2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

A General Locomotion Approach for a Novel Multi-legged Spherical Robot

Dun Yang[1], Yunfei Liu[1] and Yang Yu[1]

***Abstract*— As a kind of ground mobile robot, deformable robots have many advantages, such as solid terrain adaptability, lightweight, and portability. Among these robots, the radial skeleton robot has better stability and controllability. However, because the friction of foot and ground is hard to be predicted, the accuracy of its gait generation algorithms that have been studied is very low. Furthermore, there is currently no closed-loop control scheme for this kind of robot. We designed a 12-legged radial skeleton robot with high extension ratio legs, proposed a high-precision gait generation algorithm for any multi-legged radial skeleton robot, and first proposed a closed-loop control scheme for this kind of robot. A dynamic model considering contact friction is established. And the robot has the advantages of omnidirectional motion, high-precision trajectory tracking, and motion robustness. By conducting prototype experiments, it is verified that our method achieves the highest accuracy when tracking trajectory and holds robustness in the unknown environment.**

I. Introduction

The existing well-developed wheeled and legged robots [1]–[5]are not so suitable for some ground detection tasks in extreme terrains, such as cave inspection [6], planetary surface exploration [7], jungle explosion removal [8], and ruins infiltration [9]. It is because those traditional mobile robots hold fixed body structures, and it is hard to maintain body stability when moving [10]. Meanwhile, they can't move with omni-direction in any initial postures, which means that once fall, it will be difficult to get up in the tough environment. Shape-changing mobile robots, with symmetry structure and simple movement mode, are proposed as a promising solution for the above tough terrain exploration tasks. The potential of adjusting the contacting surface ensures they fit the unstructured ground, scale-variable crevices, and confined space. From the perspective of structure features, three-dimensional shape-changing robots could be categorized into three types: edge skeleton robots, tensegrity robots, and radial skeleton robots. Among them, the last type has higher motion stability.

Radial skeleton robot: With the multi-radial legs and spherical structure, radial skeleton [11]–[16] is a much more bio-inspired robot, aroused by the mimic of sea-urchin and tumbleweed. NASA has proposed a tumbleweed-like planetary probe [11], which can be driven by the Martian wind to realize exploration but does not have autonomous control capability. Gheorghe et al. [12] proposed a spherical robot similar to the sea urchin for the first time, which uses the telescopic mechanism in the sphere to drive the robot. Researchers in CMU designed a 5-legged pneumatic planar robot prototype [17] and spherical 12-legged pneumatic robot [13]. The spherical robot with 32 retractable legs developed by researchers at the University of Tokyo in Japan [14], [15] realizes the continuous motion mode of straight and curve based on the kinematics method. MIT researchers designed a high expansion ratio mechanism [16] used in the prototype of the sea-urchin robot. Researchers at Harvard University designed a soft sea urchin robot [18] from the perspective of soft materials and soft sensors and could achieve simple planar locomotion. Compared with the other two kinds of three-dimensional shape-changing robots, losing the ability of deformation and flexibility to a certain extent, radial skeleton robots become more controllable and predictable. These attributes are important qualities in the category of deformation robots.

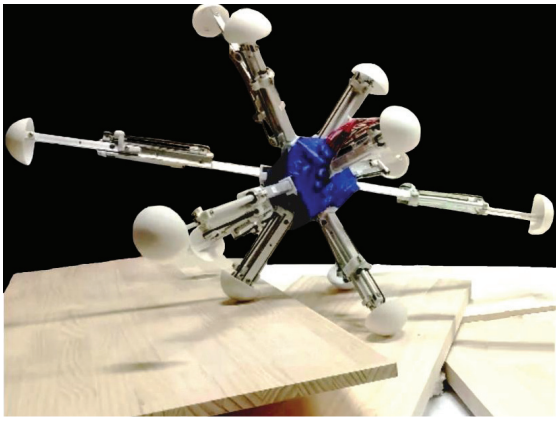

Fig. 1: 12-legged radial skeleton robot

However, the current locomotion control methods on radial skeleton don't leverage the above advantages, leading to the over simple and incomplete gait planning and control frame. The discrete manual-designed method cannot take full advantage of omnidirectional motion and is too cumbersome [13]. The inverse kinematics planning method, in the prototype experiments, is far beyond the desired distance [14]. The reason is that the front landing leg is pushed forward by the rear extended legs during the rolling process, which can't be predicted in the kinematic simulation model. Zhang [19] proposed a method based on physics-driven locomotion planning, using an artificial dynamic model, which could generate gait sequences in complex virtual terrain. However, based on too many assumptions, it is far from reality, and real-world movement cannot be achieved. Some ML methods [20] could be a constructive idea, but the generalization and

*This work is supported by National Natural Science Foundation of China Grant No. 12022212.

