Robotica (2017) volume 35, pp. 1107–1120. © Cambridge University Press 2015
doi:10.1017/S0263574715001009

Motion planning and implementation for the self-recovery of an overturned multi-legged robot

Saijin Peng, Xilun Ding*, Fan Yang and Kun Xu

School of Mechanical Engineering and Automation, Beihang University, Beijing, P. R. China

(Accepted November 4, 2015. First published online: December 23, 2015)

SUMMARY

This paper first presents a method of motion planning and implementation for the self-recovery of an overturned six-legged robot. Previous studies aimed at the static and dynamic stabilization of robots for preventing them from overturning. However, no one can guarantee that an overturn accident will not occur during various applications of robots. Therefore, the problems involving overturning should be considered and solved during robot design and control. The design inspirations of multi-legged robots come from nature, especially insects and mammals. In addition, the self-recovery approach of an insect could also be imitated by robots. In this paper, such a self-recovery mechanism is reported. The inertial forces of the dangling legs are used to bias some legs to touch the ground, and the ground reaction forces exerted on the feet of landing legs are achieved to support and push the body to enable recovery without additional help. By employing the mechanism, a self-recovery approach named SSR (Sidewise-Self-Recovery) is presented and applied to multi-legged robots. Experiments of NOROS are performed to validate the effectiveness of the self-recovery motions. The results show that the SSR is a suitable method for multi-legged robots and that the hemisphere shell of robots can help them to perform self-recovery.

KEYWORDS: Overturn; Self-recovery; Motion planning; Sidewise-Self-Recovery; Multi-legged robot.

1. Introduction

The topic of multi-legged robots has been a popular research topic in recent years.[1–5] Multi-legged robots have the advantages of the ability to perform amazing terrain adaptations and high-walking stability. Many remarkable achievements by robotics researchers have been made in the structure design, gait analysis, and stability control of multi-legged robots. Currently, multi-legged robots are walking out of laboratories and are ready to assist humans in completing hazardous tasks.

Many robotics researchers are focusing on the stability of legged robots, with many impressive works recently reported. The first static stability criterion for a walking robot was proposed by McGhee and Frank,[6,7] the position of the CoM projection in the support polygon (formed by the standing feet) was used to describe the static stability quantitatively. The Static Stability Margin (SSM) became a criterion of the stable performance of walking robots. Next, Longitudinal Stability Margin (LSM),[8] Crab Longitudinal Stability Margin (CLSM)[9] and Energy Stability Margin (ESM)[10] were developed by robotics researchers to evaluate the stable performance of static walking. For dynamic locomotion, the Centre of Pressure Method (COP)[11] was the first dynamic stability criterion. The shortest distance from the COP to the edges of the support polygon was the definition of the Dynamic Stability Margin (DSM). In addition, Tumble Stability was proposed by Yoneda and Hirose[12] to evaluate the stability of robots performing manipulations and locomotion.

The stability criteria above are all used to ensure stable locomotion preventing a robot from overturning, but few researchers have considered the solutions to enable legged robots to perform self-recovery after being overturned. However, a variety of solutions have been reported about the

* Corresponding author. E-mail: xlding@buaa.edu.cn
E-mails: saijin.peng@buaa.edu.cn, robot.fan@me.buaa.edu.cn, xk_sea@163.com.

https://doi.org/10.1017/S0263574715001009 Published online by Cambridge University Press

self-recovery (self-righting) behaviours of some specific robots, such as wheeled and tracked robots. JL-I,[13] a tracked mobile robot, could afford capabilities for 90° /180° self-recovery by changing the relative configures of three identical modules. As the first step to provide a generic self-righting solution for any generic robots, a framework[14,15] was proposed to seek an efficient path plan for self-righting and applied to a physical robot with the proprioceptive sensors using 1, 2, and 3 degrees of freedom on unknown sloped planar surfaces, it was an intuitive approach by employing the position of CoM and potential energy to analyse the problem. However, the framework was built under quasi-static assumptions and it will be very complicate for multi-legged robots with at least 18 joints. Three tipover algorithms[16] for mobile robot have been validated in the real world by a robot platform based on the iRobot PackBot. RHex[5] relies on its flexible leg morphology and excellent dynamic performance to achieve the self-recovery ability, and a controller have been designed and tested to achieve the self-recovery behaviours.[17] It can be seen that the adjustable conformation or/and predominant maneuverability of robots mainly determine the ability to perform self-recovery.

Exceptional behaviours of multi-legged robots in an unstructured environment or in the field could occur unexpectedly. Most of such exceptional behaviours, such as broken legs, power loss, falling into a pit, and so on, have been given much attention.[18] Some specific corresponding emergency treatments have been studied and effectively utilized to address the exceptional situations of robots. However, few researchers considered the situation of the overturned robot. In general, robots are ideally designed and controlled to keep from overturning. However, the occurrence of overturning cannot be completely prevented, just as an insect or mammal can slip when walking or running, even when being cautious. If a robot could not recover from the overturned situation, the task to be performed by the robot must be terminated. The possible situations causing a robot to be overturned are summarized and listed here: (a) the robot is placed into an overturned status accidentally; (b) the robot is overturned by strong external disturbances, such as strong wind or impact; (c) the slipping of a foot that leads to an instability that results in overturning; (d) the stability margin of the robot is too small in the unstructured environment; and (e) the broken terrain causes the robot to overturn, and so on.

To address the issue of overturned robots, we focus on providing strategies and solutions to enable robots to recover by themselves. The concept of the self-recovery of a robot after being overturned is first proposed. A self-recovering motion named SSR is presented and applied to multi-legged robots. Here, "Sidewise" indicates that the self-recovery direction is from one side of the body to the other side of the body. The mechanism of the self-recovery motion is based on the motion of insects via observations. Eventually, our goal is to provide the robots the ability to self-recover and to continue functioning when encountering the above-described exceptional situations. According to the structure of a six-wheel-legged robot, the detailed motion planning and implementation of the self-recovery of an overturned robot is proposed in the paper. This theoretical method could be applied to the planning of the self-recovery motion of most multi-leg robots, such as quadruped, hexapod, and octopod robots.

The rest of the paper is organized as follows. The self-recovery of an overturned insect is described in Section 2. Section 3 is a brief introduction to a six-wheel-legged robot named NOROS, in which the structure and geometric parameters of the robot are presented. The motion planning and motion implementation of self-recovery are described in Section 4 and Section 5, respectively. In Section 6, the experimental results are described, and a discussion of the results is provided in Section 7.

2. Self-Recovery of an Overturned Insect

Most inspiration regarding robot design and locomotion has been provided by nature. Multi-legged robots achieve superior stability by adopting the leg mechanical structure, gaits, and motions of insects. Animals face more complicated living terrains than robots, which almost always remain in the structured environment. The situation of being overturned arises frequently when animals live on an uneven terrain and encounter many unknown external forces. The self-recovery motions of specific animals such as beetle and turtle species have been studied. Twenty modes of stereotyped righting motions were found and a basic mode was concluded in beetles.[19] The relationship between self-recovery attempts and shell geometry in turtle species also attracted researchers' attention, and a suitable shell could facilitate the self-recovery.[20,21] Thus, animals are self-recovery experts and have methods required to address the situation of being overturned, some of the methods may be implemented by multi-legged robots.

https://doi.org/10.1017/S0263574715001009 Published online by Cambridge University Press

Fig. 1. An overturned insect (Eucryp torrhynchus brandti) performing self-recovery motions.[22]

Three crucial points for the self-recovery of insects are described here, based on careful observations of insects recovering from being overturned. (1) The back of the body contacts the ground after being overturned because the body has a larger proportion than the legs. (2) The insect swings some legs to cause the body oblique to the ground, with the goal that some other legs could touch the ground, otherwise, with all legs in the air, there are no opportunities to obtain external power to enable the insect to recover by itself. (3) All the legs are divided into two groups: the ground group and the aerial group. The ground group could touch the ground and support and push the body upright, while the aerial group is in the air and swings to generate inertial forces to promote the body for recovering, as shown in Fig. 1.

