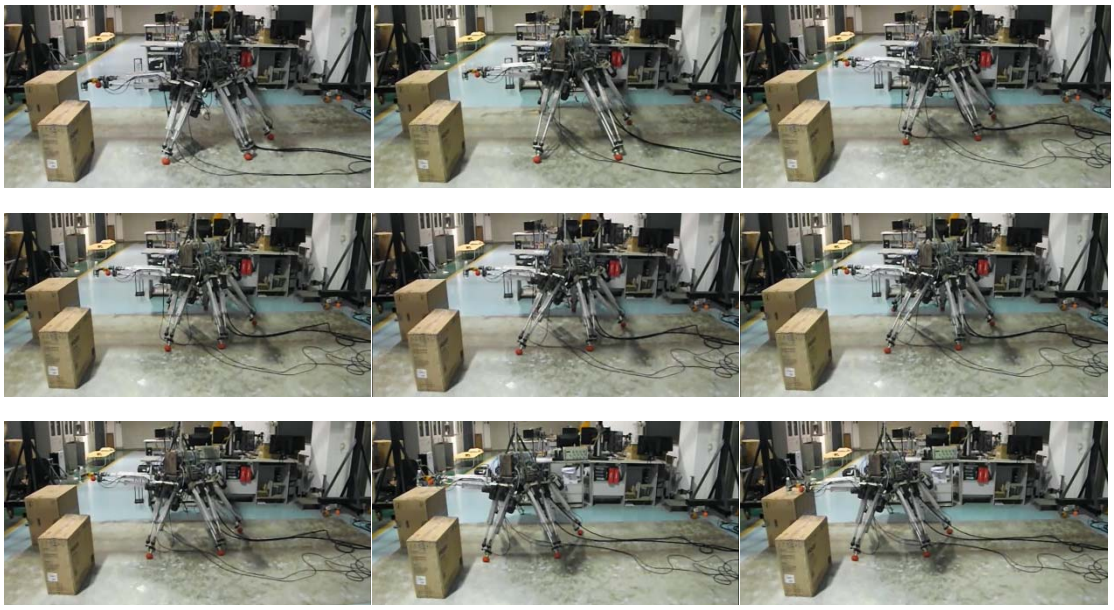

Fig. 11 Snapshoot of one-legged carrying to move a bottle of water

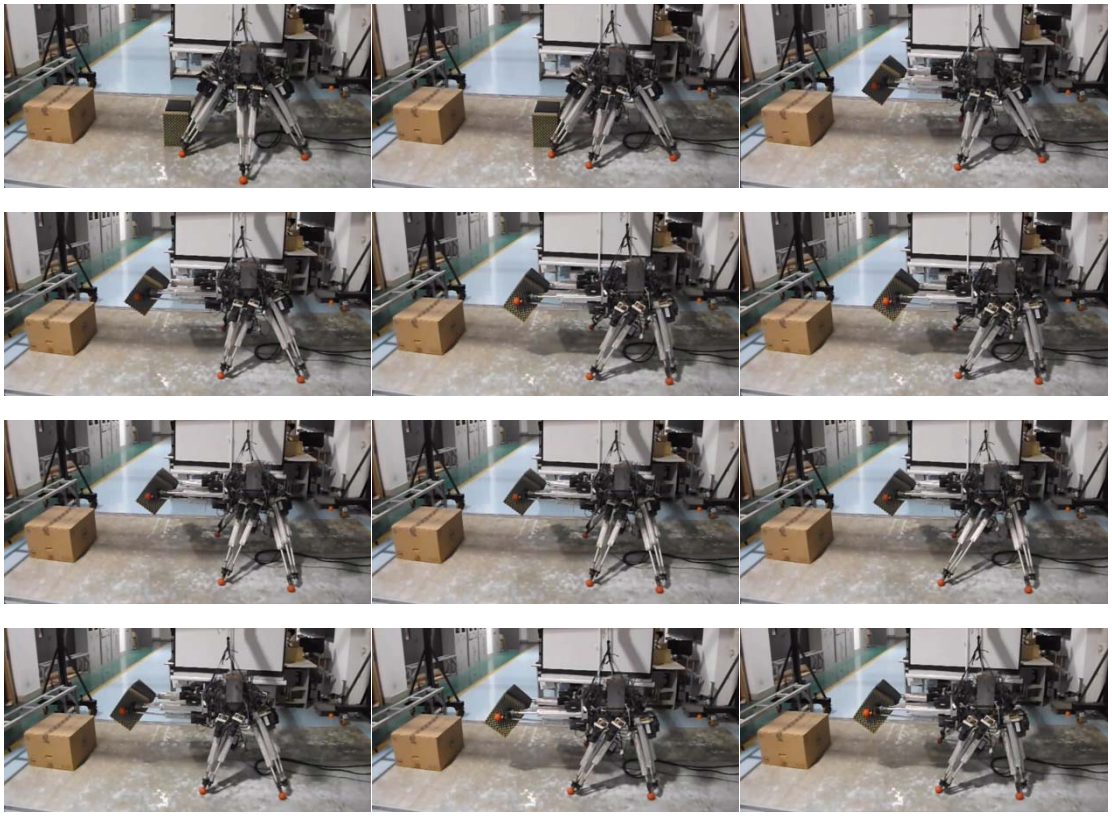

Fig. 12 Snaptshoot of two-legged carrying to move a box

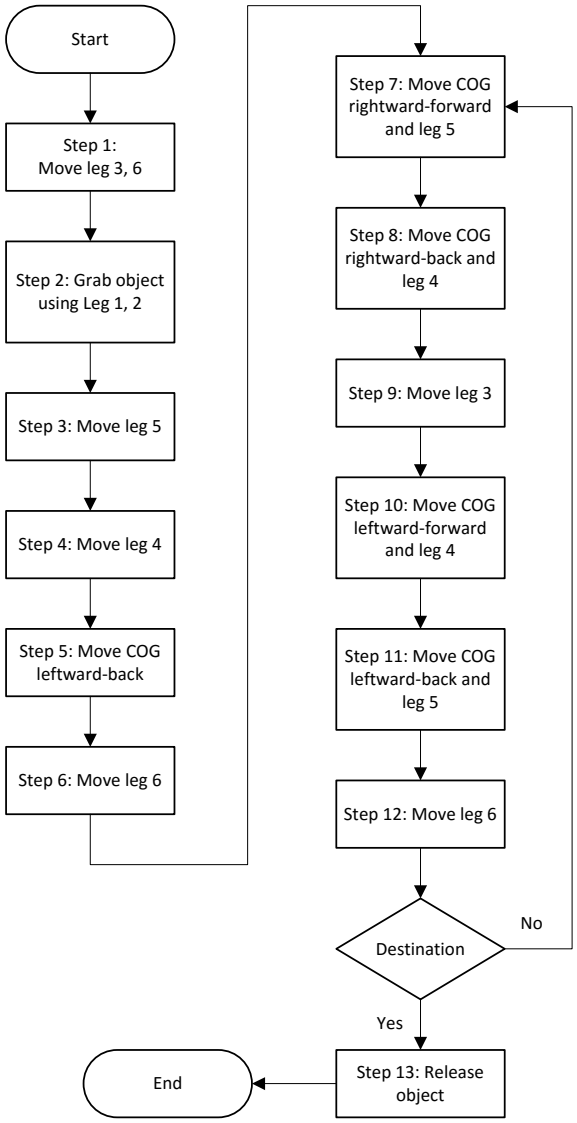

Fig. 13 Flow chart for the second experiment of two-legged carrying

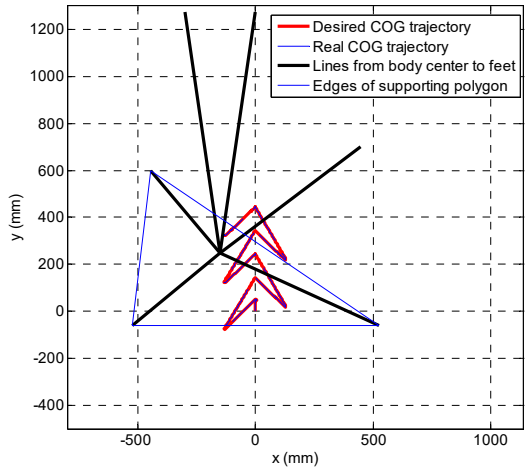

Fig. 14 COG trajectory from initial state to the end of the second cycle step (the robot is in the posture of the end step in this figure)