[1]Beihang University, Beijing, China dunyang@buaa.edu.cn, yu.yang@buaa.edu.cn

sim2real gap are difficult to bridge. One main challenge for the above problems is to predict and control the relative sliding between the legs and the contact surface. Meanwhile, the high-precision omnidirectional closed-loop gait control scheme remains a gap.

We propose a gait control framework that enables any multi-legged radial skeleton robot to carry out omnidirectional gait real-time planning under any posture and realize high-precision trajectory tracking control. A 12-legged radial skeleton robot prototype with a high expansion ratio is designed to validate the performance. Through the gait planning comparison experiments and closed-loop control experiments, the advantages of high accuracy and high robustness of the algorithm are verified. It is expected to become a unified control framework for radial skeleton robots.

The contributions are as follows:

- A high-precision gait control algorithm for an arbitrary multi-legged radial skeleton robot is proposed. Contact friction is considered in gait generation, and the robot can start omnidirectional motion from any initial attitude.
- A 12-legged radial skeleton robot with a high telescopic ratio is designed and implemented. The dynamic model is established to analyze and simulate motion.
- Build an experimental system, systematically compare the advantages and disadvantages of the current radial skeleton robot gait generation algorithms, and verify this method's advantages of high precision and robustness.

II. 12-LEGGED RADIAL SKELETON ROBOT STRUCTURE DESIGN

For the spherical radial-legged robot, the high extension ratio mechanism is vital to the robot's deformation ability, which influences the shaping changing amplitude and motion performance [20].

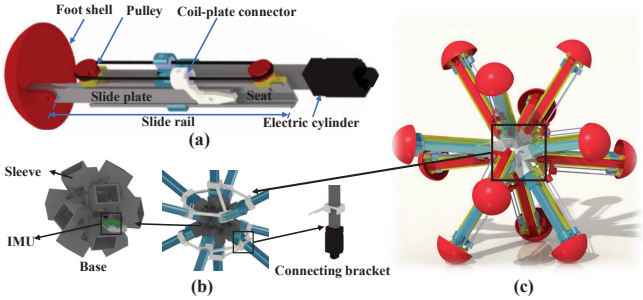

Fig. 2: (a) Telescopic mechanism (b) Components (c) Robot body

Telescopic mechanism: When designing the telescopic mechanism, it should also balance the requirements of the high elongation ratio, low complexity, large output, low weight, and so on. The high extension ratio telescopic mechanism designed in this paper adopts the two-way pulley mechanism. As seen in Fig. 2(a), the leg is driven by an electric cylinder fixed on the robot base, and the lightweight aluminum slide rail is used to expand the extension ratio. An electric cylinder and putter compose a linear actuator. The slide plate, rail seat, and pulley compose the slide rail. Coil-cylinder connectors, coil-plate connectors, and putter-seat connectors are 3D printing materials to constraint movement. When the liner actuator starts moving, the electric cylinder is fixed. Meanwhile, the pulley, rail seat, and connector begin telescoping along with the putter. It is the first part of the link. With the constraint of a fixed coil-cylinder connector and coil-plate connector, the slide plate will move under the drive of the putter. The telescopic mechanism performance is shown in Table. I.

TABLE I: Parameters of extension mechanism

Parameter	Value
Extension ratio	2.08
Max length	384.121mm
Minimum length	184.121mm
Actuator extension speed	2.8cm/s
Max force	25N
Back drive force	25N
Overall weight	144.2g

Robot body and system: The selection of telescopic feet number needs to consider many factors, such as weight, deformation capacity, design difficulty, etc. Increasing the number of telescopic feet can improve the robot's deformation ability. However, more telescopic feet will also have adverse effects such as weight increase and size increase of central connector. As shown in Fig. 2(c), by simulation experiments to test the motion performance of the different numbers of legs, we select to arrange 12 telescopic legs on the central vertical line of the dodecahedron's face, trading off the designing difficulties and motion ability. The symmetrical distribution of legs can bring the advantages of omni-direction and all-posture movement ability.

TABLE II: Parameters of the robot

Parameter	Value
Extension ratio	1.97
Max diameter	811.84mm
Minimum diameter	411.84mm
Total mass	1819.6g

The central base connects multiple telescopic mechanisms, seen in Fig. 2(b), in the form of sleeve and bolt combinations. To improve the elongation ratio of the robot, the radial size of the non-extensible parts should be as small as possible. The center base is hollowed out to reduce the weight and install a small-size IMU. The IMU can measure the orientation of the central base, in the form of the Euler angle, which will further be used in the gait planning algorithm. We use Arduino as the microcontroller to control the robot, which is mounted in the base. Electric cylinders often bear axial forces in common use, and the bearing capacity for vertical forces is poor. However, its landing legs will also be subjected to vertical force when the robot moves. After the test, the connection between the motor and the cylinder is the worst part of the whole robot, which will bend during the movement. Therefore, we design a sleeve and connecting bracket for the connection to increase structural strength. The prototype performance is shown in Table. II.

