Journal of Research and Applications in Mechanical Engineering
ISSN: 2229-2152 (Print); 2697-424x (Online)
(2024) Vol. 12, No. 2, Paper No. JRAME-24-12-023
[DOI: 10.14456/jrame.2024.23]

Research Article

Investigating Adaptive CPG-based Control of a Snake Robot with Switch Signal Input for Maneuvering in Varying Environments

P. Ngamkajornwiwat*
N. Pothita
Department of Robotics and automation, Faculty of Engineering and Technology, Panyapiwat Institute of Management, Nonthaburi, 11120, Thailand

Received 27 September 2023
Revised 22 December 2023
Accepted 1 January 2024

Abstract:
This study makes an essential contribution to robotics by revealing how to manage snake robots using adaptive Central Pattern Generators (CPGs) with switch signal input. The work optimizes CPG hyperparameters, characterizing robot behavior to improve navigation in obstacles and limited areas. Three indices are used to evaluate parameter adjustments: index based on moving time, index based on movement frequency, and index based on movement path. Quantitative investigation reveals how these hyperparameters influence the efficiency and adaptability of snake robots across situations. The findings emphasize the effectiveness of the proposed methodology by emphasizing quantitative values resulting from parameter modifications. The work uses stringent testing criteria to identify ideal parameter values ($\mu = 0.8, A = 0.007, B = 0.005$), boosting the approach's resilience and applicability.

Keywords: *Snake robot, Adaptive control, Central Pattern Generators (CPGs), Switch signal input, Hyperparameter optimization*

1. Introduction

Adaptive Central Pattern Generator (CPG)-based Control is a promising methodology in the field of robotics that has been utilized to regulate the movement and behavior of various robots. This approach is based on modeling the neural networks of animals and humans that control rhythmic movements such as walking and swimming. The CPG-based control system has been implemented in various types of robots, including humanoid, quadrupedal, and snake robots [1].

However, the conventional CPG-based control has shown limitations in its adaptability to different situations and environments [2]. This particular issue is particularly problematic in applications where robots require responsiveness to changing circumstances. For instance, snake robots possess a flexible body that necessitates a control system that can adjust to different terrains and obstacles. Traditional CPG-based control is insufficient to address this issue, making Adaptive CPG-based Control with Switch Signal Input a potential solution.

Adaptive CPG-based Control with Switch Signal Input is a variation of the conventional CPG-based control system that incorporates a switch signal input. The switch signal input allows the control system to switch between different CPG models depending on the robot's current situation. For example, if the robot encounters an obstacle, the switch signal input can trigger a different CPG model that allows the robot to navigate around the obstacle.

* Corresponding author: P. Ngamkajornwiwat
E-mail address: Potiwatnga@pim.ac.th

Despite its potential benefits, Adaptive CPG-based Control with Switch Signal Input still faces several challenges that require further attention. One challenge is the complexity of the control system, which increases as the number of CPG models and switch signal inputs increase. Another challenge is the need for further research to fully understand the capabilities and limitations of this approach. Nonetheless, this methodology offers a promising solution for controlling the movement and behavior of robots in dynamic and changing environments.

Numerous academic works highlight the need for further investigation into the impact of CPG parameters on snake-like robot locomotion [3] and its adaptive response to environmental changes [4-9]. The studies emphasize the significance of understanding how adaptive control strategies, facilitated by CPG-based controllers, allow snake robots to switch motion modes and navigate diverse terrains effectively. It is imperative to thoroughly examine the complex interplay between CPG parameters and the generation of adaptive motion through CPG-based control in snake-like robots. By understanding these dynamics better, we can fully utilize the potential of these robots in navigating challenging terrains. This exploration will lead to innovative approaches that help snake robots adapt to different terrains more efficiently and with greater stability.

Based on the recent advancements in snake-like robot control, incorporating central pattern generators for locomotion control has proven to be effective. The adaptive CPG system enables the robot to perform self-tuning behavior that is robust against short-time perturbations [10]. This research highlights the importance of investigating CPG parameters and their influence on snake-like robot locomotion, shedding light on potential advanced navigational capabilities and performance. By delving into the interplay between the CPG parameters and the generation of adaptive motion via CPG-based control in snake-like robots, researchers have laid the groundwork for a deeper comprehension of the mechanics involved. This understanding is pivotal to maximizing the efficacy of these robots in traversing diverse and demanding terrains. Recent progress in snake-like robot control has exhibited promise in integrating central pattern generators for locomotion control.

