Design and Control of a Tortoise inspired quadruped robot using CPGs*

Shivang Gangadia[1] and Amey Rawool[1]

University of Bristol, Beacon House, Queens Rd, Bristol BS81QU, UK

Abstract. Central Pattern Generators are neural circuits in animals that are responsible for controlling rhythmic movement like locomotion even in absence of rhythmic input signals. In this work, we design and develop a tortoise inspired quadruped robot, which uses underactuated limbs in it's mechanical design and a Central Pattern Generator for control. It is able to exhibit gait variations like moving in straight line, moving while turning and rotating in place using only 2 input parameters from user: speed of motion and motion type. We also compare the robot-CPG combination with other works and show future potential for development of a complete artificial animal using Artificial Intelligence.

Keywords: Quadruped robot · CPG · Gait control · Bio-inspired robot

1 Introduction

In this work, we present the design of an underactuated mechanical structure and low dimensional control system for a tortoise inspired quadruped robot. Central pattern generators (CPGs) are neural circuits found in both invertebrate and vertebrate animals that can produce rhythmic patterns of neural activity without receiving rhythmic inputs. A number of properties of CPGs make them interested for applications in robots are listed in [3]. The combination of CPGs with adaptive morphology has been a major subject of research in robotics with many positive results in terms of improving and simplifying control logic ([3], [2], [8], [1]). CPGs for robots are presently implemented in software as a network of coupled oscillators[3]. There are many good examples of the use of CPGs for generating gait patterns for quadruped robots specially utilizing the sony AIBO robot platform ([5], [4], [7]), but not many focusing on tailoring the CPG and mechanical design to each other.

However, locomotion in lower center of gravity vertebrates like tortoises is different from bigger animals like dogs and horses in terms of principles used for propagation. While the latter make extensive use of the Spring Loaded Inverted Pendulum model for energy conservation, the former do not use it extensively[10] since 1 type of gait is sufficient for their size relative locomotion needs. Thus design of robots inspired from tortoises are fundamentally different. A good example of a tortoise inspired robot utilizing soft limbs for terrestrial locomotion

* Supported by University of Bristol

is presented in [6]. However, this example does not make use of CPGs to generate gait patterns. Another example of a tortoise inspired robot for cleaning of underwater silt is given in [9], but their design requires kinematic modelling to apply the gait pattern which is also not generated using CPGs.
The primary contributions of our work are:

1. Simpler design of a multi layer central pattern generator using coupled sinusoidal oscillators.
2. Mechanical design that eliminates the need for kinematic modelling by allowing direct application of generated signal to actuators

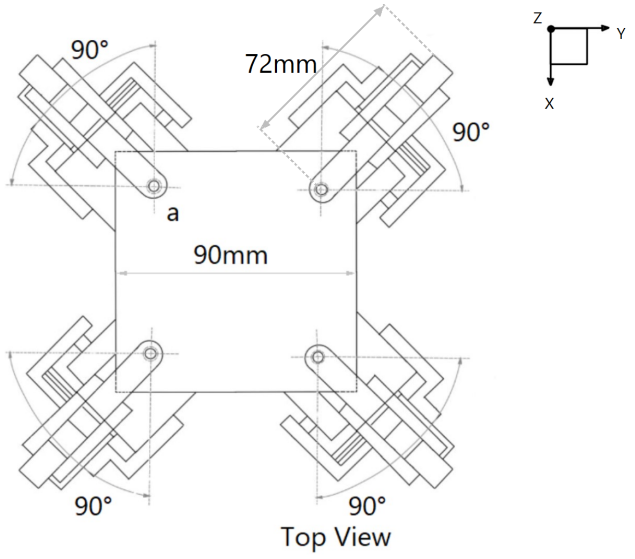
The following sections describe the mechanical design and open-loop CPG based control system design of our robot. Section 2 describes the physical and mechanical design of the robot. Section 3 describes the abstracted design of our control network and section 4 describes the mathematical equations governing the CPGs operations. Section 5 discusses the resultant motion exhibited by the robot while section 6 discusses future potential and applications of this work.