Therefore, the self-recovery mechanism characterized by the above-described three points is denoted as SSR, which is short for SSR. The SSR mechanism is the basic rule involved in all types of self-recovery of insects. Humans are much more agile than other animals because the proportion of the body and legs/hands of a human is more harmonious, which enables a human to easily complete the self-recovery movements. SSR is also a means for a human to self-recover by using the hands and feet to swing or to support and push the body.

The similarities regarding the body structure and the motions enable multi-legged robots to employ the SSR mechanism.

3. Brief Introduction to the Six-Wheel-Legged Robot

The robot involved in this paper, known as NOROS, is a six-wheel-legged robot that is capable of switching between legged motion and wheeled motion.[23] NOROS combines the good terrain adaptation of legs with the high efficiency of wheels for overcoming both the low walking efficiency of legs and the terrain limitation of wheels.[24–26] The compact structure of the robot is shown in Fig. 2(a).

The NOROS body is cylinder shaped, and a transparent plastic hemisphere shell is assembled on top of the robot body for isolating the devices from the outside harsh environment and protecting them from damage. Six wheel-legs are uniformly distributed around the robot body. Each wheel-leg has three joints: hip joint, knee joint and ankle joint. A driving wheel in the shank is designed for wheeled motion, as shown in Fig. 2(b). Gait generation, leg and wheel motion transformation, and intelligent control of serial NOROSs were studied, and some significant results were obtained.

The specific structural parameters of NOROS are presented in Table I. To transform conveniently and easily between wheeled motion and legged motion, the shank part is designed as a polygonal line with two segments: the proximal shank and the distal shank. The driving wheel is placed in the junction of the proximal shank and the distal shank. The details of the parameters of the shank are also presented in Table I.

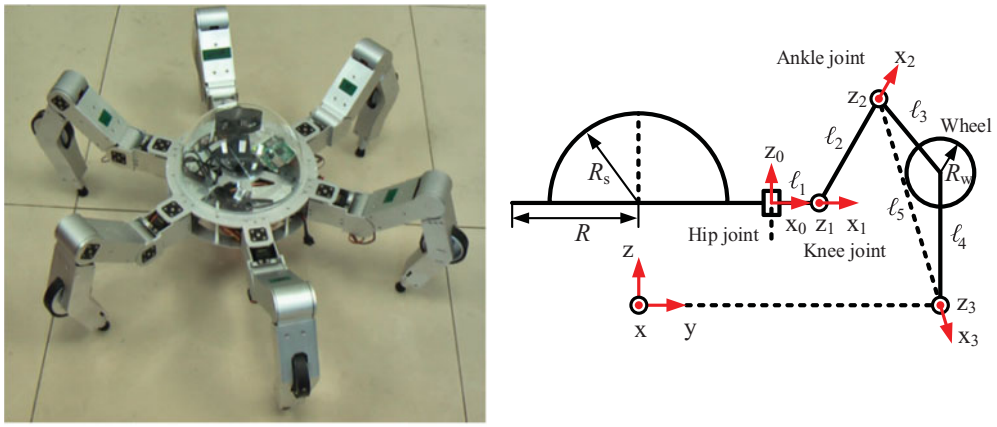
https://doi.org/10.1017/S0263574715001009 Published online by Cambridge University Press

Table I. Structural parameters of NOROS.

Body radius/R	Hemisphere shell radius/R_s	Hip/ℓ_1	Thigh/ℓ_2	Shank /ℓ_5
125.8 mm	89.4 mm	40.2 mm	120 mm	215.1 mm
Proximal shank/ℓ_3	Distal shank/ℓ_4	Shank angle/γ	Wheel radius/R_w	
96.2 mm	132.3 mm	140 degree	34 mm	

Table II. The ranges of three joints.

Hip joint	Knee joint	Ankle joint
[−90, 90]	[−45, 135]	[−120, 60]



(a) (b)

Fig. 2. The six-wheel-legged robot prototype NOROS.

The servo motors of TowerPro MG995 are used to drive the joints. According to the joint structure and limitations of servo motors, the ranges of the three joints are determined, as presented in Table II.

4. Motion Planning of Self-Recovery of the Robot

Multi-legged robots are constructed by referring to both the template of insects and engineering design principles. The similarity between insects and multi-legged robots is that their legs are distributed around the body, which results in functional similarities. Therefore, the self-recovery mechanism is also a feasible way for multi-legged robots to perform self-recovery. In addition, multi-legged robots are designed with the self-recovering ability by nature. Here, the NOROS robot is taken as an example to illustrate the motion planning of multi-legged robots.

According to the self-recovery mechanism and the SSR method, the self-recovery motions of NOROS are planned with four stages, as shown in Fig. 3.

Stage 1: The hemispherical shell touching the ground

The posture of the robot is not always the same every time the robot is overturned. The first step is to control the leg joints to adjust the robot to the desired posture (Fig. 3(a)), which is called the "starting posture". The specific strategy for NOROS is to unbend each of its six legs and distribute the legs in a circularly symmetric manner around the robot body (Fig. 4). The ideal starting posture of NOROS involves the hemisphere shell touching the ground by a contact point and the six legs being in the air.

Each leg of NOROS is marked clockwise by L_i (i = 1, 2, 3, 4, 5, and 6), as shown in Fig. 4. A virtual model of NOROS is built and simulated in MATLAB 2010b (Fig. 4(a)), which corresponds

https://doi.org/10.1017/S0263574715001009 Published online by Cambridge University Press