2. Methodology

2.1 Snake-Like Robot Platform

The research makes use of the Hirose-Fukushima Lab-designed ACM-R5 Snake robot platform in Fig. 1. The nine components that make up this snake robot's serpentine locomotion pattern each have two degrees of freedom. A length of 1100 mm and a weight of 9.5 kg are the robot's physical specifications [11]. With a speed of 0.4 m/s, it can move through a variety of surfaces, including water and land. In this project, we use proximity sensors to make object detection easier. These sensors deliver accurate information using switch signals. These sensors can detect objects from a distance of less than ten cm. CoppeliaSim Edu is used as an efficient tool for testing different scenarios and reviewing the performance results attained by our robotic system in order to simulate and evaluate the motion capabilities.

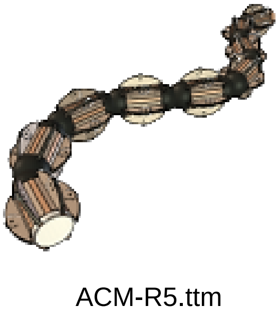


Fig. 1. ACM-R5 robot

2.2 Simulation Platform

CoppeliaSim Edu (CoppeliaSim Educational version) is a program used for simulating and prototyping robot models and controls for research and development purposes. Developed by Coppelia Robotics, it is widely used in the education and research sectors.

CoppeliaSim [12] is a versatile software that offers several features for robot development and testing. It includes physics simulation capabilities, allowing users to simulate the behavior of robots and objects. Users can create their own robot models or use existing ones from the library, and CoppeliaSim supports programming control through both scripted and graphical programming. The software also supports multiple robot formats available in the market, providing flexibility for testing different types of robots. Additionally, there is a free Edu version of CoppeliaSim that provides cost-effective solutions for educational and research purposes

CoppeliaSim Edu using Lua scripts as program and settings as follows: Physics engines as Bullet 2.78, balanced setting (default), time step dt = 50 ms (default).

2.3 Adaptive Neural Oscillator with Synaptic Plasticity

In this study, a self-adjusting oscillator network was used to control the movements of serpents using the screw drive developed by Nachstedt et al. [10]. The synaptic plasticity and adaptable neural oscillators use encoders to stimulate the oscillator. The three joints of snake robots are equipped with encoders to measure angles; each joint value is used to provide feedback, which is then used to create equations, changing the equation system. In other words, adjustable CPGs enable P equation solution and result feeding into adaptive neural oscillators, resulting in the establishment of synaptic plasticity in the control network in response to encoder value. The feedback and analytical information derived from the encoder value can be employed to navigate and adjust the frequency by the surrounding environment.

The control mechanism generated from data and integrated into the robot joints by the researchers was successfully implemented. Subsequently, this system was enhanced to incorporate a proximity sensor, whereby the switch signal assumes binary values of zero and one. This sensor enables a control mechanism that modifies the robot's motion with a single sensor, which reacts to the frequency of the robot's movement in accordance with the surrounding environment.

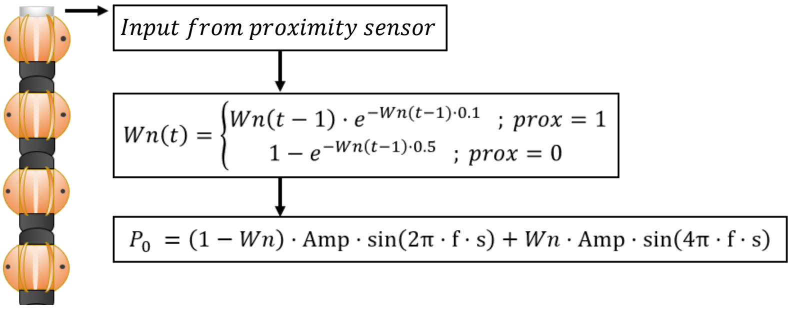

Input from proximity sensor
$Wn(t) = \begin{cases} Wn(t-1)\cdot e^{-Wn(t-1)\cdot 0.1} & ;\ prox = 1 \\ 1 - e^{-Wn(t-1)\cdot 0.5} & ;\ prox = 0 \end{cases}$
$P_0 = (1 - Wn)\cdot \mathrm{Amp}\cdot \sin(2\pi\cdot \mathrm{f}\cdot \mathrm{s}) + Wn\cdot \mathrm{Amp}\cdot \sin(4\pi\cdot \mathrm{f}\cdot \mathrm{s})$