2 Mechanical Design of Robot

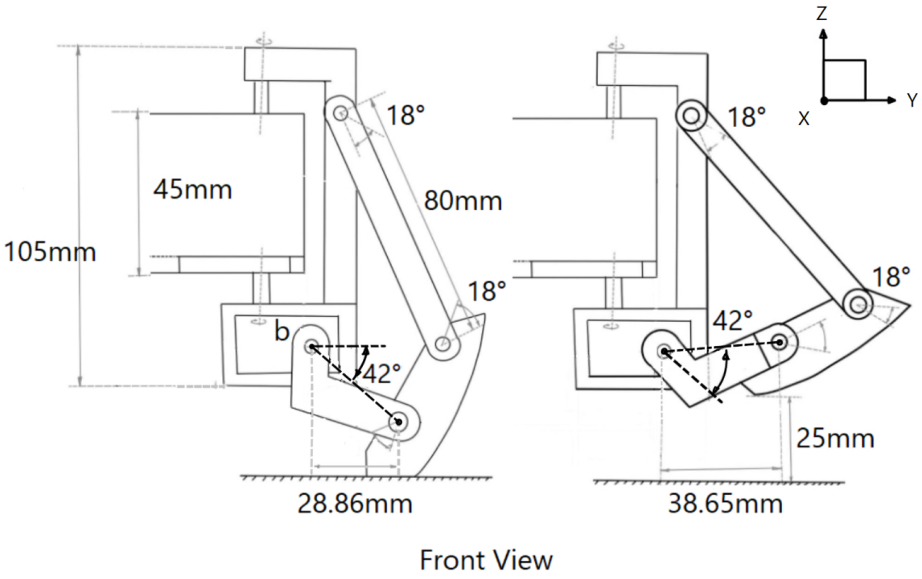
The robot is designed as a quadruped symmetrical about the X and Y axes. Each of it's 4 limbs are actuated by two servo motors for 3 degrees of motion. As shown in figure 1a, a marks the point of attachment of the first servo enabling a 90° range of rotary motion about the z-axis. As shown in figure 1b, b marks the point of attachment of the second servo enabling a 42° range of rotary motion about the local x-axis and $9.79mm$ of translatory motion along the local y-axis. Thus each limb has 2 independent and 1 constrained degrees of motion. The design of the limbs allows direct application of the CPG's output to the respective motors. This eliminates the need for kinematic modelling, thus reducing the dimensionality of the control space and reducing the number of computations required to get the desired control signal to the actuator. While the translation along the y-axis is not actively controlled, it can improve stability when the robot needs to keep stable when tilted in future design improvements. Presently, the robot walks parallel to surface with a low Center of gravity.

3 Design of the CPG

The CPG is responsible for generating gait patterns in the form of trajectories of the leg's end effectors, i.e., feet. Our CPG is designed to operate in two layers: first a central node that receives changes in state variables from a higher level controller and feeds the signals to the second layer. The second layer, consisting of pattern generators bound to each of the 4 limbs respectively. The oscillatory behaviour is achieved through use of two sinusoidal generators, each directly signalling the it's respective actuator. Here on, the 4 limbs are referred to by their positions left forward (LF), right forward (RF), left backward (LB) and



(a) Top View



(b) Side View

Fig. 1: Top to bottom, images in sequence showing the top view and front view of the robot and its respective dimensions with 'a' and 'b' being the points of attachment of servos.

right backward (RB). A graphical representation of the network is given in figure 2. It is to be noted that while the servos used for actuation internally utilize a closed loop control for maintaining position, our control system itself does not use any form of feedback from the limbs.

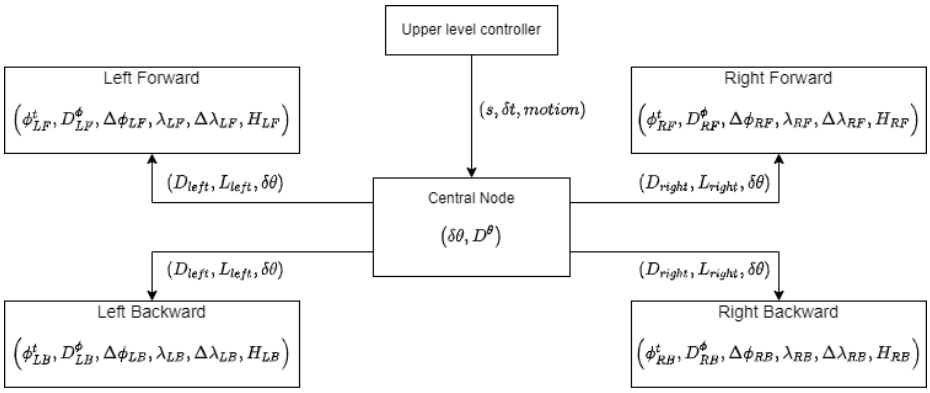


Fig. 2: The topology of the CPG network

4 CPG Operations

Note:

- All symbols used in the following sections are listed in table 1 for the reader's benefit.
- For all equations in this section, $i\epsilon\{LF, LB, RF, RB\}$ and $j\epsilon\{left, right\}$

Firstly, a higher level controller, such as a human operator or an Artificial Intelligence system expresses intended motion in the form of speed of motion and type of motion fed to the central control node. It is to be considered that the speed of motion is highly dependent on the capabilities of the actuators used (here, servo motors). For our robot which deploys off the shelf hobby servos, the speed variable (s) was an integer in the order of ones ranging from 4 to 10. Values below 4 resulted in a very low stability gait. The types of motion could be forward and backward straight line motion, forward and backward motion with turns (left and right) and rotation in place (clockwise or anticlockwise). The motion characteristic of direction (forward or backward) is reflected in the direction variable D^{θ} as:

- Forward $\rightarrow D^{\theta} = +1$
- Backward $\rightarrow D^{\theta} = -1$

Symbol	Description
δt	Time passed since last cycle of calculation
$\delta\theta$	Change in phase of the central control node as a result of δt
D^{θ}	Direction of phase change ($\delta\theta$) of the CPG network
L_{left}	Stride length of the left pair of limbs (part of central network)
L_{right}	Stride length of the right pair of limbs (part of central network)
D_{left}	Direction of phase change of the left side limbs (part of central network)
D_{right}	Direction of phase change of the right side limbs (part of central network)
ϕ_i^t	Calculated phase of the individual(i^{th}) limb at time step t
$\Delta\phi_i$	Phase offset of the individual(i^{th}) limb relative to the central network
D_i^{ϕ}	Inherent direction of oscillation of the individual(i^{th}) limb
λ_i	Stride length of the individual(i^{th}) limb
$\Delta\lambda_i$	Stride length modifier of the individual(i^{th}) limb
H_i	Stride height of the individual(i^{th}) limb

Table 1: CPG parameters and their symbols

The higher level controller must also provide time passed since last cycle of calculations (δt in table 1) to the central control node. The central control node then calculates the change in phase based on δt as follows:

$$\delta\theta = 2\pi \times s \times \delta t \times D^{\theta} \tag{1}$$

This change in phase ($\delta\theta$) along with the stride length(L) and direction of motion(D) is conveyed to the pattern generators of the individual limbs as shown in table 2 and figure 2. Using these along with the limb's inherent direction (D_i^{ϕ}) and the phase offset ($\Delta\phi_i$), the limb's pattern generator calculates it's updated phase as follows:

$$\phi_i^t = \phi_i^{t-1} + \Delta\phi_i + \left(\delta\theta \times D_i^{\phi} \times D_j\right) \tag{2}$$

The inherent direction of a limb (D_i^{ϕ}) selects the default direction of phase change of limbs between clockwise or anticlockwise. This is necessary, since otherwise all limbs will apply actuation force in the same direction resulting the overall

robot rotating in place.
The inherent phase offset ($\Delta\phi_i$) decides the gait mode for quadrupeds. Using this phase offset, an input of the same phase can be transformed into different effective phases for different limbs as shown in figure 3. Using a combination of these offsets($\Delta\phi_i$) and direction (D_i^{ϕ}), limb pairs LB & RF and LF & RB contribute to the same force vector. Note that the central node only feeds in a change in offset and each individual pattern generator keeps track of it's own phase.

