Proceedings Paper:

Lian, Yunlong, Wang, Tianyuan, Ingham, Joe et al. (2 more authors) (2023) CPG-based locomotion control of a quadruped robot with an active spine. In: Iida, Fumiya, Abdulali, Arsen and Wang, Mingfeng, (eds.) Towards Autonomous Robotic Systems (TAROS) 2023 Conference. 24th Towards Autonomous Robotic Systems Conference (TAROS), 13-15 Sep 2023 Springer , GBR .

CPG-based locomotion control of a quadruped robot with an active spine*

Yunlong Lian1[0000−0003−4230−1402], Tianyuan Wang1[0000−0003−3545−4937], Joe Ingham1[0009−0009−9725−4827], Mark A Post1[0000−0002−1925−7039], and Andy Tyrrell1[0000−0002−8533−2404]

School of Physics, Engineering and Technology
University of York, York, YO105DD, UK
`yunlong.lian@york.ac.uk`

Abstract. Central pattern generators (CPGs) are neural networks responsible for producing rhythmic behaviours and are commonly found in both vertebrate and invertebrate animals. This paper proposes a novel internal feedback mechanism for a CPG model designed to generate leg-spine coordinated locomotion in a quadruped robot with an active spine. This mechanism enables the CPG to independently control the frequency and amplitude of the stance and swing durations while also modifying the definition of stance and swing phases to generate more gaits. The CPG model's results are demonstrated on a simulated "tensegrity quadruped robot" called TQbot, which features a flexible spine with 3 degrees of freedom (DOF). By adjusting the parameters, the CPG model can generate gaits with leg-spine coordination and uses the spine for turning.

Keywords: Quadruped robot · CPG · Tensegrity spine · Spine control.

1 Introduction

In nature, a supple spine plays a crucial role in maintaining gait stability and facilitating agility for quadrupedal animals. Utilising spinal motion to generate equally agile gaits for quadruped robots has always been an exciting and challenging goal. Many research institutes and commercial organizations have developed state-of-the-art quadruped robots to achieve animal-like performance. However, due to the development cost, design and control difficulties, most quadruped robots use a rigid body [10,19,20,13,17,1,23].

Robotic spine studies are often found in crawling[11], snake[14], salamander[12] and modular robots[2]. A salamander robot with an active spine is able to perform a walking gait on the ground and swimming motion in water. The locomotion that synchronises the spine and legs is determined by coupled phase oscillators[4]. Another biologically inspired sprawling posture robot can perform agile turning behaviours with its active spine[8]. Their controller also achieves

---

* Supported by a University Research Priming Funding Grant from the University of York.

precise control and synchronises leg and spine motion by solving the inverse kinematics of the spine. Other studies investigated the influence of the spine on quadruped robots. MIT Cheetah explored the flexible spine impact on energy efficiency and high-speed running[21]. The differentially actuated spine can be actuated in the sagittal plane when the rear legs are in-phase. Serval is a quadruped robot with an active spine that has 3 DOF[5]. It was developed for utilising its spine to perform agile turning behaviours that are achieved by simply adding an offset to the spinal joint. Another study built a rat-like robot with a soft active spine to explore the effect of spine motion in the sagittal plane on the speed[9]. The results have shown a lateral flexion of an actuated spine benefit on robot's velocity.

On the other hand, the spine function of different robots depends significantly on the spine structures that require it to be specially designed. The incorporation of additional actuators at the waist of a quadruped robot with a spine can lead to an increase in overall weight. Thus, it poses a challenge to design a spine with 3 DOF while minimising weight gain. Furthermore, special mechanism design causes the control methods to be robot-specific. In the case of a rigidly connected spine, its motion can be effectively generated by solving inverse kinematics. However, employing this method to manipulate a flexible spine precisely would be difficult. Furthermore, the coordination of leg and spine movements poses an additional challenge in generating efficient and dynamic locomotion for quadruped robots with a spine. The survey revealed three main methods used to control the spinal motion of quadruped robots: modelling the robot and defining desired spine movement in Cartesian space, then using robot kinematics to fit it[24,7]; using a parameterised wave function to generate spinal motion but the cooperation of leg and spine is a problem[22]; Combining a bio-inspired method, CPG, with inverse kinematics can achieve precise control of the spine and coordinate it with legs, but this method is limited to control one of the orientations of the spine[8].