III. Dynamic model and locomotion method

The radial skeleton robots mainly move in the way of rolling by the means that the center of gravity moves out of the support triangle by expanding its legs. The friction between the leg end and the ground, during its leg's expansion and contraction, will significantly impact the robot's motion accuracy. Meanwhile, the rolling process may also cause an impact between the leg-end and the ground, which affect motion accuracy and damage the structure. This section will introduce high-precision dynamic models, gait generation, and gait control methods to solve the above problems.

A. Dynamic model and contact force calculation

Dynamic model: We use a single-rigid body to describe our radial skeleton robot because the main mass of the robot is concentrated on the base, and subsequent experiments have also proved the accuracy of this simplified model is sufficient in the process of the relatively slow movement. The state variables of the robot can be described as follows:

$$\boldsymbol{q} = \left[x, y, z, \lambda_0, \lambda_1, \lambda_2, \lambda_3, v_x, v_y, v_z, \omega_x, \omega_y, \omega_z\right]^{\mathrm{T}} \tag{1}$$

where $[x, y, z, \lambda_0, \lambda_1, \lambda_2, \lambda_3]$ represents the centroid position in inertial frame and body-fixed coordinate orientation with the form of quaternions. $\left[v_x, v_y, v_z, \omega_x, \omega_y, \omega_z\right]$ represents the centroid velocity in the inertial frame and the angular velocity in the body axes. The above state variables hold the following dynamic relationships which are deducted by Newton equation:

$$\left[\dot{x}, \dot{y}, \dot{z}, \dot{v}_x, \dot{v}_y, \dot{v}_z\right]^{\mathrm{T}} = \left[v_x, v_y, v_z, \frac{F_x}{m}, \frac{F_y}{m}, \frac{F_z}{m}\right]^{\mathrm{T}} \tag{2}$$

Where m is the weights of the robot and $\left[F_x, F_y, F_z\right]^{\mathrm{T}}$ is the force vector acting on the center of mass, including contact forces and gravity forces. Taking the derivative of quaternion to get:

$$\left[\dot{\lambda_0}, \dot{\lambda_1}, \dot{\lambda_2}, \dot{\lambda_3}\right]^{\mathrm{T}} = \frac{1}{2}\begin{bmatrix} -\lambda_1 & -\lambda_2 & -\lambda_3 \\ \lambda_0 & -\lambda_3 & \lambda_2 \\ \lambda_3 & \lambda_0 & -\lambda_1 \\ -\lambda_2 & \lambda_1 & \lambda_0 \end{bmatrix}\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \tag{3}$$

Using the Euler equation to obtain the derivation of the angular velocity, and selecting the inertial spindle coordinate system as the body axes, we can get:

$$\left[\dot{\omega}_x, \dot{\omega}_y, \dot{\omega}_z\right] = \left[\frac{M_x^{\tau}}{J_x}, \frac{M_y^{\tau}}{J_y}, \frac{M_z^{\tau}}{J_z}\right] \tag{4}$$

Where J is the inertia tensor,M is the external moment in the body axes.

Combining the above equations (2)(3)(4), we obtain the dynamic of the robot systems.

And the next key point is calculating the contact force. When the robot moves in the real world, it will have complex contact, collision, sliding, and separation behaviors with different types of ground. We introduce the discrete element method [21] used in discrete dynamics to describe the interaction between the leg-end and ground because this method describes the contact force and friction force through the relative deformation degree between the contact objects, which has wider adaptability and higher accuracy for different contact conditions.

Normal contact force: We first consider the single leg embedment depth of the sphere shell and ground, as shown in Fig. 3(a):

$$\delta_N = \left|\boldsymbol{r}_l + \boldsymbol{r}_s - \boldsymbol{r}_g\right| \tag{5}$$

Where the $\boldsymbol{r}_l$ is the leg-end shell center vector in the inertial coordinate system.

When the sphere shell, with radius $\boldsymbol{r}_s$ and position $\boldsymbol{r}_l$, and the ground, with position $\boldsymbol{r}_g$, contact each other, the degree of deformation caused by normal extrusion can be expressed by the amount of overlap δ_N. The normal contact force acting connect point could be calculated as

$$\mathbf{F}_N = k_N \delta_N \vec{\boldsymbol{n}} + C_N \mathbf{u}_N \tag{6}$$

The first item is the normal elastic restoring force produced by extrusion deformation, given by Hooke's law of linear elasticity. And the second item is viscous dissipation force due to particle viscoelasticity and normal relative motion. Where, normal unit vector is $\vec{\boldsymbol{n}} = \left(\boldsymbol{r}_g - \boldsymbol{r}_l\right) / \left|\boldsymbol{r}_g - \boldsymbol{r}_l\right|$. k_N is the normal elastic coefficient, reflecting granular materials' stiffness. C_N is the normal damping coefficient.