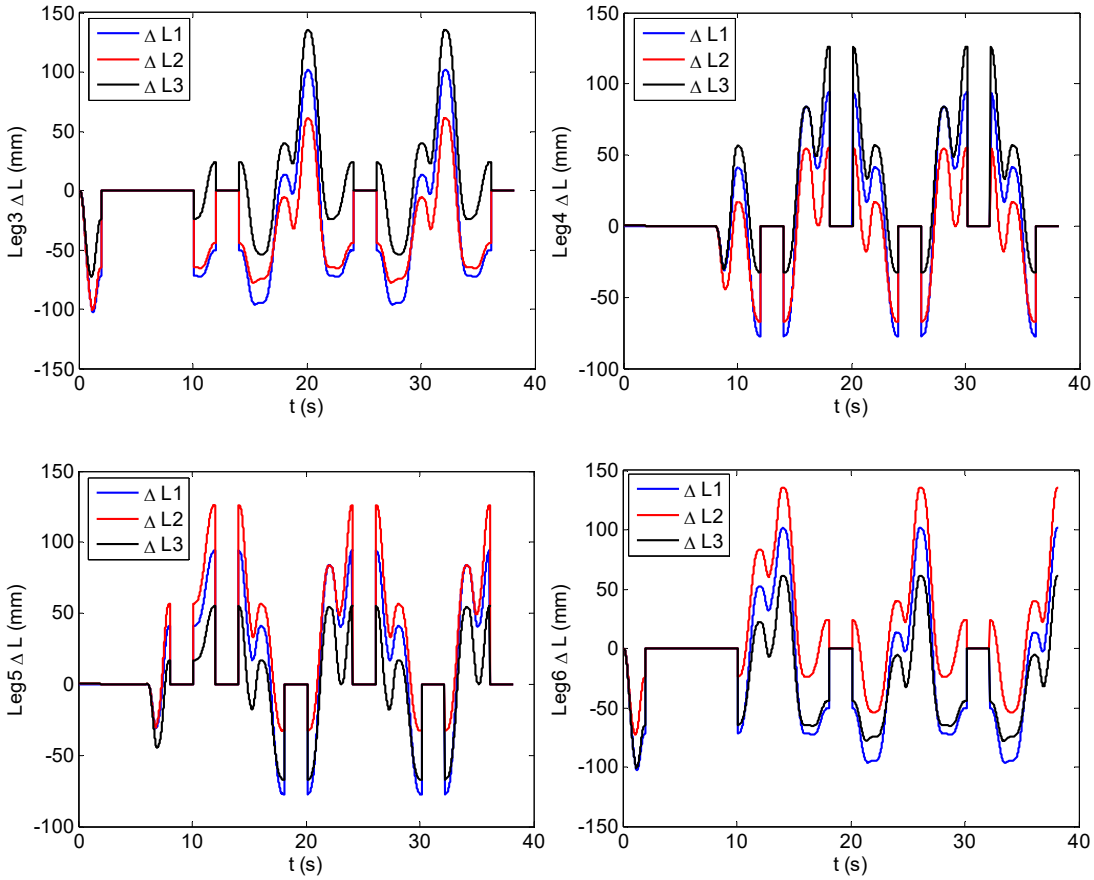

Fig. 15 Linear actuators' displacement of legs for walking in the second experiment

6. Conclusions

One-legged grasping with five-legged walking is a good choose for little object carrying because of the high stability margin the gait possesses. However, the size of gripper mounted on one leg limits the grasping capability of this gait. Therefore, two-legged grasping with four-legged walking is proposed to apply to carrying big size objects. But due to the distribution of legs around robot body we have to slide COG into supporting polygon in order to lift a certain leg. This results in a complicated gait and a problem of the calculation of joint motion based on COG trajectory. A kinematic method employing COG Jacobian is proposed to derive a precise solution of joint motion guaranteeing desired COG trajectory as well as end-effector trajectory of each leg. These gaits and computation method have been verified by experiments on our PH-Robot. If these methods are adopted by other hexapod robots that have large leg workspace, the hexapod robots will have tremendous application value in terms of carrying objects. In the future work, we will deal with the issue of fast walking when carrying objects for hexapod robots on uneven terrain.

Acknowledgment

This study was supported by the National Basic Research Program (973) of China (Grant No. 2013CB035504); the National Natural Science Foundation of China (Grant No. 51405515); the China Scholarship Council (Grant No. 201606370091).

References

[1] DRC Final Rules Book, http://archive.darpa.mil/roboticschallenge/, 2015.

[2] E. Krotkov, D. Hackett, L. Jackel, M. Perschbacher, J. Pippine, J. Strauss, G. Pratt, C. Orlowski, The DARPA Robotics Challenge Finals: Results and Perspectives, J. Field Robot. 34 (2) (2016) 229–240.

[3] M. Johnson, B. Shrewsbury, S. Bertrand, T. Wu, D. Duran, M. Floyd, P. Abeles, D. Stephen, N. Mertins, A. Lesman, Team ihmc's lessons learned from the darpa robotics challenge trials, J. Field Robot. 32 (2015) 192–208.