A quadruped robot with an actuated spine has the potential to improve the stability and speed of the whole body during movement[16]. Thus, a quadruped robot with an active spine to investigate leg-spine coordination and demonstrate in simulation is the core of this paper. Its spine joint mechanism is described in the next section. Section 3 introduces the modified CPG model in detail; in section 4, the gaits generation results of the CPG in a simulation will be demonstrated. Section 5 completes the paper with conclusions and suggestions for future work.

2 TQbot structure

The quadruped robot used in this paper is called TQbot, which stands for a tensegrity quadruped robot, shown in Fig.1(a). The robot with a flexible spine and has 15 DOF, with 3 DOF for each leg and the spine.

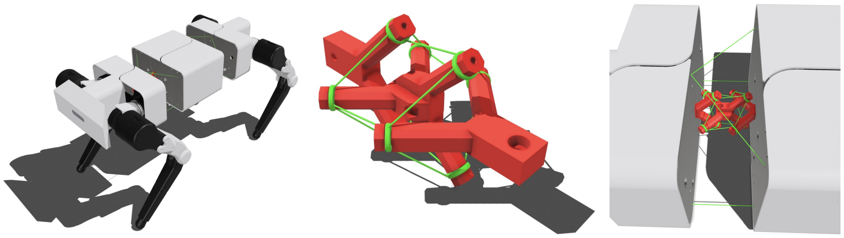

(a) TQbot model (b) Tensegrity spine (c) Spine connection

Fig. 1. TQbot structure: The robot body comprises three parts shown in (a). Two tensegrity spines are passive components that connect neighbouring body segments (b). Four braided wires around the passive spine are used to drive the spinal joints (c).

2.1 Spinal joint

Existing studies on quadruped robots with a spine typically have one or two actuated DOF in the lumbar region, and a few have 3 DOF[9,8,5]. Others research passive elastic soft spine to improve its energy efficiency while the weight of the body also increases[21]. However, the tensegrity structure as a robot spine has the following advantages compared with existing rigid actuated spines:

(1) Its independent three degrees of rotational freedom can be simultaneously actuated. The overall workspace of the robot is thus extensively increased to realise more advanced and agile manoeuvres.
(2) Tensegrity structures have higher structural efficiency compared with rigid structures, which allows a greater payload ratio of the robot.
(3) The inherent compliance of the tensegrity spine improves the robustness of the robot for situations such as external impacts and hazards.
(4) The flexibility of the tensegrity spine can potentially improve the energy efficiency of the robot and deal with external perturbation.

The spine structure has passive and active parts. The passive part is two identical tensegrity spines that only provide a connection between body segments, shown in Fig.1(b). Each is composed of one tensegrity vertebra and two half-vertebrae as the rigid elements to imitate its biological counterparts. They are continuously pretensioned in a passive manner for structural integrity and to constrain the translational displacement to a certain extent. To actuate three degrees of rotational freedom allowed by the tensegrity spine in roll, pitch and yaw orientation, the active part adopts the completely constrained parallel manipulator (CPRM) configuration. It imitates the muscles around the vertebrate to drive spinal joints to generate spinal motions. The CPRM configuration consists of eight 2mm diameter braided polyethene lines with a load capacity of 250 kg. Adjacent body segments are connected by four wires, shown in Fig.1(c). Four motors are placed in the middle body segment, and each controls two wires simultaneously and is connected to the front and rear body, respectively. Such configuration allows fewer actuators to actuate three DOF but also results in a symmetrical motion of the two spines in the time and spatial domain.