The normal relative velocity of two particles at the contact point

$$\mathbf{u}_N = \left[\left(-\mathbf{v}_p\right) \cdot \vec{\boldsymbol{n}}\right] \vec{\boldsymbol{n}} \tag{7}$$

$\mathbf{v}_p$ is the linear velocity of the contact point P_c, which is defined as the center of embedment depth along with the sphere shell radius, as seen in Fig. 3(a). $\mathbf{v}_p$ could be calculated as follows:

$$\mathbf{v}_p = \mathbf{v}_{\text{base}} + \mathbf{w}_{\text{base}} \times \mathbf{r}_p + \mathbf{v}_{\text{expend}} \tag{8}$$

Where the $\mathbf{v}_{\text{base}}$ and $\mathbf{w}_{\text{base}}$ is the velocity and angular velocity of the base, and the $\mathbf{v}_{\text{expend}}$ is the leg's radial expansion speed.

Parameter selection: The mechanical properties of the normal contact force are mainly controlled by the corresponding elastic coefficient k_N and damping coefficient C_N, which reflect the strength and energy dissipation capacity of the sphere shell materials under external pressure. In the experiment, we select these coefficients by comparing the simulation and prototype experimental phenomena. To be specific, the prototype actually contacts the rigid ground, with the large elastic coefficient k_N. When applying this k_N into the simulation, a very small simulation step size is necessary to avoid excessive contact force but leads to a large amount of calculation time. To trade off the calculation duration and simulation accuracy, we adopt the softened elastic coefficient k_N and a larger step size.

Tangential contact force: We first research the tangential relative motion between the sphere shell and the ground. The surface microstructures of the shell and ground will squeeze in the vertical direction of $\vec{n}$, resulting in the deformation that can be described by the tangential contact deformation vector:

$$\boldsymbol{\delta}_S = \int_{t_0}^{t} \boldsymbol{u}_S(\tau) d\tau + \boldsymbol{\delta}_{S0} \tag{9}$$

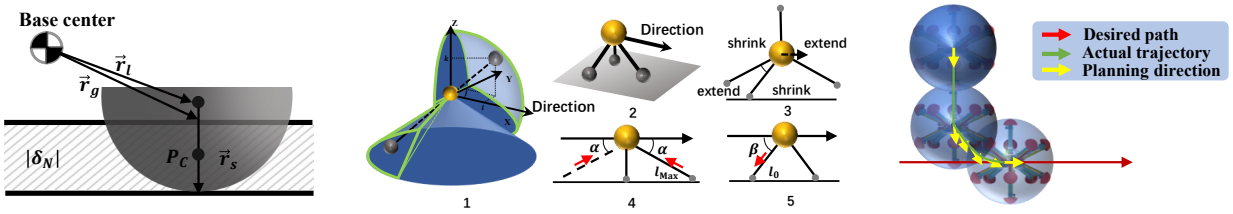

Fig. 3: (a) Contact model between spherical shell and ground. (b) Gait planning diagram: 3D perspective and plane perspective. (c) Trajectory tracking method

Where $\boldsymbol{\delta}_{S0}$ is the tangential contact deformation vector at time t_0 (if time t_0 is the initial contact time, $\boldsymbol{\delta}_{S0}$ is 0). The tangential contact force model could be calculated as

$$\mathbf{F}_S = \begin{cases} k_S \boldsymbol{\delta}_S + C_S \mathbf{u}_S, & |k_S \boldsymbol{\delta}_S| < \mu_S F_N \\ \mu_S F_N \boldsymbol{\delta}_S / |\boldsymbol{\delta}_S|, & |k_S \boldsymbol{\delta}_S| \geq \mu_S F_N \end{cases} \tag{10}$$

Similar to the normal contact model, the first item is the tangential elastic restoring force generated by tangential extrusion deformation, and the second item is the tangential viscous dissipation force. Tangential relative velocity at the contact point is: $\mathbf{u}_S = \mathbf{v}_p - \mathbf{u}_N$. Limited by contact surface shear strength, when $k_S \boldsymbol{\delta}_S$ reaches a certain value, a plastic slip will occur on the contact surface of particles. The slip conditions can be given by the coulomb friction coefficient μ_S and the contact normal pressure amplitude F_N. The tangential stiffness coefficient k_S, damping coefficient C_S, and tangential friction coefficient μ_S adopt empirical coefficient.