[4] S. Kohlbrecher, A. Romay, A. Stumpf, A. Gupta, O. Von Stryk, F. Bacim, D.A. Bowman, A. Goins, R. Balasubramanian, D.C. Conner, Human‐robot Teaming for Rescue Missions: Team ViGIR's Approach to the 2013 DARPA Robotics Challenge Trials, J. Field Robot. 32 (2015) 352–377.

[5] S. Karumanchi, K. Edelberg, I. Baldwin, J. Nash, J. Reid, C. Bergh, J. Leichty, K. Carpenter, M. Shekels, M. Gildner, Team RoboSimian: Semi‐autonomous Mobile Manipulation at the 2015 DARPA Robotics Challenge Finals, J. Field Robot. 34 (2) (2017) 305–332.

[6] Meet DRC Team Chiron, https://www.youtube.com/watch?v=7fDRTjm7k6w, 2013.

[7] S. Behnke, M. Schwarz, T. Rodehutskors, D. Droeschel, M. Schreiber, A. Topelidou-Kyniazopoulou, D. Schwarz, C. Lenz, S. Schuller, J. Razlaw. Team NimbRo Rescue at DARPA Robotics Challenge Finals, in: Proceedings of IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids), 2015, pp. 554–554.

[8] J. Lim, I. Lee, I. Shim, H. Jung, H.M. Joe, H. Bae, O. Sim, J. Oh, T. Jung, S. Shin, Robot System of DRC‐HUBO+ and Control Strategy of Team KAIST in DARPA Robotics Challenge Finals, J. Field Robot. 00 (0) (2016) 1–28.

[9] C.G. Atkeson, B. Babu, N. Banerjee, D. Berenson, C.P. Bove, X. Cui, M. DeDonato, R. Du, S. Feng, P. Franklin. No falls, no resets: Reliable humanoid behavior in the DARPA robotics challenge, in: Proceedings of IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids), 2015, pp. 623–630.

[10] Boston Dynamics, Introducing SpotMini, https://www.youtube.com/watch?v=tf7IEVTDjng, 2016.

[11] Boston Dynamics, Dynamic Robot Manipulation, https://www.youtube.com/watch?v=2jvLalY6ubc, 2013.

[12] S. Kalouche, D. Rollinson, H. Choset. Modularity for maximum mobility and manipulation: Control of a reconfigurable legged robot with series-elastic actuators, in: Proceedings of IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), 2015, pp. 1–8.

[13] P.G. de Santos, E. Garcia, R. Ponticelli, M. Armada, Minimizing Energy Consumption in Hexapod Robots, Adv. Robotics, 23 (2009) 681–704.

[14] Zenta, A-Pod part 2, https://www.youtube.com/watch?v=GDaNkff5Yyg, 2010.

[15] W.A. Lewinger, M.S. Branicky, R.D. Quinn, Insect-inspired, Actively Compliant Hexapod Capable of Object Manipulation, Springer Berlin Heidelberg, Berlin, Heidelberg, 2006, pp. 65–72.

[16] N. Koyachi, T. Arai, H. Adachi, K. Asami, Y. Itoh. Hexapod with integrated limb mechanism of leg and arm, in: Proceedings of IEEE International Conference on Robotics and Automation, 1995, pp. 1952–1957.

[17] T. Arai, N. Koyachi, H. Adachi, K. Homma. Integrated arm and leg mechanism and its kinematic analysis. in: Proceedings of IEEE International Conference on Robotics and Automation, 1995, pp. 994–999.