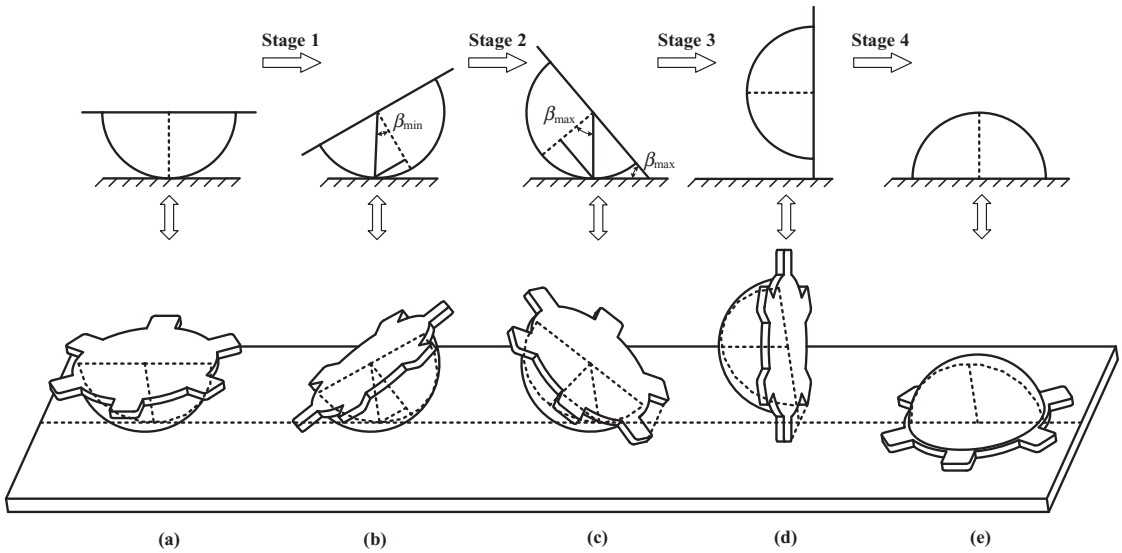


Fig. 3. Motion plan of the SSR method depicted with a series of typical body postures. These body postures are showed with 2D and 3D diagrams. Each 2D diagram in the top row is corresponding to the 3D diagram which is in the bottom row and directly below the 2D diagram. The 2D diagrams are the front view of the plane where the dotted lines lie in the 3D diagrams. Four stages are planned to perform the self-recovery motions of an overturned robot.

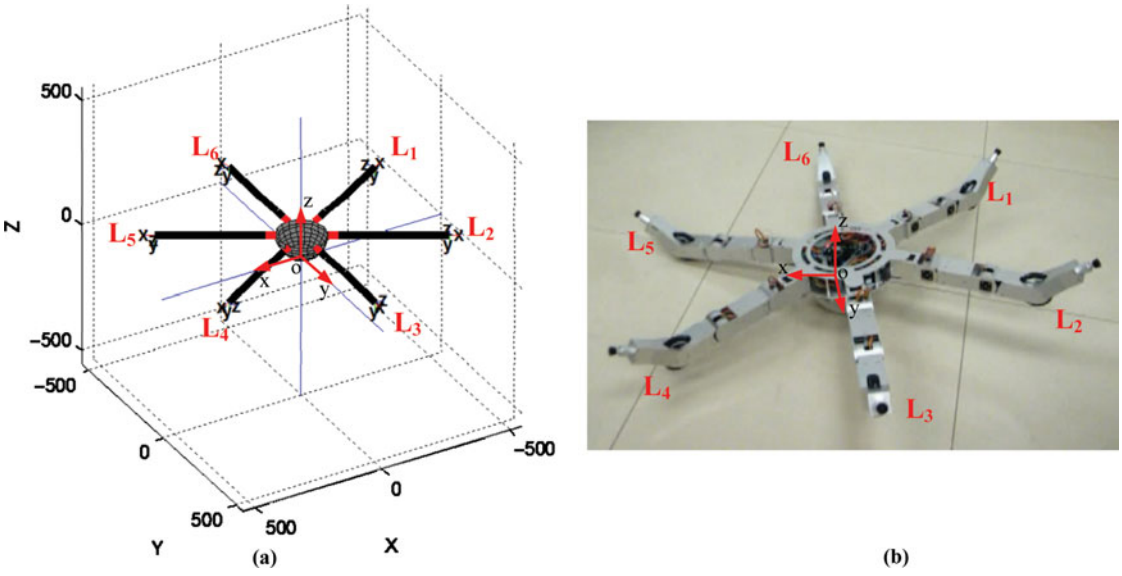


Fig. 4. Starting posture of the self-recovery process.

with the posture of Fig. 3(a). The prototype of NOROS shown in Fig. 4(b) corresponds with Fig. 3(b). Here, the recovering direction is along -x axis.

Stage 2: Rolling along the hemispherical shell

Two legs L_is (i = 1 and 2) are planned to lie in a straight line on the ground to establish the recovering axis in Stage 3 and Stage 4. The other legs L_is (i = 3, 4, 5, 6) support and push the body upright, their motions in this stage are to reach the ground and push the body to roll along the hemisphere shell.

L_is (i = 1, 2) orient themselves parallel to the y axis by rotating the hip joints, while L_is (i = 3, 4, 5, 6) orient themselves parallel to the x axis. Therefore, the position of the CoM of NOROS is moved to the +x direction with four legs in the +x axis region and two legs in the −x axis region,

https://doi.org/10.1017/S0263574715001009 Published online by Cambridge University Press

the posture of NOROS is similar to Fig. 3(b). The results of these motion sequences lead to L_is (i = 3, 4, 5, 6) touching the ground and L_is (i = 1, 2) in the air.

By planning a trajectory along the x axis for the feet of the legs L_is (i = 3, 4, 5, 6), the robot retracts these legs to support and push the body to roll along the shell until the legs L_is (i = 1, 2) reach the ground.

Stage 3: Self-recovery motion with L_is (i = 1, 2) as a rotation shaft

In this stage, the body cannot roll any further due to the encountering of a fixed rotation shaft (Fig. 3(c)), which is the intersecting line of L_1 and L_2 with the ground. In this case, the robot can't continue retracting L_is (i = 4, 5) to support and push the body because the two legs are not long enough to remain in contact with the ground, while L_is (i = 3,6) can continue to retract.

The legs L_is (i = 4, 5) lose the ability to provide propulsion motion with their feet leaving the ground, however, they can move their masses towards the recovering direction as much as possible by rotating the knee joints and the ankle joints. The bias of the CoM of the robot towards the recovery direction relieves the burdens of L_is (i = 3,6). The actual recovering propulsion in this stage reduces from four legs to two legs.

A specific trajectory of L_is (i = 3,6) instead of a straight line parallel to the x axis on the ground should be planned for the feet. The trajectory should satisfy two constraints: (1) on the ground; (2) in the reachable space of the leg.

The purpose of this stage is to push and recover the robot to the posture shown in Fig. 3(d), where the robot body is upright.

Stage 4: Self-recovery motion under the gravity

When passing through the posture Fig. 3(d), the robot body will fall naturally to the ground (Fig. 3(e)) under the influence of both the inertial force and the force of gravity. L_is (i = 4, 5) are scheduled as the first parts to reach the ground, instead of the robot body, to protect the instruments in the robot body against damage during the fall.

Up to Stage 4, the motion plan of the self-recovery process is completed. During the self-recovery motion, the hemispherical shell of the robot is in contact with the ground in Stage 1 and Stage 2. An instantaneous rotational axis is established for the rolling motion between the robot body and the ground. The position and direction of the instantaneous axis is continuously varying. The position is determined by the contact point, while the direction is along the −y direction.

Cg indicates the contact point, as shown in Fig. 5. The coordinates of Cg have the following form:

$$\begin{cases} x_{Cg} = -R_s\beta \\ y_{Cg} = 0 \\ z_{Cg} = 0 \end{cases} \tag{1}$$

where $z_{Cg} = 0$ indicates that Cg is always touching the ground. $y_{Cg} = 0$ indicates that Cg is along the x axis. The direction of the instantaneous axis is along the -y direction when the configuration of the robot body and legs remain symmetric about xz plane.