B. Omni-direction gait generation

To simplify the gait planning using robot symmetry, the leg speed command needs to be given according to the leg's spatial position relative to the forward direction. First, leg spatial position should be determined, as shown in Fig. 3(b)-1, establishing the forward direction coordinate system, the coordinate origin is the robot base, the x-axis is parallel to the forward direction at the current time, and the z-axis is vertically upward to establish the right-hand coordinate system. By calculating the leg direction vector $\overrightarrow{r_l} = (\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k})$ in this coordinate system, the spatial position can be determined. Then, the vector value is used to calculate the control quantity $v_i = v(\vec{r}_i)$, where v_i is the motion command of leg$_i$, that is the movement speed of the telescopic leg.

The design of the leg control function v_i needs to meet the ability of omnidirectional movement, joint motion amplitude constraint and a simple form to real-time calculate. In short, our method is as shown in planar front view Fig. 3(b)-3, taking the robot base as the center, the front landing leg extends forward in the forward direction, and the rear landing leg also extends to support the robot rolling. The other feet should be shortened to adjust the mass distribution, but the precise range needs to be defined by the following critical state.

Critical state: In the Fig. 3(b)-4 and 5, l_0 is the shortest length of the leg, and $l_{\max}$ is the largest length, taking $1.97125 l_0$. α and β are the critical state angles formed by the landing legs and the horizontal plane.

The first critical state determines when the front legs begin to shorten, as shown in Fig. 3(b)-4. At this moment, suppose that we have three landing legs, two legs are in the shortest state, vertical to the ground, and the most front leg is in the longest state. The angle between the longest leg and the horizontal plane is the boundary of contacting the ground. The front leg should start shrinking at this angle to smooth the movement. The smaller angle α will cause unstable rolling movement, and the bigger angle α will cause movement obstruction. Due to the symmetry, the rear legs should also stop further elongation at the angle α. The exact value of α is determined in the experiment.

The second critical state determines when the rear legs begin to extend, as shown in Fig. 3(b)-5. When the three landing legs are in the shortest length, the angle between the plane formed by the rear two legs and the horizontal plane is the boundary angle β, which holds the span of $[60°, 90°]$. In prototype experiments, β will influence the accuracy of trajectory. We test the appropriate value through experiments.

Except for these above leg spaces, the control command of other legs belongs to this law: the front legs extend, and the rear legs shrink, resulting in the efficient centroid configuration. The whole control laws are as follows:

$$\begin{cases} v_i = -v & k < 0, i^2 + j^2 < \cos^2(\alpha), i > -\cos(\beta) \\ v_i = v & k < 0, i^2 + j^2 < \cos^2(\alpha), i \leq -\cos(\beta) \\ v_i = -v & \{k < 0, i^2 + j^2 \geq \cos^2(\alpha), i < 0\} \cup \{k > 0, i < 0\} \\ v_i = v & \{k < 0, i^2 + j^2 \geq \cos^2(\alpha), i > 0\} \cup \{k > 0, i > 0\} \end{cases} \tag{11}$$

Finally, a relatively simple form v_i can be obtained. The direction vector space of the legs is divided into four regions as shown in Fig. 3(b)-3, where v is the maximum leg speed, and the extension is taken as the positive direction. In the prototype experiment, the unit direction vector of each leg in the body-fixed coordinate system is certain, and the IMU sensor can provide the base posture. According to the robot's attitude and forward direction information, the control vector of each leg can be obtained. Then the microcontroller determines the control command of each leg and sends the signal to the actuator.

C. Trajectory tracking method

When tracking with a desired path, the traditional robot needs to consider the limits of the turning radius. However, the radial skeleton robot is not affected by the posture of the robot itself. Experiments show that the radial skeleton robot can turn in situ, change the direction of motion and

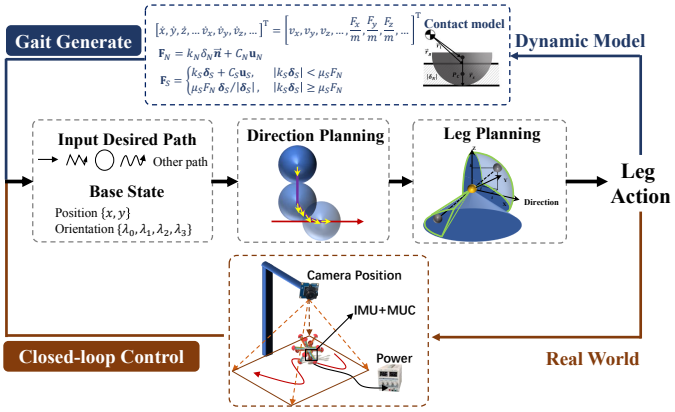

Fig. 4: Locomotion framework

turn simultaneously when the forward direction command changes. Therefore, particularly designed turning and other actions do not need in our method. Our method adopts a simple scheme to realize the direction planning. As shown in Fig. 3(c), when the distance between the robot and the desired path exceeds the robot radius, the planning direction is vertical to the desired path. When the distance is less than the robot radius, the planning direction points to the intersection of the circle and the desired path. This planning method fully uses the omnidirectional motion ability, which can quickly return to the desired path in distance and realize the real-time deviation correction in near, with a relatively simple form and calculation.