3 Central pattern generators

As a neural circuit widely found in vertebrate and invertebrate animals, the CPG has a natural advantage in coordinating and synchronizing the motion of the spine and legs. A modified CPG model with an internal feedback mechanism is proposed to generate diverse locomotion for a quadruped robot with an actuated spine. The improved CPG model is capable of independently controlling the stance and swing phases of a gait. Moreover, adjusting a specific parameter can modify the definition of stance and swing phases to produce more behaviours. The mathematical equations utilised in the CPG will be elaborated, followed by a presentation of the CPG topology structure and reference oscillator.

3.1 The modified phase oscillator and internal feedback mechanism

The modified oscillators that composed the CPG model are based on [4], which is a set of differential equations. Due to the special mechanical design, the robot has two symmetric spine joints, which can only simultaneously be in a symmetrical orientation. Thus an oscillator is used to drive one orientation of the spine joint, a total of three oscillators controlling the spine of the robot, which are responsible for controlling the rotation in transverse, sagittal and coronal planes separately. The equations of the modified phase oscillators used for all joints are given in equations 1-7:

$$\dot{\phi} = 2\pi * f(v_i) + \sum_j (\omega_{ij} sin(\phi_j - \phi_i - \varphi_{ij})) \tag{1}$$

$$\dot{r}_i^{st} = a_r(R_i^{st} - r_i^{st}) \tag{2}$$

$$\dot{r}_i^{sw} = a_r(R_i^{sw} - r_i^{sw}) \tag{3}$$

$$\dot{x}_i = a_x(X_i - x_i) \tag{4}$$

$$\theta_i = x_i + f(r_i) cos(\phi_i) \tag{5}$$

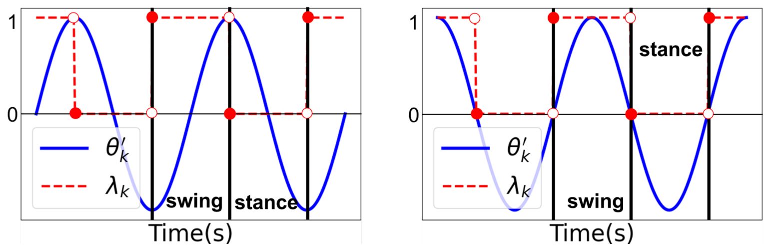
with

$$f(r_i) = \lambda_i(r_i^{st} + r_i^{sw}) + r_i^{sw} \tag{6}$$

$$f(v_i) = \lambda_i(v_i^{st} + v_i^{sw}) + v_i^{sw} \tag{7}$$

where ϕ_i and v_i are the phase and frequency of the *i-th* oscillator separately. The parameters w_{ij} are the coupling weights that are zero if oscillators i and j do not have a connection while φ_{ij} represents the phase difference between oscillator i and j. θ_i is the set-point generated by the oscillator that can be regarded as an angle, torque or angular of a joint. It varies with time, and the unit can be determined according to the actual application. In this study, θ_i represents a joint angle, and a PD controller is used to follow it. X_i and x_i are target and current offset of *i-th* oscillator. a_r and a_x are constant positive gains that represent the convergence speed of amplitude and offset.

However, the amplitude and frequency of some quadruped gaits vary in the stance and swing phases. The independent control of the amplitude and frequency allows for the potential generation of more dynamic and stable gaits. To achieve this, equations 2 and 3 are used to generating target amplitude for the *i-th* oscillator in the stance and swing phase. R_i^{st} and R_i^{sw} control the desired stance and swing amplitudes, and the current amplitudes are r^{st} and r^{sw}. The stance and swing frequency, v^{st} and v^{sw}, will be directly input and each set to the same in the same gait. For the automatic selection of parameters corresponding to the phase in gaits, an internal feedback mechanism is proposed in the CPG model (as shown in Fig.3). The equation 6-7 utilises the mechanism to dictate the amplitude and frequency in each leg's stance and swing phases. λ_i is an internal feedback signal obtained from the *k-th* hip joint oscillator ($k \in (8, 9, 10, 11)$). It is used to determine the stance and swing phases for the shoulder and knee joint oscillators of the same leg. Fig.2 shows the value of λ_k varies with the *k-th* hip joint phase and how μ_k's value to affect the definition of stance and swing phase.