Fig. 2. Design control equations for sensing robots as input switches for use with Adaptive CPGs Show the order of operations in the robot

The process of working equations in Fig. 2 that affect the movement of the robot. First, input data from the proximity sensor. The robot's actions are received by the proximity sensor with readings of 0 and 1.

$$Wn(t) = \begin{cases} Wn(t-1)\cdot e^{-Wn(t-1)\cdot 0.1} & ;\ prox = 1 \\ 1 - e^{-Wn(\ -1)\cdot 0.5} & ;\ prox = 0 \end{cases} \tag{1}$$

The sensor's value, also known as the reliability value, is transformed to fit the Wn(t) equation. From receiving data that is 1, Wn increases greatly when the proximity value prox = 1, and if it is less than 0, Wn is set to 0.01 when prox = 0. By changing the value collected from the sensor, this equation operates. The greater the input value during that time, such as 1, the greater the value of Wn. The value of Wn is reduced with the frequency of the input from the proximity by making a value of Wn without using the obtained value when the value is 0 during that time and the graph generated from the Wn equation has a value equal to 1 in that case. This is the sensor's process of learning as it compares input values of only 0 and 1 to data of wide or narrow space values as shown in Fig. 3. To achieve learning from that signal, direct input is used.

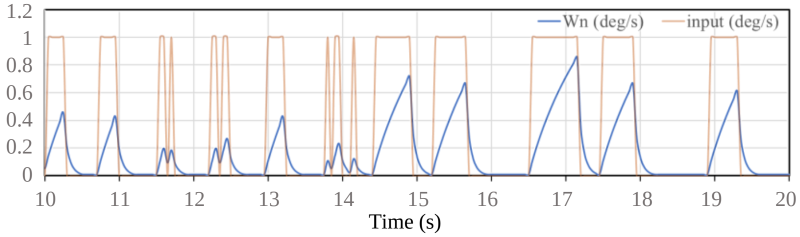


Fig. 3. Example of working equation Wn(t) while receiving a signal.
To be calculated with the equation of Adaptive CPGs.

The system's functioning with the snake robot, including calculating values from the proximity sensor and converting them to movement patterns. The variables and equations utilized to calculate adaptive CPGs and define movement patterns are illustrated in Fig. 4. This section also explains how motor control is achieved by employing the values acquired from CPG calculations.

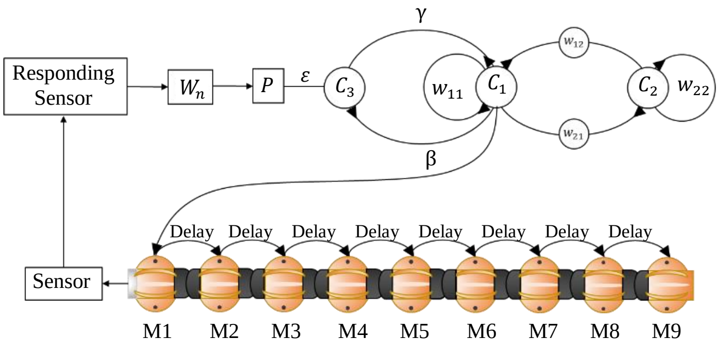


Fig. 4. The output from C_1 is O_1 is used to govern the movements of each robot joint to demonstrate.

$$\beta(t+1) = \beta(t) - A \cdot a_1(t) \cdot a_3(t) - B \cdot (\beta(t) - \beta_0) \tag{2}$$

$$\gamma(t+1) = \gamma(t) - A \cdot a_3(t) \cdot a_1(t) - B \cdot (\gamma(t) - \gamma_0) \tag{3}$$

$$\varepsilon(t + 1) = \varepsilon(t) + A \cdot P(t) \cdot a_3(t) - B \cdot (\varepsilon(t) - \varepsilon_0) \tag{4}$$