D. Locomotion framework

We use the above three parts to realize two types of movement, gait generates in simulation for open-loop deployment and movement control for closed-loop deployment in the real world, as seen in Fig. 4.

First, we could use the dynamic model to generate the gait sequence in the simulation and deploy it into open-loop experiments. The follow-up experiments proved that, compared with the other methods, our method could predict the slipping of feet on the ground and find the new gait that cannot be predicted by other methods, which makes the high accuracy of open-loop gait deployment.

Second, we proposed the closed-loop motion control framework for real radial skeleton robots. The state estimation is achieved by onboard and external sensors. In detail, we use an external camera to get the position of the robot base and the orientation of the base will be measured by IMU. Additionally, the control command of the motor is telescopic speed. Through real-time planning and deviation correction, we can leverage the structural advantages of the deformation robot. The details will be carried out in the experiment part.

IV. Experiments

This section sets up the accuracy comparison experiments of the open-loop method with other algorithms and the tracking accuracy and robustness test of our closed-loop method.

A. Open-loop gait deployment experiments

In this section, The compared methods include discrete manual-designed gait generation algorithm [13], inverse kinematics gait generation algorithm [14]. The physics-driven gait planning method [19] also provide a generation algorithm but can't be applied to the real robot through our pre-test. We compared the error and standard deviation of the centroid trajectory in three trajectories.

Open-loop trajectory tracking test: Results are shown in Fig. 5. We deployed the generated gait by the three methods in the straight line, S-shaped trajectory, and O-shaped trajectory. Each method is repeated ten times under each scenario, and calculating the mean and standard deviation, as shown in Fig. 5(a). In all experiments, our method achieves the minimum error and maintains the minimum standard deviation, which indicates that our method is highly accurate and stable. In the straight-line experiment, the inverse kinematics method gradually deviates from the benchmark, and the discrete gait presents a typical zigzag trajectory because it's based on the support triangle. Our method completes a nearly linear trajectory. In the S-shaped trajectory, the inverse kinematics method presents an actual trajectory far beyond the established trajectory. The tracking error of discrete gait is large when turning radius. Our approach can still track the track best. In the circle trajectory, the inverse kinematics can only track a quarter circle, so the error cannot be calculated. Our method can still achieve the highest accuracy and stability. This experiment validates that our simulation model can accurately predict the occurrence of sliding.

B. Closed-loop gait control experiments

To verify the accuracy improvement compared with the formal gait method, we tested it with four types of trajectory, and each trajectory test for ten times.

Closed-loop trajectory tracking test: We compared the closed-loop method with open-loop gait generation method. The results are shown in Fig. 6. In the straight-line experiment, the accuracy of the open-loop method is slightly higher than the closed-loop method. In all other experiments, the

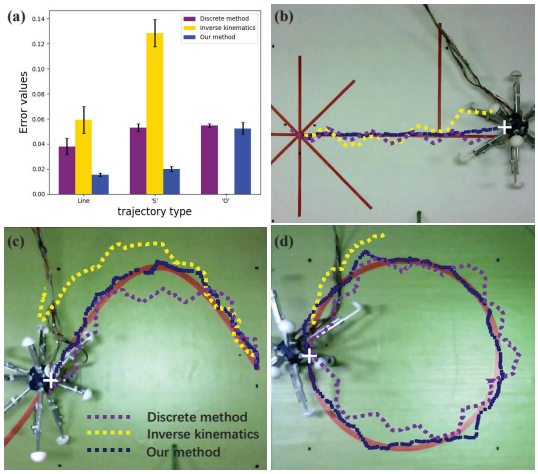

Fig. 5: (a) Open-loop trajectory tracking errors (b-d) Trajectories comparison of a single experiment