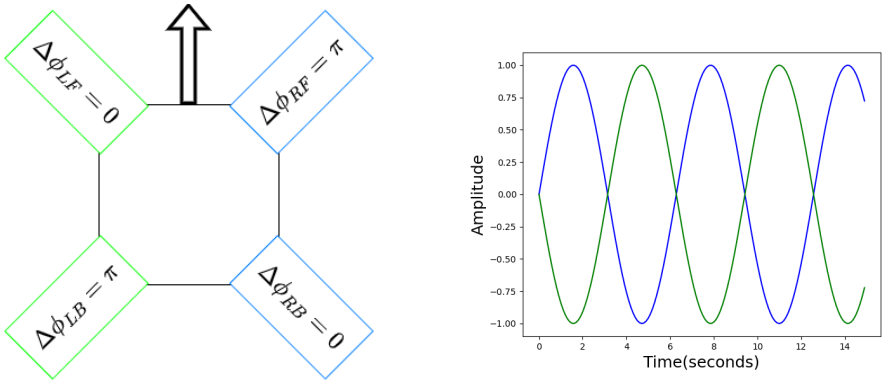


Fig. 3: Phase offsets for limbs (left) and it's effects (right). The arrow in the left image shows the forward direction for the robot.

The stride length of the individual pattern generators is calculated as follows:

$$\lambda_i = L_j + \Delta\lambda_i \tag{3}$$

Note that the offset ($\Delta\lambda_i$) is an intrinsic parameter which may be used to limit the motion of the legs for reasons such as physical constrains.
Finally, the output of the pattern generator are calculates using the phase oscillator equations as follows:

$$Z_i = \lambda_i \times cos\left(\phi_i^t\right) \tag{4}$$

$$X_i = H_i \times sin\left(\phi_i^t + \left(\frac{\pi}{4} \times m\right)\right) \tag{5}$$

The values of Z_i and X_i are in units of radians and are applied to servos 'a' and 'b' respectively. The range of motion in terms of actual distance depends on the dimensions of the different parts of the mechanical design as shown in figure 1a and 1b. λ_i influences the length of the step that each limb takes and H_i influences the height of the step, and D_i^{ϕ} influences it's direction as shown in figure 5b.

Parameter	Straight	Left Turn	Right Turn	Anticlockwise	Clockwise
Stride length	$L_{left} = L_{right}$	$L_{left} < L_{right}$	$L_{left} > L_{right}$	$L_{left} = L_{right}$	$L_{left} = L_{right}$
D_{left}	1	1	1	-1	1
D_{right}	-1	-1	-1	-1	1

Table 2: Parameter values as controlled and tracked by the Central node. These are then passed on to the individual limbs as shown in figure 2

5 Results

The gait follows the principle of a rimless wheel, which means that each diagonal pair of legs exhibit similar gaits and each linear pair (forward to backward) of side limbs exhibit gaits such that at any point in time, they form a rimless wheel in contact with the ground as shown in figure 5a. Variance of parameters as in table 2 results in end-effector trajectories as shown in figure 6.
Figure 8 shows the robot in action, executing walking in straight line, rotating in place and turning left while walking respectively. While other works also change the phase offset ($\Delta\phi_i$) to switch between gait modes like walking, trotting and galloping ([5], [4], [7]), our design keeps these values constant throughout since small tortoises, from where the design is inspired, do not exhibit such gaits. The straight line walking behavior exhibited walking speed slightly less than half the robot's body length per second (here, 39mm per second). This is directly proportional to the dimensions of the limbs and the capabilities of the actuators used. Physically, the speed is dependent on the stride length(λ_i), length of limb and the speed of actuation of the servos. For example, as shown in figure 1a, the length of limb from the z-motor attachment ('a') is $l = 72mm$. The range of motion of the limbs can be geometrically represented as shown in figure 4. Let $\lambda_i = \frac{\pi}{4}$, then as shown in the diagram (4), $\alpha = \frac{\pi}{8}$. Then, the values L_1 & L_2 can be calculated as:

$$L_1 = l \times sin\left(\frac{\pi}{4} - \alpha\right) \tag{6}$$

$$L_2 = l \times sin\left(\frac{\pi}{4} + \alpha\right) \tag{7}$$

$$\text{Effective Stride Length} = L_2 - L_1 \simeq 39mm \tag{8}$$

Since the gait trajectories follow the principle of a rimless wheel for the given set of phase offsets ($\Delta\phi_i$), the turning radius can be approximated by assuming each side pair of limbs as 1. Now, the difference in stride lengths (L_{left} & L_{right}) can be used to approximate the turning radius as shown in figure 7b. Assume taking a left turn. As shown in table 2, $L_{left} < L_{right}$. The term h_1 in the figure 7b refers to the distance between the two pair of legs. From figure 1a, this can be