The parameter φ in Eq. 5 is adjusted accordingly. According to the frequency modulation rule with learning rate μ:

$$\varphi(t+1) = \varphi(t) + \mu \cdot \gamma(t) \cdot o_2(t) \cdot w_{01}(+) \cdot o_2(t) \tag{5}$$

The weights β and γ decay toward their relaxation values if the perturbation P(t) and the output a_1 (t) are different, and P(t) predominates in the output of a_3. The outputs of the oscillator a_1 and a_3 are positively correlated as soon as it has adapted to the external frequency. As a result, until the feedback signal from a_1 to a_3 almost entirely balance the external perturbation, γ decreases and β increases (according to amount). This causes ε to decay toward its relaxation values, and as a result, β and γ as well. Feedback is no longer required if the value of P(t) is zero, indicating that the synapses have converged to their relaxation levels.

$$P_0 = (1 - Wn)\mathrm{Amp} \cdot \sin(2\pi \mathrm{fs}) + Wn \cdot \mathrm{Amp} \cdot \sin(4\pi \mathrm{fs}) \tag{6}$$

The third step makes advantage of the adaptive central pattern generators idea. P_0 equations are produced, which disrupt the C_3 signal and enable the robot to modify its frequency in order to operate autonomously in a smaller area. The variables utilized in the computation are taken from the P_0 equation and used to calculate the values Amplitude = 2, frequency = 0.6, s = time, and Wn = based on the response of the incoming input signal. Through calculations, CPG3(C_3) is altered, and CPG1(C_1) also experiences an additional influence that modifies the frequency in accordance with the moving area.

3. Parameter Evaluation

In this research, the principle of evaluating the movement efficiency of the robot has 3 indexes: 1. Index based on moving time, 2. Index from movement frequency, and 3. moving path. There are methods for evaluating all 3 indices by observing the movement of the robot from the beginning to the end as in Fig. 5. The details of calculating all 3 indices are as follows.

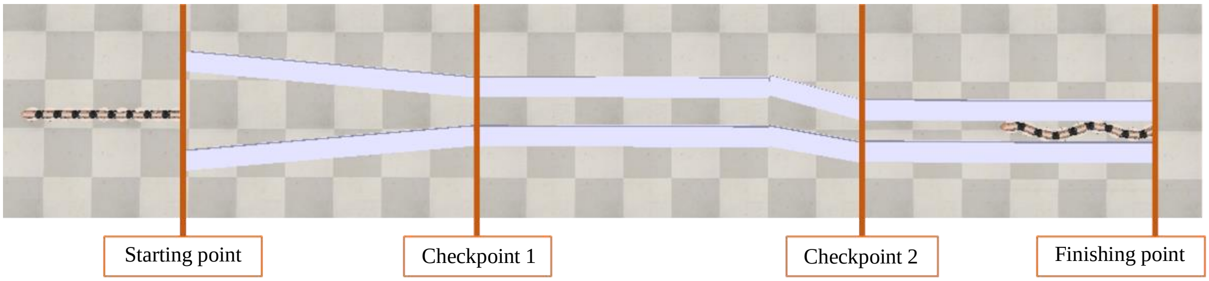


Fig. 5. Indicates the test location in a narrow space. This area has 4 observation points (Start, Checkpoint 1, Checkpoint 2 and Finish).

3.1 Index Based on Moving Time

Index ranges from 1 and up, with the highest value is the best value and the worst value is 1. The principle for calculating the time index is to measure the time from the starting point and stop the time when the robot's head touches the finishing point, as shown in Fig. 5. Divide the highest time value by the time in each cycle as in Eq. (7).

$$Time\ score = \frac{Max\ TimeTime}{(Iteration)} \tag{7}$$

3.2 Index from Movement Frequency

The index has the same range and effect as 3.1, with a different calculation principle. Calculating the frequency which indicates the energy used in the movement of the robot by means of finding Motor value of experimental results analyze the frequency data with Fourier analysis and transform the results into graphs. Amplitude vs. Frequency and then calculate the area under the graph.