Here, we assume the poses and the motions of L_is (i = 2, 3, 4) are symmetric with L_is (i = 1,6,5) about the xz plane. However, the actual situations are more complex because the legs of a robot cannot move consistently in the unstructured environment. Nevertheless, the situation presented here is basic and representative. The approach that the robot uses to implement the instantaneous axis must follow the rule closely as possible for better control of the self-recovery process.

$$\begin{cases} \ell_6 = R_s\cos(\beta) \\ \ell_7 = R_s\sin(\beta) \end{cases} \tag{2}$$

where $\beta_{\min} \le \beta \le \beta_{\max}$, $\beta_{\min} = -\arcsin(\frac{R_s}{\ell_1+\ell_2+\ell_5})$, $\beta_{\max} = \arcsin(\frac{R_s}{R})$. The situations of $\beta_{\min}$ and $\beta_{\max}$ are shown in Fig. 3(b) and 3(c), respectively. The other parameters are listed in Table I.

5. Implementation of the Self-Recovery Process of the Robot

Stage 1: The hemispherical shell touching the ground

https://doi.org/10.1017/S0263574715001009 Published online by Cambridge University Press

Table III. The details of the motions implemented in Stage 2.

Symmetric Legs	Motions (the simplest way)
L_1 and L_2	The leg joints remain stationary
L_3 and L_6	The feet of the two legs move forward respectively along two straight lines which are both parallel to the x axis and on the ground
L_4 and L_5	The feet of the two legs move forward respectively along another two straight lines which are both parallel to the x axis and on the ground

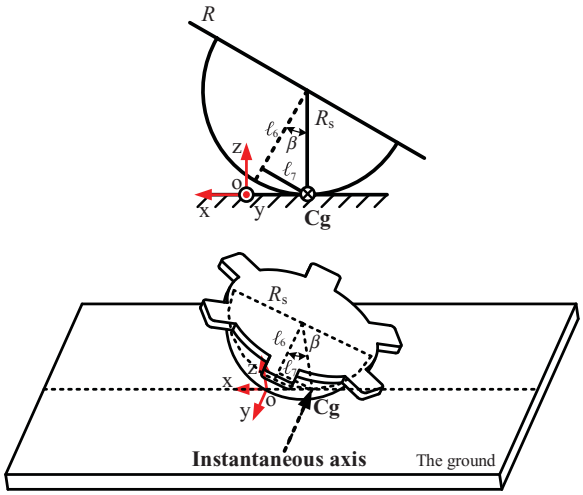


Fig. 5. The diagrams of the instantaneous axis in Stage 1 and Stage 2. The 2D diagram in the top demonstrates the instantaneous axis and the contact point Cg in the xz plane. In this view, the instantaneous axis reduces into a "cross" symbol. The 3D visualization in the bottom shows a perspective of the whole robot body and the ground.

The objective of this stage is to obtain the "starting posture". The specific implementation method is to unbend six legs and distribute them symmetrically around the robot body. The knee joint and the ankle joint are maintained at zero position to unbend a leg. The hip joint is also kept at zero position to make the thigh and shank lie in the radial direction of robot body. After the other five legs perform the same actions, the robot will be in the "starting posture". The duration of Stage 1 is $[0, t_1]$.

Stage 2: Rolling along the hemispherical shell

In this stage, the most important characteristic is that the robot performs the self-recovery motion along the hemispherical shell, which is similar to a ball rolling on the ground. The ground reaction forces exerted at the feet of L_is(i = 3, 4, 5, 6) support and push the rolling body.

Here, for consistent motions, the legs move symmetrically about the xz plane. The details of the motions implemented are presented in Table III.

The symmetric motions of the legs confirm that the contact point between the hemispherical shell and the ground lies in the xz plane and in the x axis exactly. As presented in Table III, the leg joints L_is (i = 1, 2) remain stationary, and no additional motions are required in Stage 2. However, the other four legs, L_is (i = 3, 4, 5, 6), should keep their feet in contact with the ground and moving forward respectively along four straight lines which are all parallel to the x axis. The implementation of these motions should be analysed. For such an analysis, L_3 is selected as an example, noting that L_is (i = 4, 5, 6) are similar to L_3.

Here, a four DOFs equivalent manipulator is utilized to build the model of L_3 and the body. The movement between the hemispherical shell and the ground is equivalent to a rotation motion whose rotation axis, the instantaneous rotation axis, is moving forward.

The sketch diagram of L_3 is shown in Fig. 6, and the DH parameters[27] of the equivalent manipulator are described in Table IV.

https://doi.org/10.1017/S0263574715001009 Published online by Cambridge University Press

Table IV. The DH parameters of an equivalent manipulator of L_3 in Stage 2.

Joints	θ_j	d_j	α_j	a_j
1	θ_1	$-R$	$\pi/2$	ℓ_7
2	θ_2	$-\ell_6$	$\pi/2$	ℓ_1
3	θ_3	0	0	ℓ_2
4	θ_4	0	0	ℓ_5