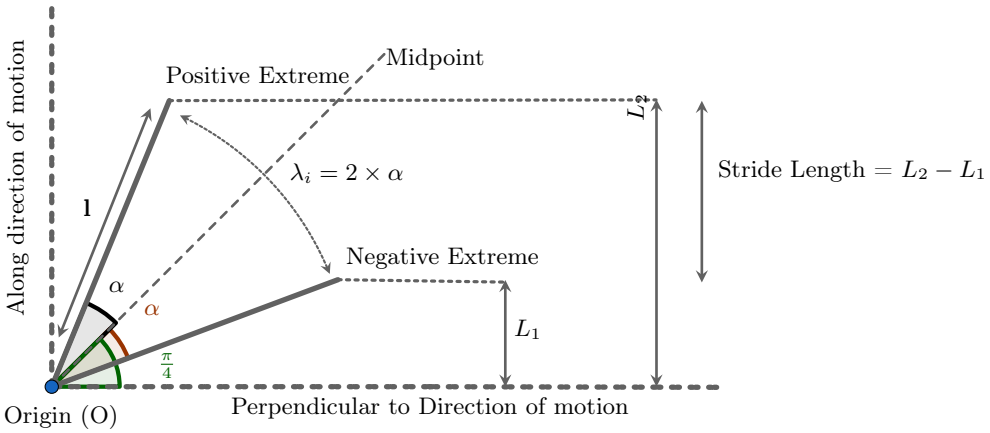
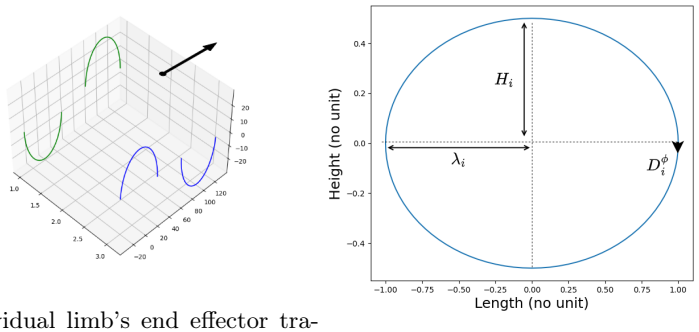


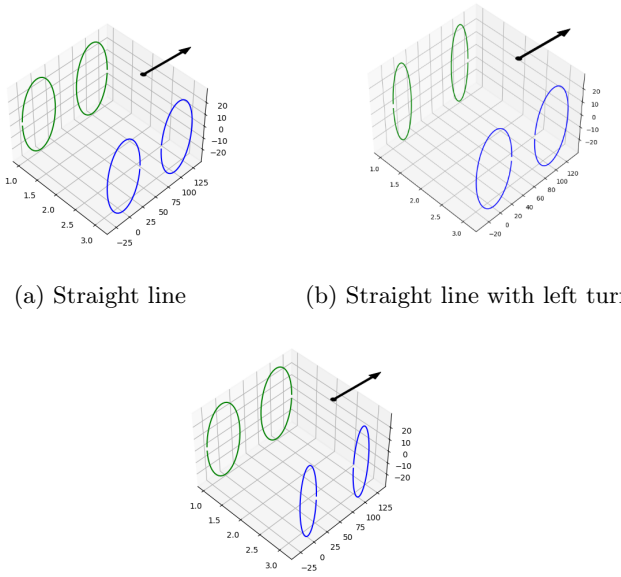
Fig. 4: Geometric representation of influence of λ_i on stride length



(a) Individual limb's end effector trajectories exhibiting the rimless wheel principle for legged robots

(b) Trajectory parameters and their effect on the trajectory's shape

Fig. 5: Trajectory realization



(a) Straight line

(b) Straight line with left turn

(c) Straight line with right turn

Fig. 6: End effector trajectories resulting from change in CPG parameter values

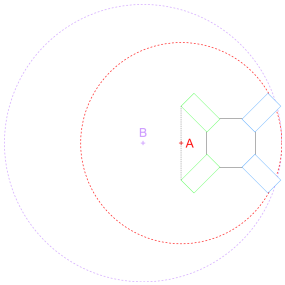
approximated to $90mm$, i.e. the width of the robot. The general approximation of the turning radius (R_{turn}) can be calculated as:

$$R_{turn} = \frac{L_{left} \times h_1}{L_{right} - L_{left}} + L_{right} \tag{9}$$

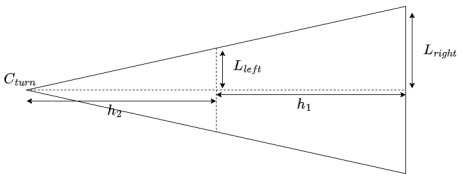
The shortest turning radius is achieved when L_{left} or L_{right} is 0 as shown by the center of turning point 'A' in figure 7a and center of turning for an arbitrary turning radius is marked by 'B'. While equation 9 does not give an exact value for the turning radius, it is useful to approximate it to $\pm 20mm$ as verified by experiments.