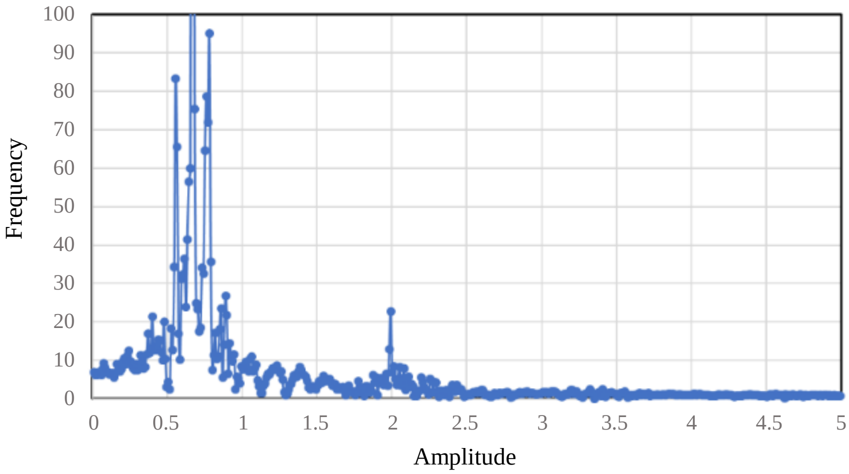


Fig. 6. The graph represents the amplitude by frequency that results from the Fast Fourier Transform (FFT).

How to find space from the frequency of movement:

1) Prepare the recorded results CPG_Output1 From the time the robot starts the timer from the start point to the finish point (The outcome is equal to the time in 3.1) the value from the motor controlled in the robot Caused by Adaptive CPG calculations by taking the calculation of the values released as CPG_Output1 Used to create frequencies to control the motor.

2) Take the results of the experiment to perform the analysis process using Fourier analysis to convert the function f(t) from time domain to frequency domain using Discrete Fourier Transform (DFT) from the Eq. (8), x(ω) is Frequency domain, x(t) is time domain.

$$x(\omega) = F[x(t)] \tag{8}$$

3) Combine all the results to get the area under the function graph f(t) in the period (start time, End time) Let's analyze the results according to the starting, finishing point of the obstacle. Next, create a point to find each frequency, with the formula being point at that time divided by (time at the end point - start time).

4) Then bring to Amplitude arising from Fast Fourier Transform (FFT) in order to convert the real and imaginary coefficient into complex numbers of the Amp + Freq(i) and apply to the Fourier analysis value to get the value of Amplitude in Fig. 6.

5) The next step is to find point to point of area. Which is a search for each range point in the Discrete format from the Eq. (9) and bring the total area under each graph together until the last time. The equation is Sum area = The sum of each area. From the start time to the end time in Eq. (10) will get the area under the graph of μ value, it will be able to find the energy used. To travel, find the sum of the space under the graph to compare energy efficiency in each parameter value.

$$Area = \frac{(Amp_i + Amp)}{2(Freq_i - Freq)} \tag{9}$$

$$Sum\ Area = \sum_{i=1}^{n} Area_i \tag{10}$$

The index that brings the time value that is equal to the move time index divided by the area value under the maximum graph, as in the Eq. (11) ranges from 1 point and above, with the greater the value, the better and the worst value being 1 point. The Fourier analysis is the guiding principle. The frequency that makes movement during this time use less energy is indicated by the (FA) score at Much.

$$FA\ Score = \frac{Time(\mu)}{Max\ Area} \tag{11}$$

3.3 Index from the Path of Movement

The location values (x, y) of the robot's movement are gathered from the position data collection and used to examine the outcomes. The distance from each coordinate point to be calculated is utilized to approximate the distance while determining the distance. Each point to point has a distance of 0.05 seconds, which is calculated by using its coordinates (x, y) to get its distance value in Eq. (12). Next, the sum of the distances traveled from each point's coordinates to the starting position is calculated in Eq. (13). It comes to an end when it achieves its conclusion. From finding the sum of the distances used this makes it possible to analyze data in which the robot uses the shortest moving distance to make decisions, as shown in Eq. (14).