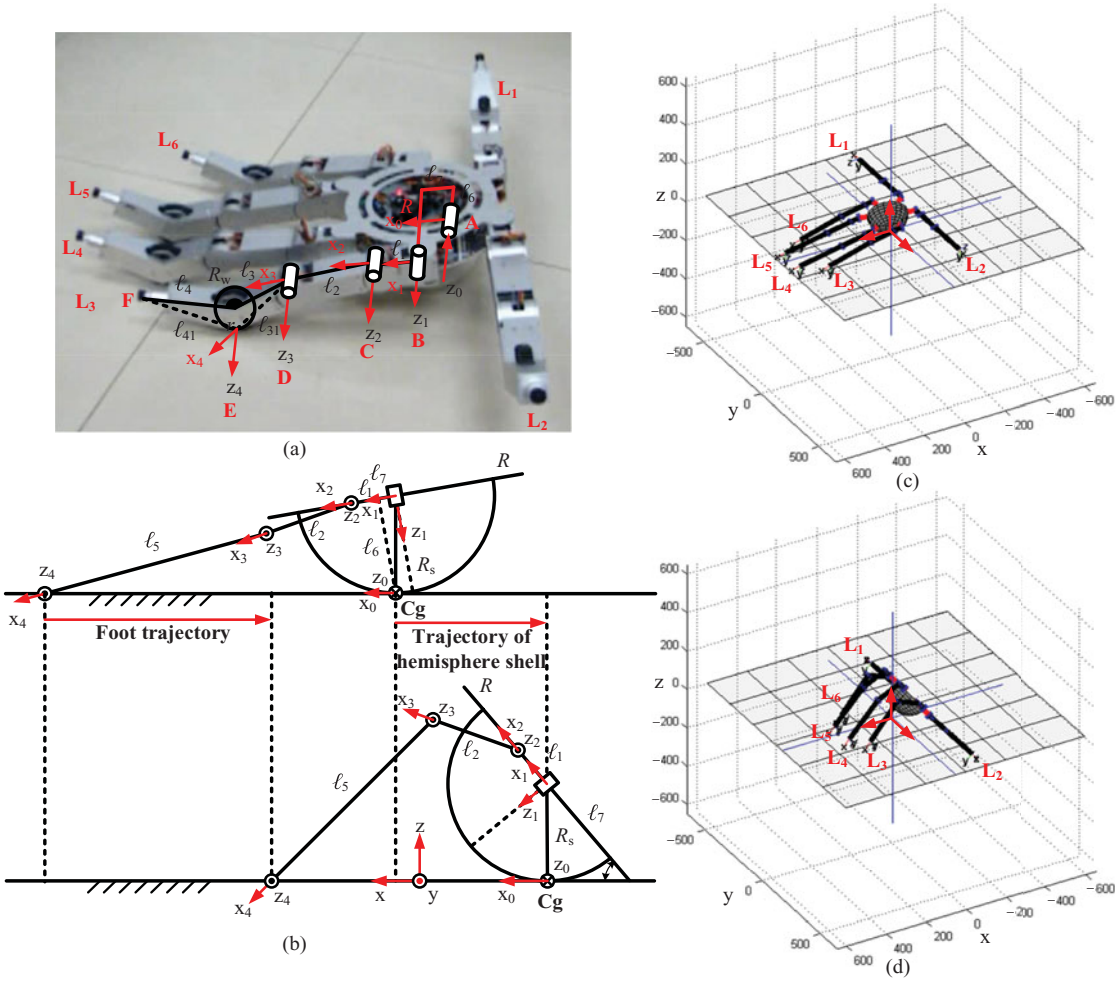


Fig. 6. The sketch diagram of L_3 in Stage 2. (a) shows the overturned NOROS robot in Stage 2 and the equivalent manipulator of L_3. The foot trajectory of L_3 and the hemisphere shell trajectory in Stage 2 are shown in (b). Following the foot trajectory, the robot can self-recover from the posture in the top of (b) corresponding to the 3D diagram in (c) to the posture in the bottom of (b) corresponding to the 3D diagram in (d). The two 3D diagrams in (c) and (d) are drawn in MATLAB 2010b.

The foot trajectory is given as follows:

$$p(t) = \begin{bmatrix} x(t) & R & 0 \end{bmatrix}^{\mathrm{T}},\ t \in [t_1, t_2] \tag{3}$$

where $x(t)$ is a linear function of time.

The position of the contact point Cg is a function of the roll angle β:

$$p_{Cg}(t) = \begin{bmatrix} -R_s\beta(t) & 0 & 0 \end{bmatrix}^{\mathrm{T}},\ t \in [t_1, t_2] \tag{4}$$

where $\beta(t_1) = \beta_{\min},\ \beta(t_2) = \beta_{\max}$.

https://doi.org/10.1017/S0263574715001009 Published online by Cambridge University Press

Table V. The DH parameters of an equivalent manipulator of L_3 in Stage 3.

Joints	θ_j	d_j	α_j	a_j
1	θ_1	0	$\pi/2$	ℓ_7
2	θ_2	0	$\pi/2$	ℓ_1
3	θ_3	0	0	ℓ_2
4	θ_4	0	0	ℓ_5

Table VI. The ranges of the joints of the equivalent manipulator in Stage 3.

Joint 1 [degree]	Joint 2 [degree]	Joint 3 [degree]	Joint 4 [degree]
[θ_{10},90]	[−180,0]	[−45, 135]	[−120, 60]

By determining the base position and the end position, the inverse kinematics of the equivalent manipulator could be solved in Stage 2. In addition, the motion of θ_1 is passive and must follow the rules: $\theta_1(t) = \beta(t)$.

Therefore, the movement of the joints (θ_2, θ_3, θ_4) of L_3 could be achieved by solving the inverse kinematics[27] of the equivalent manipulator.

Stage 3: Self-recovery motion with L_is (i = 1, 2) as a rotation shaft

In this stage, the most important characteristic is the robot body rotates with a settled shaft produced by L_is(i = 1, 2) in contact with the ground. The treatment is similar to that of Stage 2 by adding the body rotation to the legs and establishing equivalents to four DoFs manipulators. The leg L_3 is assigned as the analysis object.

The implementation of the motion of L_3 is shown in Fig. 7. Under the limitations of the leg joints, the foot of L_3 cannot continue moving along a straight line. A new foot trajectory must be attained to complete the motion. Here, the DH parameters of the equivalent manipulator are described in Table V. The ranges of joints are presented in Table VI. All joints in Stage 3 should acquire values within the ranges:

$$\theta_{i.\min} \le \theta_i \le \theta_{i,\max},\ i = 2, 3, 4 \tag{5}$$

For the settled shaft, the values of ℓ_6 and ℓ_7 can be obtained by:

$$\begin{cases} \ell_6 = 0 \\ \ell_7 = R\sin(60^\circ) \end{cases} \tag{6}$$

The robot body continuously rotates and recovers with the settled shaft. As a result, the angle of joint 1 of the equivalent manipulator should vary continuously to maintain stable. Note that the process of orienting the robot body to be upright is equivalent to joint 1 varying from θ_{10} to 90 degree. θ_{10} is an angle between the ground and the cross section of the robot body when the Lis (i = 1, 2) touches the ground. We generally take $\beta_{\max}$ as the value of θ_{10}.

When joint 1 reaches 90 degree, the robot body is completely upright. Thus, θ_1 is constrained by:

$$\theta_1(t) = \theta_{10} + \frac{(90 - \theta_{10})}{t_3 - t_2}(t - t_2),\ t \in [t_2, t_3] \tag{7}$$

Here, a foot trajectory is given:

$$\begin{cases} p_x(t) = x_{\mathrm{Cg}} + R_{\mathrm{traj}}\cos(\gamma(t)) \\ p_y(t) = y_{\mathrm{Cg}} + R_{\mathrm{traj}}\sin(\gamma(t)), \quad t \in [t_2, t_3] \\ p_z(t) = 0 \end{cases} \tag{8}$$

https://doi.org/10.1017/S0263574715001009 Published online by Cambridge University Press