(a) phase offset $\mu_k = 0$ (b) phase offset $\mu_k = \frac{\pi}{2}$

Fig. 2. Internal feedback signal: The dashed red line is the signal (λ_k), and the blue line is the trajectory of θ'_k. In (a), The definition of swing phase is defined as a rising edge of the blue line, while the stance phase is the falling edge. In (b), the swing phase corresponds to θ'_k value greater than 0, and the stance phase is the opposite. The red solid point indicates that the λ_k is defined, while the hollow point is not defined.

The λ_k is calculated using equations 8-9:

$$\lambda_k = \begin{cases} \frac{|\dot{\theta}'_k| - \dot{\theta}'_k}{2|\dot{\theta}'_k|}, & \dot{\theta}'_k \neq 0 \\ 1, & \dot{\theta}'_k = 0, \ddot{\theta}'_k \leq 0 \\ 0, & \dot{\theta}'_k = 0, \ddot{\theta}'_k > 0 \end{cases} \tag{8}$$

with

$$\theta'_k = cos(\phi_k + \mu_k) \tag{9}$$

Equation (8) is a linear piecewise function that only has two values: 0 and 1, which represent the swing and stance phases respectively. According to the definition of hip joint angle, the stance phase is a falling edge from the highest point to the lowest point during movement, while the swing phase is the opposite. Hence, determining whether the leg is in the swing or stance phase can be achieved by calculating the first and second-order differential of (5). Moreover, the stance and swing phase definition should be changeable, considering those are different in some behaviours, for instance, walking on the spot. To achieve this, equation (9) is introduced, which takes the *k-th* hip joint phase, ϕ_k, and a phase offset, μ_k, as the input. The θ'_k has the same phase as the *k-th* oscillator and the parameter μ_k is used to change the definition of stance and swing phase in gaits.

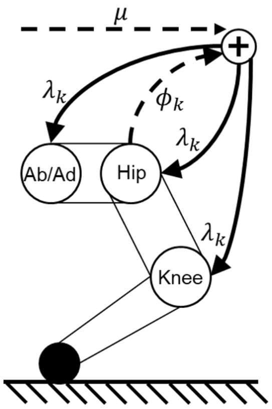


Fig. 3. Internal feedback mechanism in a leg: μ_k is an user input and ϕ is obtained from a hip joint. Output λ_k value to shoulder(Ab/Ad), hip and knee joint to independent control of swing and stance amplitude and frequency.

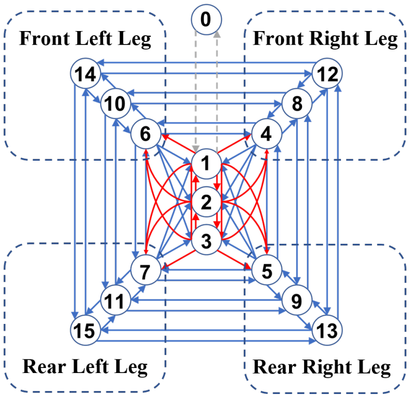


Fig. 4. Topology structure of the CPG: The red lines indicate spinal oscillators connected to other joints, and the blue lines are leg joint oscillators' connections. The arrow shows the connection direction between each oscillator.

3.2 CPG model structure

The CPG model can easily generate rhythmic patterns for robots. However, as robots have different structures and DOF, it is necessary to design a suitable topology for a particular robot to generate its desired pattern. Fish and salamander-shaped robots usually use chain CPGs[4,12,11,3], while quadrupedal robots usually use mesh structures[18,15,6]. The purpose of the CPG model is to synchronize the movement of the limbs with the spine for TQbot. To achieve this, the CPG consists of 15 oscillators, each of which corresponds to a single DOF. The topology of the CPG model is a network with a complicated coupling, shown in Fig.4. The oscillators of spinal joints are connected to each other and have a full connection with the oscillators of shoulder joints. The oscillators of the shoulder, hip and knee joints in one leg are connected to each other in turn, and each oscillator is interconnected with adjacent oscillators. Owing to the complexity of the CPG network, the coupling weights for spinal joints to shoulder joints are double those of the other joints to ensure fast convergence of the phase among the oscillators towards the desired value (i.e., $\omega_{ij} = 8$ where $i \in (1, 2, 3)$, otherwise, $\omega_{ij} = 4$.). This emphasizes the significant influence that the phase of the spinal joint exerts on the shoulder joint to a certain degree.