$$Distance\ point = \sqrt{(x_{i+1} - x_i)^2_+ (y_i - y)^2} \tag{12}$$

$$Sum\ Distance = \sum_{i=1}^{n} Distance\ point_i \tag{13}$$

$$Distance\ Score = \frac{Time(\mu)}{Sum\ Distance} \tag{14}$$

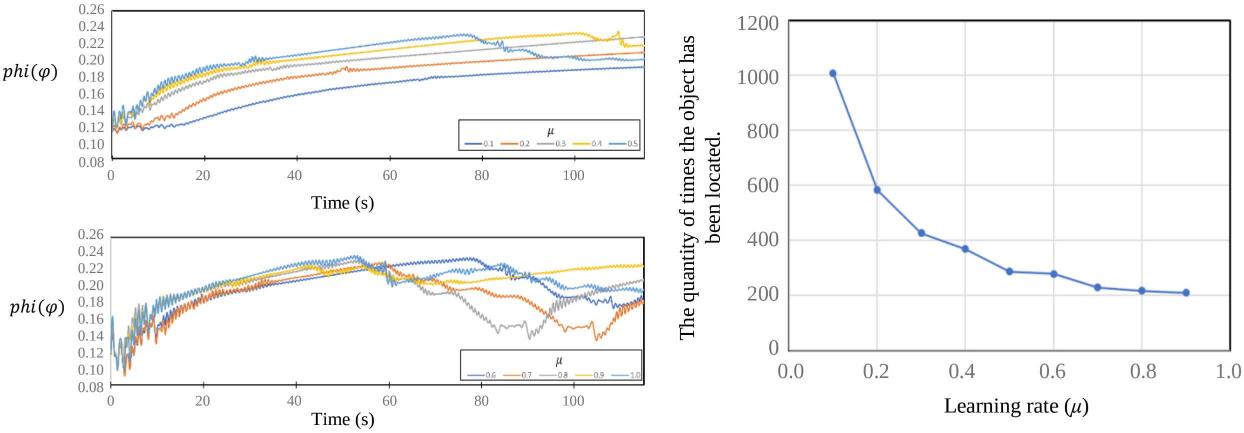
3.4 Evaluate All Results

The score will be determined by multiplying all three index values, as shown in equation (14), following the collection of all three index values. The smallest sums are those that are closer to 1, and the best values to evaluate using these three indexes are those that are highest.

4. Experiment

4.1 Learning Rate (μ) Parameter Experiment

To determine the relationship between changes in movement patterns that react to environment in various ways, conduct an experiment taking the parameter value at the mew value into consideration. There is a technique for taking into account all 3 aspects of the index and comparing the outcomes to gauge how effectively it is being used to the ideal parameter values to be used.



(a)

(b)

Fig. 7. a) Value of μ with values of 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9 and 1.0., b) graph that displays the frequency of object detection in the simulated area

From Fig. 7 a) the experiment changed the value of μ with values of 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9 and 1.0. It can be seen that the value of μ affects the change in the value of phi, causing the adaptive of Moving in a narrow space in the snake robot experiment in a narrow space, the phi value will be higher if the robot moves in a narrow space with an input signal of 1 for a long time. Using the test area as shown in Fig. 5.

b) A graph is presented in Fig. 7. Show as the frequency of object detection in the simulated area with varying μ values of 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, and 0.9. As can be seen, the range of values with increasing mew causes a decrease in the frequency of encounters with objects as it moves out of space but converges to a particular value. The experimental values were determined: A = 0.005, B = 0.0038, C_3 = 0.5, C_1 = 0.6, C_2 = 0.6, Amplitude = 2, frequency =0.6, phi =Π/28.

When taking the data from the graph in Fig. 7, the values obtained from all 3 results of experiments are shown in Table 1.

Table 1: Experimental results, μ value and 3 results from the experiment

μ	Time (s)	FA area	Distance (m)
0.4	54	52.369	16.455
0.5	46	38.054	18.589
0.6	45	36.951	14.285
0.7	39	41.047	12.958
0.8	38.6	40.591	12.934
0.9	38	42.493	12.977

Table 2: Evaluate all results scores.

μ	Time score	FA score	Distance score	Result
0.4	1.0	1.00	1.13	1.13
0.5	1.17	1.38	1.00	1.62
0.6	1.20	**1.42**	1.30	2.21
0.7	1.38	1.28	1.43	2.53
0.8	1.38	1.29	**1.44**	**2.57**
0.9	**1.42**	1.23	1.43	2.51

After conducting an experiment that took the parameter value at the mew value into consideration, it was found that the value of μ has a significant impact on the movement patterns of the snake robot when navigating through narrow spaces. By comparing the outcomes of all 3 aspects of the index, it was determined that the value with the highest score was 0.8, followed by 0.7, 0.9, 0.6, 0.5, and 0.4, respectively, as shown in Table 2.