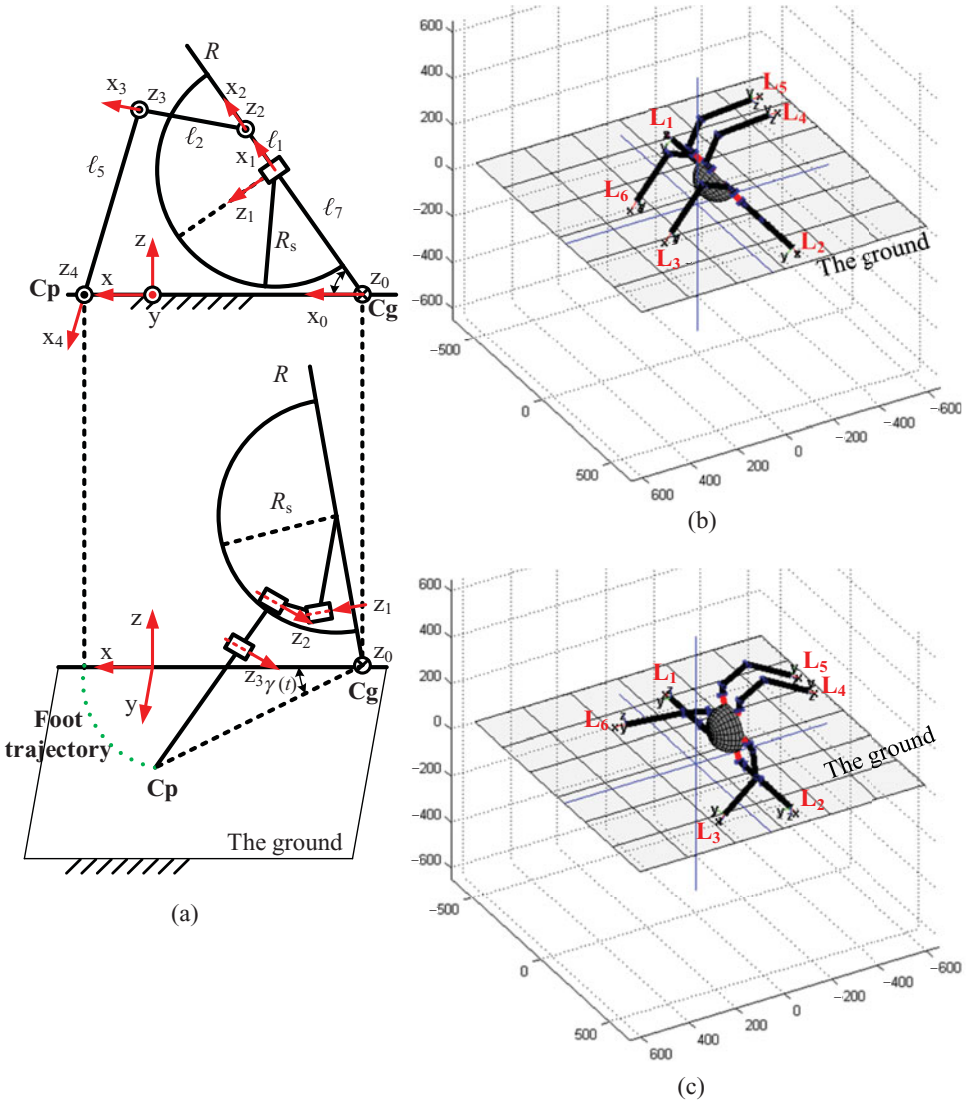


Fig. 7. The sketch diagrams of L_3 in Stage 3. The arc foot trajectory of L_3 in Stage 3 are shown in (a). Following the foot trajectory, the robot can self-recover from the posture in the top of (a) corresponding to the 3D diagram in (b) to the posture in the bottom of (a) corresponding to the 3D diagram in (c). The two 3D diagrams in (b) and (c) are drawn in MATLAB 2010b. In this stage, only two legs (L_3 and L_6) can be employed to perform the self-recovery motions.

where R_{traj} is the radius of the foot trajectory, and

$$\gamma(t) = \frac{45^\circ}{t_3 - t_2}(t - t_2), t \in [t_2, t_3] \tag{9}$$

By determining $\theta_1(t)$ and the foot trajectory $p(t)$ and using the DH parameters, the analytic solution of the inverse kinematics of the equivalent can be attained. Thus, we can determine the values of (θ_2, θ_3, θ_4) to control the legs to push and recover the robot body.

Stage 4: Self-recovery motion under gravity

In Stage 3, the legs L_is (i = 3, 4, 5, 6) contact the ground with their feet to exert the ground reaction forces required to support and push the robot body upright. The force of gravity on the robot body contributes negative work to accomplish the self-recovery motion described in the previous three stages. In contrast, it is the forces of the gravity of the robot body and L_is (i = 4, 5) that compel the robot to fall to the ground in this stage. The six legs only keep the respective joints stationary, i.e.,

https://doi.org/10.1017/S0263574715001009 Published online by Cambridge University Press

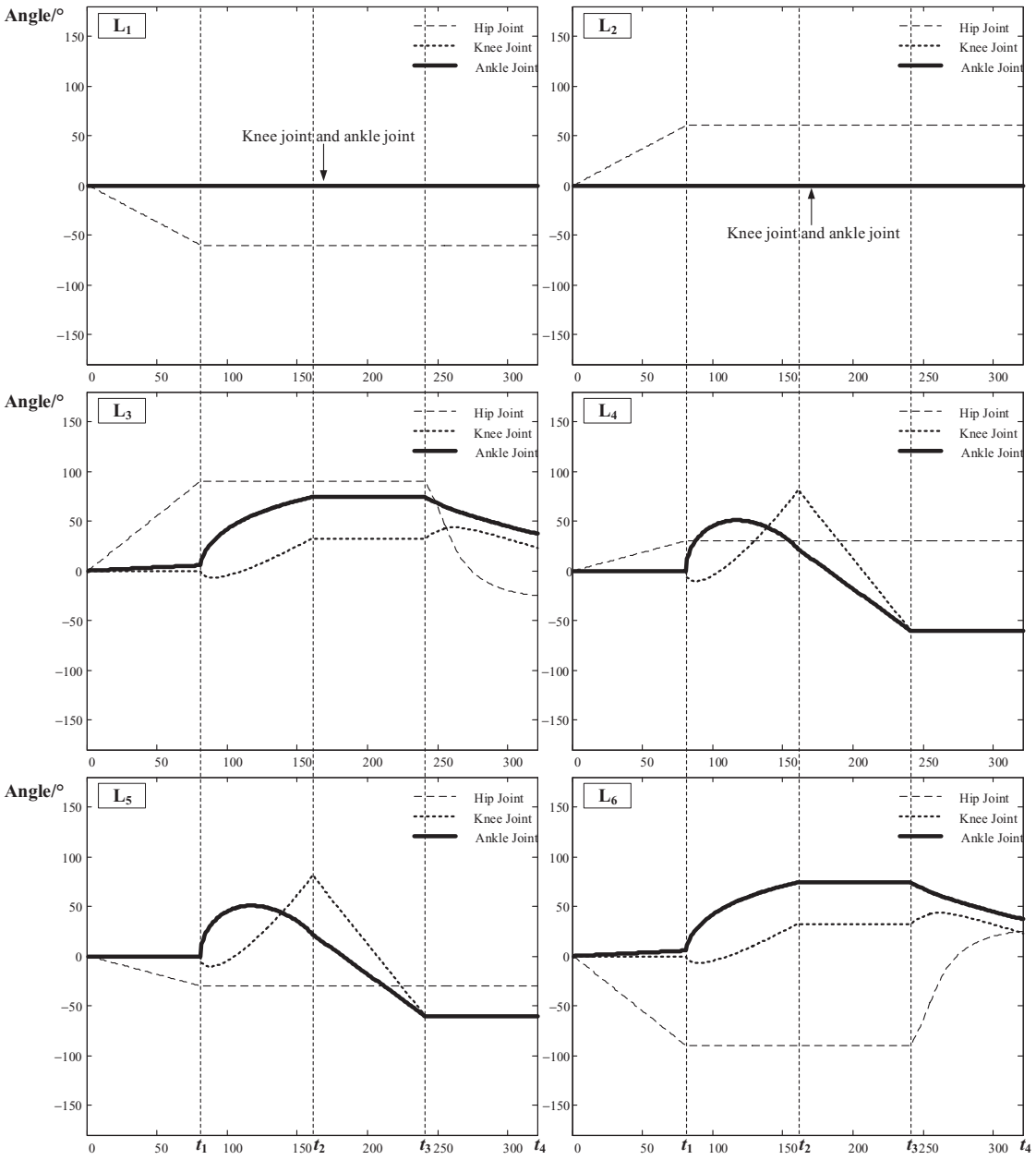


Fig. 8. Variations of the hip, knee, ankle joints of L_is (i = 1, 2, 3, 4, 5, and 6).

L_is (i = 1, 2) remain in contact with the ground and form the rotational shaft of the body falling, and L_is (i = 3, 4, 5, 6) retain their postures in the air and fall naturally under the force of gravity.