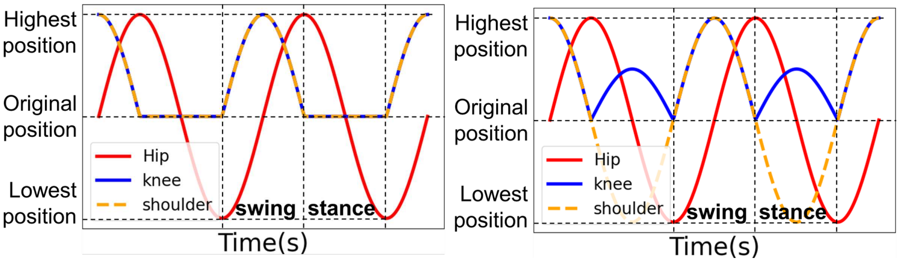
In Fig.4, 0 is a reference oscillator that does not correspond to any joints. 1-3 are spinal joint oscillators responsible for generating reference trajectories in roll, pitch, and yaw orientation. 4-7, 8-11, and 12-15 oscillators are shoulder joints, hip joints and knee joints respectively.

3.3 Reference Oscillator

For CPG models with a limited number of oscillators and simple connection relationships, the phase matrix utilises relative phase to determine the phase difference between oscillators. However, as the number of oscillators grows and the topology structure becomes increasingly complex, identifying the relative phase is challenging. For example, a fully connected CPG network comprising four oscillators necessitates at least six relative phases to derive a complete phase matrix. In our 15-oscillator CPG network, a minimum of 27 relative phases between oscillators must be determined due to the specified connection way.

Although it is possible to calculate the entire phase matrix from the relative phase of any one of the oscillators to the others, this approach leads to modifying the phase difference with all other oscillators when the phase of that oscillator needs to be altered. This undoubtedly complicates the process of adjusting the phase difference matrix. For instance, if the oscillator that controls the roll motion of the spine joint is employed as a reference oscillator, adjusting its phase in a specific gait would require changing the phases of all other oscillators. Consequently, using the relative phase to obtain the entire phase difference matrix can be troublesome for complex CPG models.

To resolve this issue, a reference oscillator is introduced to acquire the phase difference from others and calculate the relative phase between oscillators, thereby obtaining the entire phase matrix. The reference oscillator in the CPG is labelled as zero and is connected solely to the second oscillator, marked as one, that controls the roll orientation of the spinal joint(as shown in Fig 4). The reference oscillator does not output externally and maintains an unchanged phase.



(a) Joints outputs in walk, trot, pace gaits (b) Joints outputs in gallop gaits

Fig. 5. Single leg joints trajectories shaped by the internal feedback mechanism: phase and amplitude relationships in gaits.

4 Simulation

Currently, CPG is not the main gait generation method for quadruped robots because its ability to generate dynamic gaits is limited. One of the reasons for this is that the mechanism utilising feedback to shape locomotion is unknown in biology. The conventional CPG model with phase oscillators can only generate rhythmic but unnatural gaits for quadruped robots[4]. Other modified CPG models usually use hard-code programs to control the amplitude and frequency in the stance and swing phases, but it is difficult to generate different patterns[6]. However, the proposed internal feedback mechanism provides a way to shape more natural and various gaits that are difficult for other models. In this section, several gaits are analysed for tuning the parameters of the CPG to generate different frequencies and amplitude in the stance and swing phases. Especially trot in place gait, which other CPG models can hardly generate. The results of the diversity and effectiveness of gait patterns shaped by the internal feedback mechanism are verified by conducting experiments in a simulation called Isaac Sim[1].