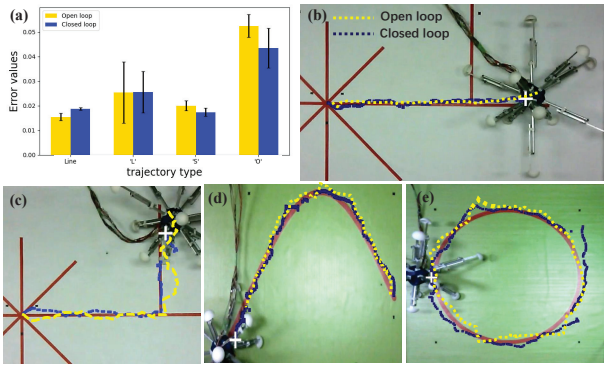

Fig. 6: (a) Closed-loop trajectory tracking errors (b-e) Trajectories comparison of a single experiment

accuracy of the closed-loop method is higher than the open-loop method. This shows that in simple tracking tasks, the gait switching is smooth, so the simulation model can achieve good results and accurately pre-adjust the deviation, even better than the closed-loop control. But in more complicated tracking tasks, errors are more likely to occur, and the closed-loop scheme can be re-planned in real time to bridge the errors, so the accuracy of open-loop gait is lower than that of the closed-loop algorithm.

C. Robustness experiments

To verify the robustness, we conducted repeated experiments in various interference environments and unseen obstacles.

Complex terrain: We set up two experiments to test the robustness, as seen in Fig. 7(a)(b). First, we built a complex obstacle terrain using wood boards, whose maximum undulation can reach 50% of the robot's structural size. Moreover, the robot can't perceive complex terrain and completely relies on the symmetrical multi-legged structure with no overturning and a high tolerance for terrain adaptation. In 20 times experiments, a 70% success rate can be achieved.

External interference: We manually intervened the robot on the predetermined linear motion to make it deviate from the trajectory greatly. The progress is shown in Fig. 7(c). When the distance from the desired trajectory exceeds a certain value, it will be judged as deviation. In all 20 times experiments, it can recover the originally established trajectories autonomously, and the recovery time is determined by the movement speed and the original deviation position.

Outdoor unknown environment: We used the untethered robot prototype and placed it in the rocky ground and dusty complex terrain for open-loop tests, as seen in Fig. 7(d)(e). The size of the rocks ranges from 0.5 to 2 times the radius of the robot foot-shell and the dusty terrain includes slopes, contact surfaces with variable friction and hardness, debris, and undulating obstacle. In both tests, the robot can achieve a stable landing gait and periodic locomotion mode.

V. Conclusion

This paper presents a new radial skeleton robot. The high telescopic ratio leg and symmetrical structure give the robot the advantage of no overturning and omnidirectional movement. A high-precision gait generation algorithm is proposed, which can be applied to any multi-legged radial skeleton robot, and a closed-loop locomotion frame is first proposed so that it can track paths with high accuracy, stability, and robustness. Finally, a prototype is made, and our method is compared with other algorithms to verify the advantages of the proposed algorithm. The robot can be used to explore unknown complex terrain and rugged surfaces. However, it also has some limitations worthy of future research. These include but are not limited to the following aspects:

(1) More body-sensing devices (force sensors, cameras, laser) can be installed to avoid relying on external perception. For example, the force sensors could be mounted in the leg-end to calculate the real impact force.

(2) A data-driven control strategy can be established to achieve autonomous movement, such as the neural network-based control policy for terrain adaptation.

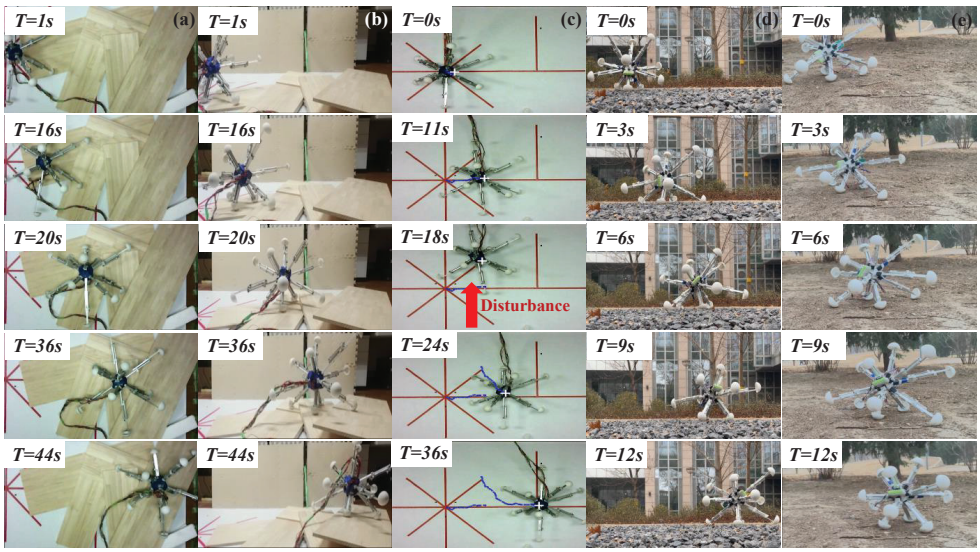

Fig. 7: (a-b) Obstacle terrain experiments (c) Interference experiments (d) Outdoor rocky environment untethered experiments (e) Outdoor dusty complex terrain untethered experiments

References

[1] Z. Li, X. Cheng, X. B. Peng, P. Abbeel, S. Levine, G. Berseth, and K. Sreenath, "Reinforcement learning for robust parameterized locomotion control of bipedal robots," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 2811–2817.