The highest efficiency in terms of time taken to pass test points was achieved with $\mu = 0.9$, which had a score of 1.42 due to its high learning rate. This rapid convergence allows for quick adjustment to a high frequency, resulting in the shortest amount of time required to navigate narrow spaces.

Based on the efficiency in terms of energy consumption, it was observed that a value of $\mu = 0.6$ resulted in the lowest energy usage when moving through the test point with a score of 1.42. In order to minimize power consumption during frequency adjustments, selecting a value of $\mu = 0.6$ is considered to be the optimal choice for reducing power requirements.

Finally, in terms of the distance traveled, the value $\mu = 0.8$ was determined to be the most efficient, with a score of 1.44. The scores for values 0.7 and 0.9 were both equal at 1.43, which is only slightly lower by 0.01 points compared to the score for $\mu = 0.8$.

4.2 A and B Parameter Experiment

The values of parameters A and B should be within the proper range. It must lead to adaptive by adjusting of CPGs, too. The signal interference between C_3 and C_1 will not be adaptive, resulting in no change in frequency in accordance with the environment that occurs or creating behavior in Fig. 8, if there is too much or too little adjusted value that is out of balance with the P equation slowly alters movement value, causing distorted movement patterns.

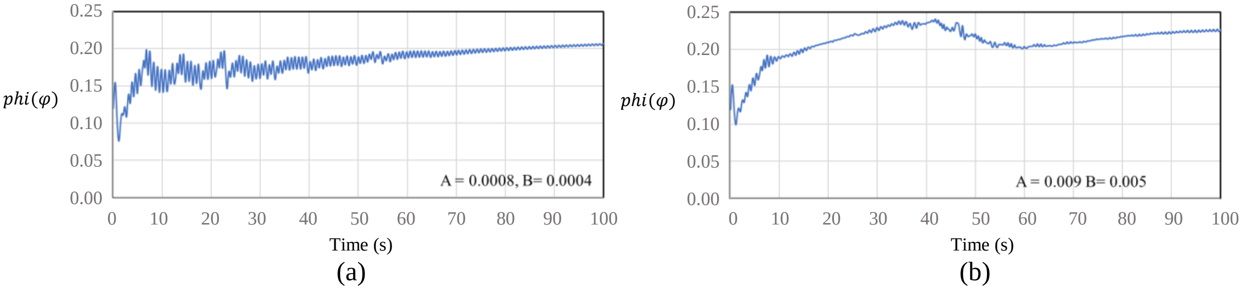


Fig. 8. (a) Graph of resulting phi values of A = 0.0008, B = 0.0004, (b) Graph of resulting phi values of A = 0.0009, B = 0.0005

(a) The Pattern of graph resulting phi value has no effect on adaptive movement behaviour. Additionally, there is a propensity for the value of phi to increase, leading to a higher movement frequency.

(b) With each exit of the finishing point, phi's value rises and falls. To put it another way, the frequency will drop off when there are no obstacles present and this is in accordance with the equation that requires reducing the frequency when no obstacles are encountered.

Table 3: A The possible range for changing the frequency.

B A	0.00005	0.0005	0.005	0.05
0.0006	-	-	-	-
0.006	-	-	✓	-
0.06	-	-	-	-
0.6	-	-	-	-

Parameters A = 0.006 and B = 0.005, illustrated in Table 3, represent a specific configuration under scrutiny. Our experimental findings suggest that maintaining A and B within their specified ranges has the potential to significantly lower operational frequency, which holds promise for enhancing system performance. Expanding our investigation, we've extended the parameter range to encompass A = (0.001 - 0.012) and B = (0.001 - 0.009), as shown in Table 4.

Table 4: A and B values can be utilized in the score table.