6 Conclusion and Future work

A quadruped robot was developed with underactuated legs and gait inspired by a tortoise and controlled using a Central Pattern Generator. The legs have 2 independent and 1 constrained degrees of freedom. It exhibits only 1 mode of gait (walking) but can shows all behaviours within that gait like translating forward, backward, turning while translating and rotating in place. The application of this CPG in software benefits a great deal from the mechanical design since no kinematic models need to be developed and the outputs from the CPG are



(a) Shortest turning point 'A' and arbitrary turning point 'B'



(b) Approximating the turning radius.

applied directly onto the legs of the quadruped through signals to two actuating servos per leg. This adds to the simplicity of the control system's design which presents an interface of only 2 control parameters, i.e. speed of motion (s) and motion type supplied either by a manual control system, or AI. This interface is similar to how organic brings neural systems work where a Central Pattern Generator control's it's gait while the brain only needs to expresses intent.
The next steps in this line of research would therefore be emulating a brain using an Artificial Intelligence system like deep reinforcement learning with inputs like images from a front facing onboard camera and an onboard inertial measurement unit, and agency over the robot's movement using the controllable variables previously mentioned (s, *motion*). This would mean the AI system only needs to learn where to go instead of having to learn limb control and gait from scratch. Such a robot could be deployed on exploratory missions with small form factor requirements or to study response of real tortoises to it's presence.

Acknowledgements We would like to thank the University of Bristol for giving us an opportunity during our term time to develop this idea and work on this project as part of our coursework and specially Dr. Sabine Hauert and Dr. Dandan Zhang for mentoring us throughout the process.

References

1. Corucci, F., Calisti, M., Hauser, H., Laschi, C.: Evolutionary discovery of self-stabilized dynamic gaits for a soft underwater legged robot. In: 2015 International Conference on Advanced Robotics (ICAR). pp. 337–344. https://doi.org/10.1109/ICAR.2015.7251477
2. Garrad, M., Rossiter, J., Hauser, H.: Shaping behavior with adaptive morphology **3**(3), 2056–2062. https://doi.org/10.1109/LRA.2018.2807591
3. Ijspeert, A.J.: Central pattern generators for locomotion control in animals and robots: A review **21**(4), 642–653. https://doi.org/https://doi.org/10.1016/j.neunet.2008.03.014, https://www.sciencedirect.com/science/article/pii/S0893608008000804, robotics and Neuroscience