[18] N. Koyachi, T. Arai, H. Adachi, A. Murakami, K. Kawai. Mechanical design of hexapods with integrated limb mechanism: MELMANTIS-1 and MELMANTIS-2. in: Proceedings of 8th International Conference on Advanced Robotics, 1997, pp. 273–278.
[19] N. Koyachi, H. Adachi, M. Izumi, T. Hirose. Control of walk and manipulation by a hexapod with integrated limb mechanism: MELMANTIS-1. in: Proceedings of IEEE International Conference on Robotics and Automation,.2002, pp. 3553–3558.
[20] Y. Sun, X. Chen, T. Yan, W. Jia. Modules design of a reconfigurable multi-legged walking robot. in: Proceedings of IEEE International Conference on Robotics and Biomimetics, 2006, pp. 1444–1449.
[21] X. Chen, H. Pu, X. Wang, Y. Sun, W. Jia, Control System of a Modular and Reconfigurable Multilegged Robot, in: Proceedings of International Conference on Mechatronics and Automation,.2007, pp. 1926–1931.
[22] T. Booysen, F. Reiner. Gait adaptation of a six legged walker to enable gripping. in: Proceedings of Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference (PRASA-RobMech), 2015, pp. 195–200.
[23] G. Heppner, A. Roennau, J. Oberlander, S. Klemm, R. Dillmann. Laurope-six legged walking robot for planetary exploration participating in the spacebot cup. in: Proceedings of the International Conference on Automation and Robotics in Space (ASTRA 2015), 2015.
[24] A. Roennau, G. Heppner, M. Nowicki, R. Dillmann. LAURON V: A versatile six-legged walking robot with advanced maneuverability. in: Proceedings of IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), 2014, pp. 82–87.
[25] A. Cully, J. Clune, D. Tarapore, J. Mouret, Robots that can adapt like animals, Nature, 521 (2015) 503–507.
[26] U. Asif, Improving the navigability of a hexapod robot using a fault-tolerant adaptive gait, Int. J. Adv. Robot. Syst. 9 (2012) 34.
[27] X.K.D. Xilun, Gait analysis of a radial symmetrical hexapod robot based on parallel mechanisms, Chin. J. Mech. Eng-en. 27 (2014) 1.
[28] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, H. Hirukawa. Biped walking pattern generation by using preview control of zero-moment point. in: Proceedings of IEEE International Conference on Robotics and Automation, 2003, pp. 1620–1626.
[29] J. Liu, M. Veloso. Online ZMP sampling search for biped walking planning. in: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, 2008
[30] K. Erbatur, O. Kurt, Natural ZMP trajectories for biped robot reference generation, IEEE T. Ind. Electron. 56 (2009) 835–845.
[31] G. Xin, H. Deng, G. Zhong, H. Wang, Hierarchical kinematic modelling and optimal design of a novel hexapod robot with integrated limb mechanism, Int. J. Adv. Robot. Syst. 12 (2015) 123.
[32] S. Kuindersma, F. Permenter, R. Tedrake. An efficiently solvable quadratic program for stabilizing dynamic locomotion. in: Proceedings of IEEE International Conference on Robotics and Automation (ICRA), 2014, pp. 2589–2594.
[33] S. Kuindersma, R. Deits, M. Fallon, A. Valenzuela, H. Dai, F. Permenter, T. Koolen, P. Marion, R. Tedrake, Optimization-based locomotion planning, estimation, and control design for the atlas humanoid robot, Auton. Robot. 40 (2016) 429–455.
[34] T. Sugihara, Y. Nakamura. Whole-body cooperative balancing of humanoid robot using COG Jacobian. in: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, 2002, pp. 2575–2580.

[35] Y. Choi, D. Kim, B. You. On the walking control for humanoid robot based on the kinematic resolution of com jacobian with embedded motion. in: Proceedings of IEEE International Conference on Robotics and Automation, 2006, pp. 2655–2660.
[36] J. Buchli, M. Kalakrishnan, M. Mistry, P. Pastor, S. Schaal. Compliant quadruped locomotion over rough terrain. in: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, 2009, pp. 814–820.
[37] J. Estremera, P.G. de Santos, Generating continuous free crab gaits for quadruped robots on irregular terrain, IEEE T. Robot. 21 (2005) 1067–1076.
[38] G. Xin, H. Deng, G. Zhong, Closed-form dynamics of a 3-DOF spatial parallel manipulator by combining the Lagrangian formulation with the virtual work principle, Nonlinear Dyn. 86 (2016) 1329–1347.
[39] G. Zhong, H. Deng, G. Xin, H. Wang, Dynamic hybrid control of a hexapod walking robot: experimental verification, IEEE Trans. Ind. Electron. 63 (8) (2016) 5001–5011.
[40] G. Zhong, Z. Shao, H. Deng, J. Ren, Precise position synchronous control for multi-axis servo systems, IEEE Trans. Ind. Electron. DOI: 10.1109/TIE.2017.2652343, 2017.
[41] M. Agheli, L. Qu, S.S. Nestinger, SHeRo: Scalable hexapod robot for maintenance, repair, and operations, Robot. Comput.-Integr. Manuf. 30 (5) (2014) 478–488.
[42] G. Zhong, L. Chen, H, Deng, A performance oriented novel design of hexapod robots, IEEE-ASME T. Mech. DOI: 10.1109/TMECH.2017.2681722, 2017.