4.1 Gaits shaped by the internal feedback mechanism

Basic gaits Following the gaits often observed in quadruped mammal walking videos, the phase difference between each joint can be determined. As illustrated in Fig 5(a), during the swing phase, the leg must be lifted off the ground to allow for effective forward motion. To achieve this, the knee and shoulder joints are contracted and lifted separately during this phase. Then as the hip joint swings back to its original position from the highest position, the shoulder and knee joints are raised to their highest points. At the start of the stance phase, the shoulder and knee joints return to their original positions, and the hip joints reach the lowest position. The knee and shoulder joints generally remain stationary during the stance phase to ensure the stability of the robot while in motion

---

[1] All experimental results can be found at https://youtu.be/NJrMatc6PwU

unless the swing of the hip joint is too large, in which case the knee and shoulder joints need to be extended until they hit the ground and then bent back into their original position. In the walk, trot and pace experiment, the swing of the hip joint was kept at small angles.

Fig 5(b) shows the leg joints phase in the gallop gait. The shoulder joint did not remain in position in the stance phase but continued to swing to obtain a greater stride length to increase speed. The phase of the knee joint is also slightly different from other gaits in the gallop gait due to the hip joint swinging at big angles. In order to maintain stability during the run, the knee joint in the stance phase will, the same as in the swing phase, contract slightly at first, reaching its highest position at the original position of the hip swing and then gradually extending back to its original position.

Trot in place gait Except for the four basic gaits mentioned above, the gait of the trot in place is another commonly used gait for quadruped robots. The swing phase is when the hip joint swings backwards from the original position to the lowest position and back to the original position, while the whole leg does not move in the stance phase, as shown in Fig.6. It can be easily generated by adjusting the phase offset ($\mu_k = \frac{\pi}{2}$) in the internal feedback mechanism.

The usage of spinal motion Typically, spine flexion at the coronal plane increases the length of the stride to improve speed, while flexion in the transverse plane helps with turning. In these experiments, all four basic gaits except the pace gait utilise the spine to varying extents. Walk and trot gaits incorporate the spine movements in the coronal and transverse planes. As the robot's front leg swings forward, the curvature of the spine in the coronal plane is toward the side of the front swinging forward leg. The movement of the spine in the transverse plane serves as a functional equivalent to the shoulder joint, lifting the leg during the swing phase. In the gallop gait, the spine bends downwards in the sagittal plane as the two front legs swing forward and upwards in the stance phase of the front legs, increasing the stride and speed of the robot. The results of the CPG output and gait transition of five basic gaits are shown in Fig.6.

4.2 Multi-direction movement in the trot gait

Move backwards In addition to implementing the basic gait for forward movement, a trot gait to enable the robot to move backwards has been developed. Notably, opposite to the forward gait, the hip joint in the backward gait enters the stance phase while swinging forward and the swing phase while swinging backwards. Thanks to the reference oscillator, the new phase difference matrix can be obtained by swapping the phase of the hip joints of the left and right legs. Other parameters and phase differences are the same as the forward trot gait parameters.

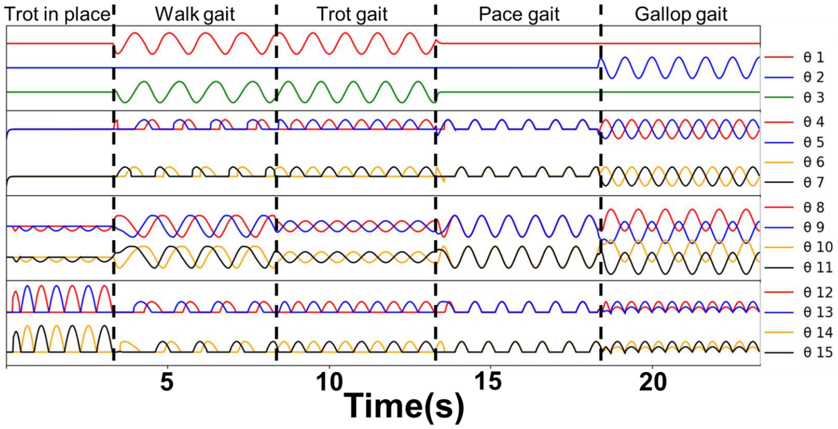
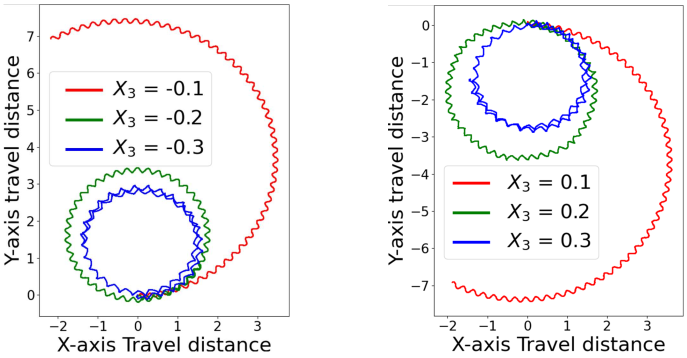