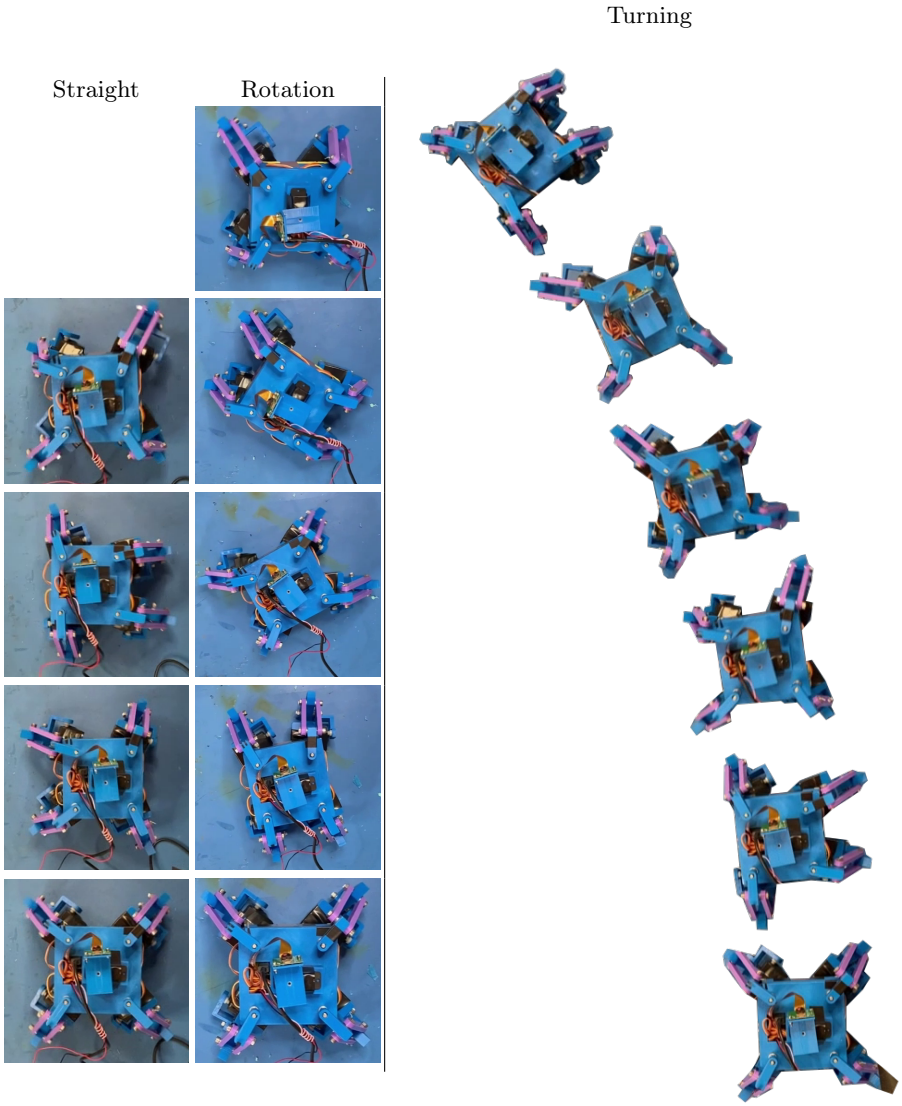


Fig. 8: Left to right, bottom to top, images in sequence showing limb positions chronologically resulting in straight line motion (left) and turning in place (center) and turning while moving (right) respectively. Note that the distance between subsequent poses for turning while moving has been exaggerated for clarity and the actual turning radius is much smaller.

4. Liu, C., Chen, Q., Zhang, J.: Coupled van der pol oscillators utilised as central pattern generators for quadruped locomotion. In: 2009 Chinese Control and Decision Conference. pp. 3677–3682 (2009). https://doi.org/10.1109/CCDC.2009.5192385
5. Liu, C., Chen, Y., Zhang, J., Chen, Q.: Cpg driven locomotion control of quadruped robot. In: 2009 IEEE International Conference on Systems, Man and Cybernetics. pp. 2368–2373. https://doi.org/10.1109/ICSMC.2009.5346399
6. Mao, S., Dong, E., ZHOU, L., Jin, H., TAN, Y., Liu, G., Xu, M., Low, K.H.: Design and gait analysis of a tortoise-like robot with soft limbs. In: CLAWAR 2015: 18th International Conference on Climbing and Walking Robots and the Support Technologies for Mobile Machines. pp. 215–223 (10 2015). https://doi.org/10.1142/9789814725248_0029
7. Matos, V., Santos, C.P.: Omnidirectional locomotion in a quadruped robot: A cpg-based approach. In: 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 3392–3397 (2010). https://doi.org/10.1109/IROS.2010.5652667
8. Nakajima, K., Hauser, H., Kang, R., Guglielmino, E., Caldwell, D., Pfeifer, R.: A soft body as a reservoir: case studies in a dynamic model of octopus-inspired soft robotic arm **7**. https://doi.org/10.3389/fncom.2013.00091, https://www.frontiersin.org/articles/10.3389/fncom.2013.00091
9. Wang, T., Wang, Z., Zhang, B.: Mechanism design and experiment of a bionic turtle dredging robot **9**(5). https://doi.org/10.3390/machines9050086, https://www.mdpi.com/2075-1702/9/5/86
10. Zani, P.A., Gottschall, J.S., Kram, R.: Giant Galapagos tortoises walk without inverted pendulum mechanical-energy exchange **208**(8), 1489–1494. https://doi.org/10.1242/jeb.01554, https://doi.org/10.1242/jeb.01554