The falling process will result in the legs L_is (i = 4, 5) first reaching the ground instead of the robot body to avoid damage of the body due to impact with the ground. This is a vital step in the last stage of recovery. Finally, the robot body will reach the ground gently with the extending motion and the support of L_is (i = 4, 5). At this point, the robot has completed the self-recovery motion successfully.

6. The Experiments of the NOROS Self-Recovery Process

NOROS is used to verify the motion planning and implementation of the SSR method for self-recovery. The drive servo-motors of NOROS are controlled by position, which makes each robot leg move easily via issuing of the angle values to the respective joints. Obviously, the self-recovery motion is a quasi-static motion of the NOROS robot.

https://doi.org/10.1017/S0263574715001009 Published online by Cambridge University Press

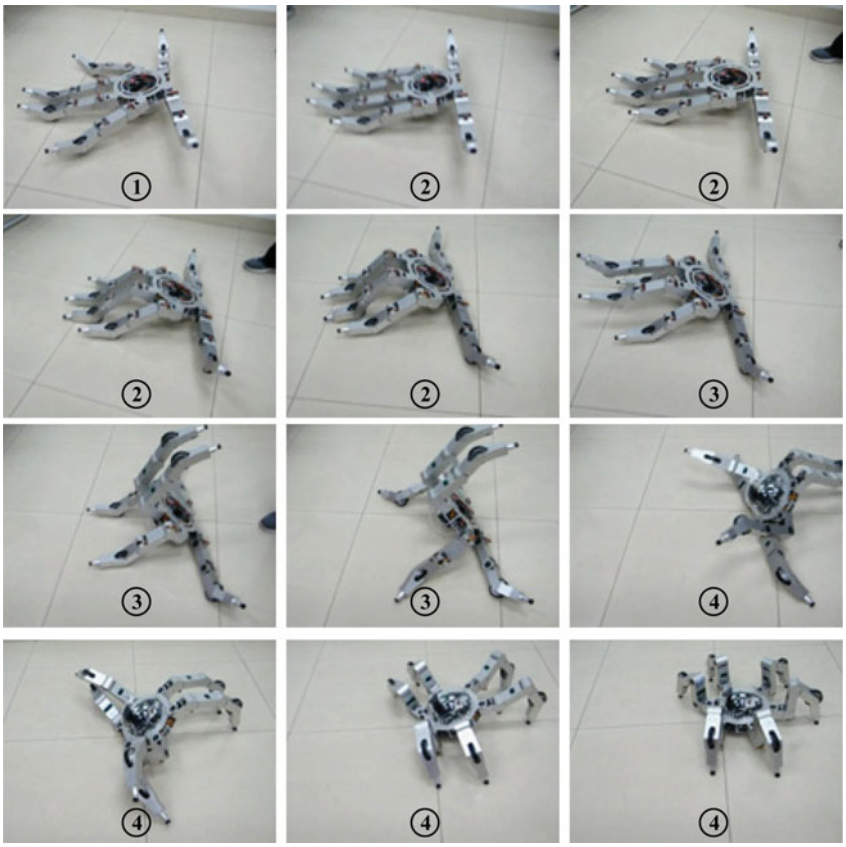


Fig. 9. Self-recovery experiment for NOROS. The number in the bottom of each picture characterizes the stage of the self-recovery motions.

According to the planning and implementation of the self-recovery motion of the robot, the foot trajectories of the six legs are respectively established in each stage. The inverse kinematics of the legs or the equivalent manipulator are used to calculate all of the values of the joints. By setting the joint angles, the feet of the six legs will move along the desired trajectories. The variations of each joint of each leg are shown in Fig. 8. The time range of Stage 1 is $[0, t_1]$, that of Stage 2 is $[t_1, t_2]$, and that of Stage 3 is $[t_2, t_3]$.

Some key postures of NOROS during the self-recovery process are captured from the experiment, as shown in Fig. 9. The number in each picture bottom is the stage identification of the self-recovery process of NOROS. The last picture shows that NOROS has successfully performed the self-recovery process after being in an overturned situation, which confirms that the SSR method is feasible for NOROS.

Conclusion and Discussion

Multi-legged robots are walking out from laboratories and now are ready to perform hazardous tasks, which enables humans to avoid performing these tasks. An unexpected overturned situation may occur at any time when robots performing tasks. The robot should enable to perform self-recovery and continue to complete the task. Therefore, a self-recovery solution for multi-legged robots, named the SSR method, was presented in this paper. The self-recovery of an overturned insect was first introduced and three crucial points for the self-recovery were concluded. The insects utilize and control their legs to perform self-recovery without additional help. The dangling legs move to bias the body and the landing legs support and push the body to enable recovery. Detailing the motion of each leg provides a way to implement the SSR method. The similarities between multi-legged robots and insects indicates that the SSR method can be applied on multi-robots for performing self-recovery. We planned and implemented the self-recovery motion of a multi-legged robot. The motion

https://doi.org/10.1017/S0263574715001009 Published online by Cambridge University Press

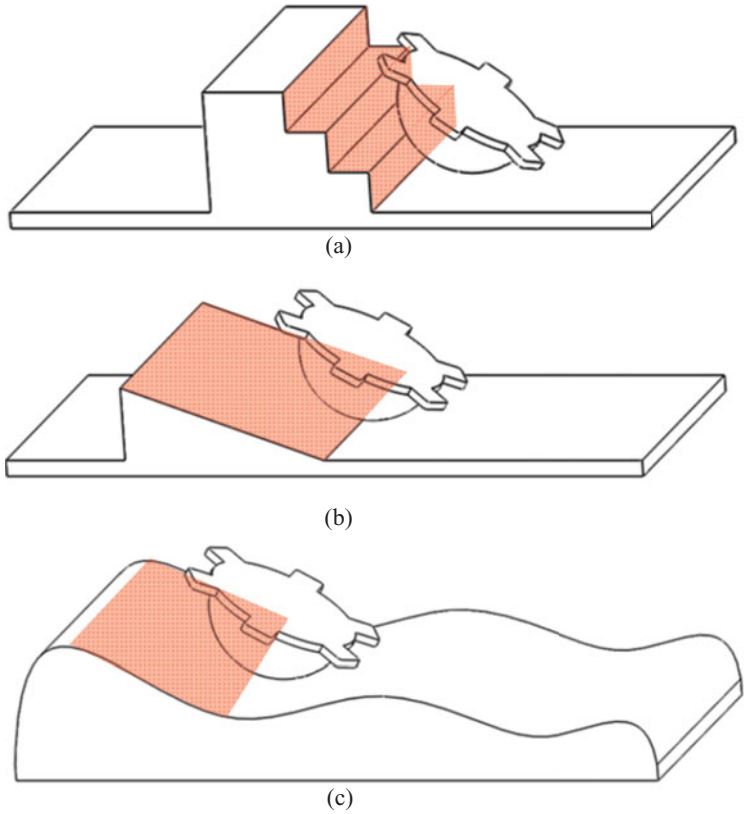


Fig. 10. (a) The terrain is characterized by stairs. (b) The terrain is characterized by a slope surface. (c) The terrain is characterized by a curved surface. Obviously, compared with the flat terrain (as shown in Fig. 3), the Stage 1 and Stage 2 are not required on the listed terrains ((a), (b) and (c)), which will save energy for the robot accomplishing the first two stages of self-recovery motions. The red areas (transparency 63%) are the convex portions which the feet and the body could employ to easily obtain the adequate ground reaction forces and reduce the joint torque for the self-recovery motions.

was divided into four stages and in each stage the movement of each leg of the robot was given. In addition, the robot named NOROS was introduced and used to perform self-recovery for validating the feasibility of the SSR method. The results of the self-recovery experiments demonstrated that the NOROS can successfully complete the self-recovery motion according to the planned motions, the SSR is a suitable method for robots that have six or more legs.