[2] J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning quadrupedal locomotion over challenging terrain," *Science robotics*, vol. 5, no. 47, p. eabc5986, 2020.

[3] C. Yang, K. Yuan, Q. Zhu, W. Yu, and Z. Li, "Multi-expert learning of adaptive legged locomotion," *Science Robotics*, vol. 5, no. 49, p. eabb2174, 2020.

[4] X. B. Peng, E. Coumans, T. Zhang, T.-W. Lee, J. Tan, and S. Levine, "Learning agile robotic locomotion skills by imitating animals," *arXiv preprint arXiv:2004.00784*, 2020.

[5] X. B. Peng, M. Andrychowicz, W. Zaremba, and P. Abbeel, "Sim-to-real transfer of robotic control with dynamics randomization," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 3803–3810.

[6] S. Dubowsky, S. Kesner, J.-S. Plante, and P. Boston, "Hopping mobility concept for search and rescue robots," *Industrial Robot: An International Journal*, vol. 35, no. 3, pp. 238–245, 2008.

[7] J. Antol, "A new vehicle for planetary surface exploration-the mars tumbleweed rover," in *1st Space Exploration Conference: Continuing the Voyage of Discovery*, 2005, p. 2520.

[8] M. W. Tilden, "Biomorphic robots as a persistent means for removing explosive mines," Los Alamos National Lab.(LANL), Los Alamos, NM (United States), Tech. Rep., 1995.

[9] T. Takemori, M. Tanaka, and F. Matsuno, "Gait design for a snake robot by connecting curve segments and experimental demonstration," *IEEE Transactions on Robotics*, vol. 34, no. 5, pp. 1384–1391, 2018.

[10] G. Xian and F. Yongchun, "Locomotion gait control for bionic robots: A review of reinforcement learning methods," *CAAI transactions on intelligent systems*, vol. 15, no. 01, pp. 152–159, 2020.

[11] J. L. Wilson, A. P. Mazzoleni, F. R. DeJarnette, J. Antol, G. A. Hajos, and C. V. Strickland, "Design, analysis and testing of mars tumbleweed rover concepts," *Journal of spacecraft and rockets*, vol. 45, no. 2, pp. 370–382, 2008.

[12] V. Gheorghe, N. Alexandrescu, D. Duminica, and L. A. Cartal, "Rolling robot with radial extending legs," in *2010 3rd International Symposium on Resilient Control Systems*. IEEE, 2010, pp. 107–112.

[13] B. Wagenknecht and D. Apostolopoulos, "Locomotion strategies and mobility characterization of a spherical multi-legged robot," in *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, vol. 44106, 2010, pp. 1049–1058.

[14] H. Nozaki, Y. Kujirai, R. Niiyama, Y. Kawahara, T. Yonezawa, and J. Nakazawa, "Continuous shape changing locomotion of 32-legged spherical robot," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 2721–2726.

[15] H. Nozaki, R. Niiyama, T. Yonezawa, and J. Nakazawa, "Shape changing locomotion by spiny multipedal robot," in *2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, 2017, pp. 2162–2166.

[16] L. A. Mateos, "Bionic sea urchin robot with foldable telescopic actuator," in *2020 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*. IEEE, 2020, pp. 1063–1068.

[17] R. A. Luders, D. Apostolopoulos, and D. Wettergreen, "Control strategies for a multi-legged hopping robot," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2008, pp. 1519–1524.

[18] T. Paschal, M. A. Bell, J. Sperry, S. Sieniewicz, R. J. Wood, and J. C. Weaver, "Design, fabrication, and characterization of an untethered amphibious sea urchin-inspired robot," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 3348–3354, 2019.

[19] F. Zhang and Y. Yu, "Physics-driven locomotion planning method for multilegged robots."

[20] D. Yang, S. Yang, Y. Yu, and Q. Wang, "Design and experiment of complex terrain adaptive robot based on deep re-inforcement learning," *Journal of Astronautics*, vol. 43, pp. 18–27, 2022.

[21] C. Crowe, M. Sommerfeld, Y. Tsuji *et al.*, *Multiphase Flows with*. Ž, 1998.