Fig. 6. Gait transitions: Achieved by replacing the parameter configurations of different gaits. In a real application, using a spline function can achieve a smooth transition. Changed at 3, 8, 13 and 18 seconds. Initial positions of all joints are 0. θ 1 to 3 are spinal joints corresponding to the movement in the transverse, sagittal and coronal planes. 4-7, 8-11, and 12-15 are shoulder, hip and knee joints that follow the RF-RR-LF-LR leg sequence.

Sideways movement Additionally, sideways movements are implemented based on the trot gait, which allows the robot to move laterally to the left or right. In this gait, the shoulder joints of the left and right legs are in phase, while that of the front and rear legs are in phase for half a cycle. In other words, the shoulder joints of the left front and right front legs are simultaneously extended to their highest position while the shoulder joints of the left rear and right rear legs are simultaneously contracted to their lowest position. It is important to note that the hip joint does not move during this gait; also, the stance/swing phase of shoulder and knee joints is different when moving left and right. For translating to the left, the shoulder joint of the left front leg is abduction in the swing phase, and that of the right rear leg is adduction simultaneously. Meanwhile, the knee joints of the left front leg and right rear leg rotate to the highest position, while that of the right front and left back legs are rotated to the lowest position. The shoulder joint is in the opposite phase when moving to the right, which can be achieved by swapping the phase difference between the right and left shoulder joints.

Turn in circles Regarding turning encircles gaits, the quadrupedal robots use the spine to make turns by simply giving it a bias in the coronal plane without modifying the trajectory of the leg joints. In this gait, the spine is not moved in the transverse and sagittal planes. The results in the simulation showed that the robot could generate efficient and stable turning gaits when the curvature of the spine in the coronal plane is within plus or minus 0.3 radians. Fig.7 shows the travel trajectories using the turning gait with a bending spine in the simulation.



(a) Turning left (b) Turning right

Fig. 7. Turning by bending the spinal joint in the coronal plane: The robot started to walk at (0,0) coordinates and moved for 50 seconds. X_3 is the offset of the spinal joint in the coronal plane, which is negative to turn left and positive to turn right.

5 Conclusion

This paper presents a modified CPG model with an internal feedback mechanism, which is capable of coordinating a quadruped robot's legs with a 3-DOF spine to generate more natural and dynamic gaits. Additionally, the spinal motion employed in the trot gait is beneficial for executing circular turns. A reference oscillator is incorporated to adjust the phase difference matrix based on the absolute phase difference, simplifying the modification of oscillator phases and enhancing interpretability.

However, generating gait patterns using CPG models requires prior knowledge of the gaits and then tuning parameters by hand, making it challenging for robots to generate dynamic and robust gaits on rough terrains. In conclusion, future research should focus on the following aspects: (1) conducting parametric learning to enable robots to explore gaits autonomously; (2) developing a reflex system that integrates sensory information to generate dynamic gaits on unstructured terrains; and (3) combining other learning methods to achieve self-balancing in various terrains.