However, the effectiveness of the SSR method has only been tested in the flat terrain, and only the kinematics of motion was used to obtain the analytic solutions of the joints. The four stages for self-recovery motion are quasi-static, and the dynamics effects of the swing legs have been neglected. Actually, some rough terrains could help the robot for self-recovery. When the robot is overturned on a rough terrain, such as stairs, a slope or a convex curve surface (as shown in Fig. 10), the convex portions of the rough terrain can support the feet of legs for more easily obtaining the adequate ground reaction forces than flat terrain. In fact, the most challenge condition to prevent the robot from recovering is that the robot body is blocked at the bottom of the steep pit, this is a very rare circumstance. In future work, posture sensors (IMU) mounted on the centre of the body and force sensors mounted on the ends of the feet will be employed to measure the posture of the body and to detect whether the feet have contacted with the terrain respectively. Thus, the motion planning method for the robot self-recovering in flat terrain will be extended to the rough terrain case with consideration of dynamics analysis for high performance and adaptability.

Acknowledgements
The work has been supported by the National Science Foundation for Distinguished Young Scholars, China (Grant No. 51125020).

https://doi.org/10.1017/S0263574715001009 Published online by Cambridge University Press

References

1. K. J. Waldron, V. J. Vohnout, A. Pery and R. B. McGhee, "Configuration design of the adaptive suspension vehicle," *Int. J. Robot. Res.* **3**(2), 37–48 (1984).
2. B. H. Wilcox, T. Litwin, J. Biesiadecki, J. Matthews, M. Heverly, J. Morrison, J. Townsend, N. Ahmad, A. Sirota and B. Cooper, "ATHLETE: A cargo handling and manipulation robot for the moon," *J. Field Robot.* **24**(5), 421–434 (2007).
3. J. A. Cobano, J. Estremera and P. G. de Santos, "Accurate tracking of legged robots on natural terrain," *Auton. Robot.* **28**(2), 231–244 (2010).
4. K. C. Galloway, J. E. Clark and D. E. Koditschek, "Variable stiffness legs for robust, efficient, and stable dynamic running," *J. Mech. Robot.* **5**(1), 0110091 (2013).
5. U. Saranli, M. Buehler and D. E. Koditschek, "RHex: A simple and highly mobile hexapod robot," *Int. J. Robot. Res.* **20**(7), 616–631 (2001).
6. R. B. McGhee and A. A. Frank, "On the stability properties of quadruped creeping gaits," *Math. Biosci.* **3**, 331–351 (1968).
7. R. B. McGhee and G. I. Iswandhi, "Adaptive locomotion of a multilegged robot over rough terrain," *IEEE Trans. Syst. Man and Cybern.* **9**(4), 176–182 (1979).
8. C. Zhang and S. Song, "Gaits and geometry of a walking chair for the disabled," *J. Terramech.* **26**(3), 211–233 (1989).
9. C. D. Zhang and S. M. Song, "Stability analysis of wave-crab gaits of a quadruped," *J. Robot. Syst.* **7**(2), 243–276 (1990).
10. D. A. Messuri, "Optimization of the Locomotion of a Legged Vehicle with Respect to Maneuverability," *Ph.D. Thesis* (The Ohio State University, Columbus, 1985).
11. D. E. Orin, "Interactive Control of a Six-Legged Vehicle with Optimization of Both Stability and Energy," *Ph.D. Thesis* (The Ohio State University, Columbus, 1976).
12. K. Yoneda and S. Hirose, "Tumble Stability Criterion of Integrated Locomotion and Manipulation," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 1996)*, Osaka, Japan (Nov. 4–8, 1996), vol. 2, pp. 870–876.
13. Z. Guanghua, D. Zhicheng and W. Wei, "Realization of a Modular Reconfigurable Robot for Rough Terrain," *IEEE International Conference on Mechatronics and Automation*, Luoyang, China (Jun. 25–28, 2006) pp. 289–294.
14. C. C. Kessens, D. C. Smith and P. R. Osteen, "A Framework for Autonomous Self-Righting of a Generic Robot on Sloped Planar Surfaces," *IEEE International Conference on Robotics and Automation (ICRA 2012)*, Saint Paul, MN (May 14–18, 2012) pp. 4724–4729.
15. J. Collins, C. C. Kessens and S. J. Biggs, "Proprioceptive Sensing for Autonomous Self-Righting on Unknown Sloped Planar Surfaces," *IEEE International Symposium on Robotic and Sensors Environments (ROSE)*, Washington, DC (Oct. 21–23, 2013) pp. 160–165.
16. P. R. Roan, A. Burmeister, A. Rahimi, K. Holz and D. Hooper, "Real-World Validation of Three Tipover Algorithms for Mobile Robots," *IEEE International Conference on Robotics and Automation (ICRA 2010)*, Anchorage, AK (May 3–7, 2010) pp. 4431–4436.
17. U. Saranli, A. A. Rizzi and D. E. Koditschek, "Model-based dynamic self-righting maneuvers for a hexapedal robot," *Int. J. Robot. Res.* **23**(9), 903–918 (2004).
18. Y. Jung-Min and K. Jong-Hwan, "Fault-tolerant locomotion of the hexapod robot," *IEEE Trans. Syst. Man Cybern.—part b: Cybern.* **28**(1), 109–116 (1998).
19. L. Frantsevich, "Righting kinematics in beetles (Insecta: Coleoptera)," *Arthropod Struct. Dev.* **33**(3), 221–235 (2004).
20. G. Domokos and P. L. Várkonyi, "Geometry and self-righting of turtles," *Proc. R. Soc. B*, **275**(1630), 11–17 (2008).
21. G. Ana, T. Ljiljana and I. Ana, "Geometry of self righting: The case of Hermann's tortoises," *Zool. Anz.* **254**, 99–105 (2015).
22. Website: http://content.mbc.co.kr/english/documentary/Nature/2386551_56289.html.
23. Website: http://www.shixuyao.com/noros.html.
24. Z. Wang, X. Ding, A. Rovetta and A. Giusti, "Mobility analysis of the typical gait of a radial symmetrical six-legged robot," *Mechatronics* **21**(7), 1133–1146 (2011).
25. K. Xu and X. Ding, "Typical gait analysis of a six-legged robot in the context of metamorphic mechanism theory," *Chin. J. Mech. Eng.* **26**(4), 771–783 (2013).
26. X. Ding, K. Li and K. Xu, "Dynamics and wheel's slip ratio of a wheel-legged robot in wheeled motion considering the change of height," *Chin. J. Mech. Eng.* **25**(5), 1060–1067 (2012).
27. J. J. Craig, *Introduction to Robotics: Mechanics and Control*, (Upper Saddle River, NJ, USA: Pearson/Prentice Hall, 2005).

https://doi.org/10.1017/S0263574715001009 Published online by Cambridge University Press