References

1. BostonDynamics: Spot. Available from: https://www.bostondynamics.com/products/spot Accessed April 17th, 2023
2. CH, S.S.R., et al.: 2dxopod-a modular robot for mimicking locomotion in vertebrates. Journal of Intelligent & Robotic Systems **101**, 1–16 (2021)
3. Crespi, A., et al.: Controlling swimming and crawling in a fish robot using a central pattern generator. Autonomous Robots **25**(1), 3–13 (2008)
4. Crespi, A., et al.: Salamandra robotica ii: an amphibious robot to study salamander-like swimming and walking gaits. IEEE Transactions on Robotics **29**(2), 308–320 (2013)

5. Eckert, P., et al.: Towards rich motion skills with the lightweight quadruped robot serval-a design, control and experimental study. In: International Conference on Simulation of Adaptive Behavior. pp. 41–55. Springer (2018)
6. Gay, S., et al.: Learning robot gait stability using neural networks as sensory feedback function for central pattern generators. In: 2013 IEEE/RSJ international conference on intelligent robots and systems. pp. 194–201. Ieee (2013)
7. Hatton, R.L., Choset, H.: Generating gaits for snake robots: annealed chain fitting and keyframe wave extraction. Autonomous Robots **28**, 271–281 (2010)
8. Horvat, T., et al.: Spine controller for a sprawling posture robot. IEEE Robotics and Automation Letters **2**(2), 1195–1202 (2017)
9. Huang, Y., et al.: Enhanced quadruped locomotion of a rat robot based on the lateral flexion of a soft actuated spine. In: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 2622–2627. IEEE (2022)
10. Hutter, M., et al.: Anymal-a highly mobile and dynamic quadrupedal robot. In: 2016 IEEE/RSJ international conference on intelligent robots and systems (IROS). pp. 38–44. IEEE (2016)
11. Ijspeert, A.J., et al.: From swimming to walking with a salamander robot driven by a spinal cord model. science **315**(5817), 1416–1420 (2007)
12. Ijspeert, A.J., et al: Online trajectory generation in an amphibious snake robot using a lamprey-like central pattern generator model. In: Proceedings 2007 IEEE International Conference on Robotics and Automation. pp. 262–268. IEEE (2007)
13. Kenneally, G., et al.: Design principles for a family of direct-drive legged robots. IEEE Robotics and Automation Letters **1**(2), 900–907 (2016)
14. Liljeback, P., et al.: Snake robot locomotion in environments with obstacles. IEEE/ASME Transactions on Mechatronics **17**(6), 1158–1169 (2011)
15. Liu, C., et al.: Cpg driven locomotion control of quadruped robot. In: 2009 IEEE International Conference on Systems, Man and Cybernetics. pp. 2368–2373. IEEE (2009)
16. Pouya, S., et al.: Spinal joint compliance and actuation in a simulated bounding quadruped robot. Autonomous Robots **41**, 437–452 (2017)
17. Raibert, M., et al.: Bigdog, the rough-terrain quadruped robot. IFAC Proceedings Volumes **41**(2), 10822–10825 (2008)
18. Righetti, L., Ijspeert, A.J.: Pattern generators with sensory feedback for the control of quadruped locomotion. In: 2008 IEEE International Conference on Robotics and Automation. pp. 819–824. IEEE (2008)
19. Semini, C.: Hyq-design and development of a hydraulically actuated quadruped robot. Doctor of Philosophy (Ph. D.), University of Genoa, Italy (2010)
20. Semini, C., et al: Design of hyq–a hydraulically and electrically actuated quadruped robot. Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering **225**(6), 831–849 (2011)
21. Seok, S., et al: Design principles for highly efficient quadrupeds and implementation on the mit cheetah robot. In: 2013 IEEE International Conference on Robotics and Automation. pp. 3307–3312. IEEE (2013)
22. Tesch, M., et al: Parameterized and scripted gaits for modular snake robots. Advanced Robotics **23**(9), 1131–1158 (2009)
23. UnitreeRobotics: B1 robot. Available from: https://www.unitree.com/b1 Accessed April 17th, 2023
24. Yamada, H., Hirose, S.: Study on the 3d shape of active cord mechanism. In: Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006. pp. 2890–2895. IEEE (2006)