B A	0.001	0.002	0.003	0.004	0.005	0.006	0.007	0.008	0.009
0.001	-1	-1	-1	-1	-1	-1	-1	-1	-1
0.002	0	-1	-1	-1	-1	-1	-1	-1	-1
0.003	1	1	-1	-1	-1	-1	-1	-1	-1
0.004	1	1	1	-1	-1	-1	-1	-1	-1
0.005	0	1	1	1	-1	-1	-1	-1	-1
0.006	0	0	1	1	1	-1	-1	-1	-1
0.007	0	0	1	1	1	1	-1	-1	-1
0.008	0	0	1	1	1	1	1	-1	-1
0.009	0	0	0	1	1	1	1	1	-1
0.010	0	0	0	1	1	1	1	1	1
0.011	0	0	0	1	1	1	1	1	1
0.012	0	0	1	1	1	1	1	1	1

The experimental findings show values A and B. Values with a -1 indicate that they cannot pass through the simulated area, while values with a 1 show that the robot can respond to environmental changes can do departing from the end point and lower the frequency in 100 sec. 0 means that the robot can only move through the area once it has passed the endpoint. After that, there is nothing but empty space with no potential pitfalls. In the first 100 seconds, the robot cannot switch to a lower frequency.

Table 5: The results of the change in phi value with the signal being reversed.

B A	0.001	0.002	0.003	0.004	0.005	0.006	0.007	0.008	0.009
0.001	-1	-1	-1	-1	-1	-1	-1	-1	-1
0.002	-1	-1	-1	-1	-1	-1	-1	-1	-1
0.003	0	0	-1	-1	-1	-1	-1	-1	-1
0.004	0	0	0	-1	-1	-1	-1	-1	-1
0.005	0	0	0	0	-1	-1	-1	-1	-1
0.006	0	0	0	1	0	-1	-1	-1	-1
0.007	0	0	0	0	1	0	-1	-1	-1
0.008	0	0	0	0	1	0	0	-1	-1
0.009	0	0	0	0	0	0	0	0	-1
0.010	0	0	0	0	0	0	0	0	0
0.011	0	0	0	0	0	0	0	0	0
0.012	0	0	0	0	0	0	0	0	0

The test results for the value of phi (φ) that results in an inversion of the signal are recorded in the Table 5. If the robot's value is 1, the signal will be inverted as it moves lower and lower in frequency and the value obtained with this score will show the behavior as shown in Fig. 9 and Fig. 10. Therefore, 0 is the value that can pass the test point, increasing higher frequencies even when not activated by a signal. However, because there is no change in value prior to the signal reversal, there is no reversal of the signal value. Values with a -1 indicate that they cannot pass through the simulated area.

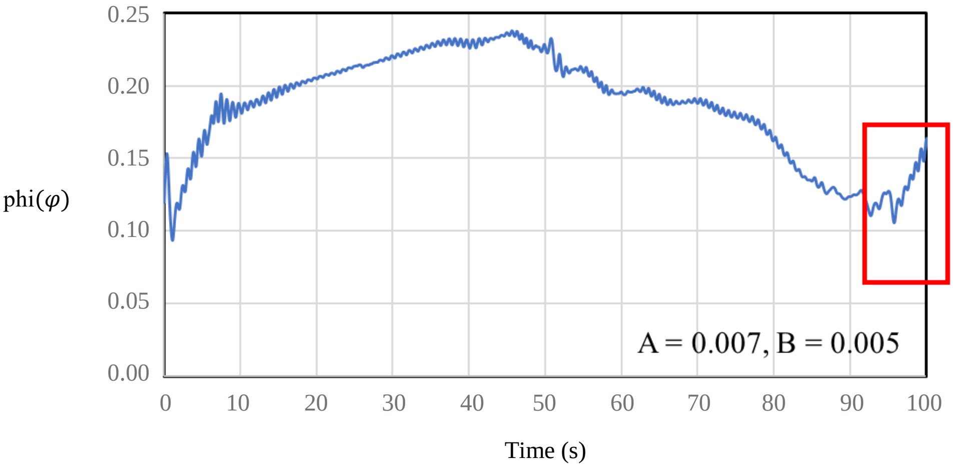


Fig. 9. Shows the difference that occurs with learning where signal degradation causes signal reversal. During the period when there are no obstacles, the phi value continues to decrease, affecting the variables in the CPG equation. After adjusting μ = 0.8, A = 0.007, B = 0.005, Time spent passing through the experimental area was 34.75 seconds, the FA area was 38.82 and the distance used was 12.23 m.

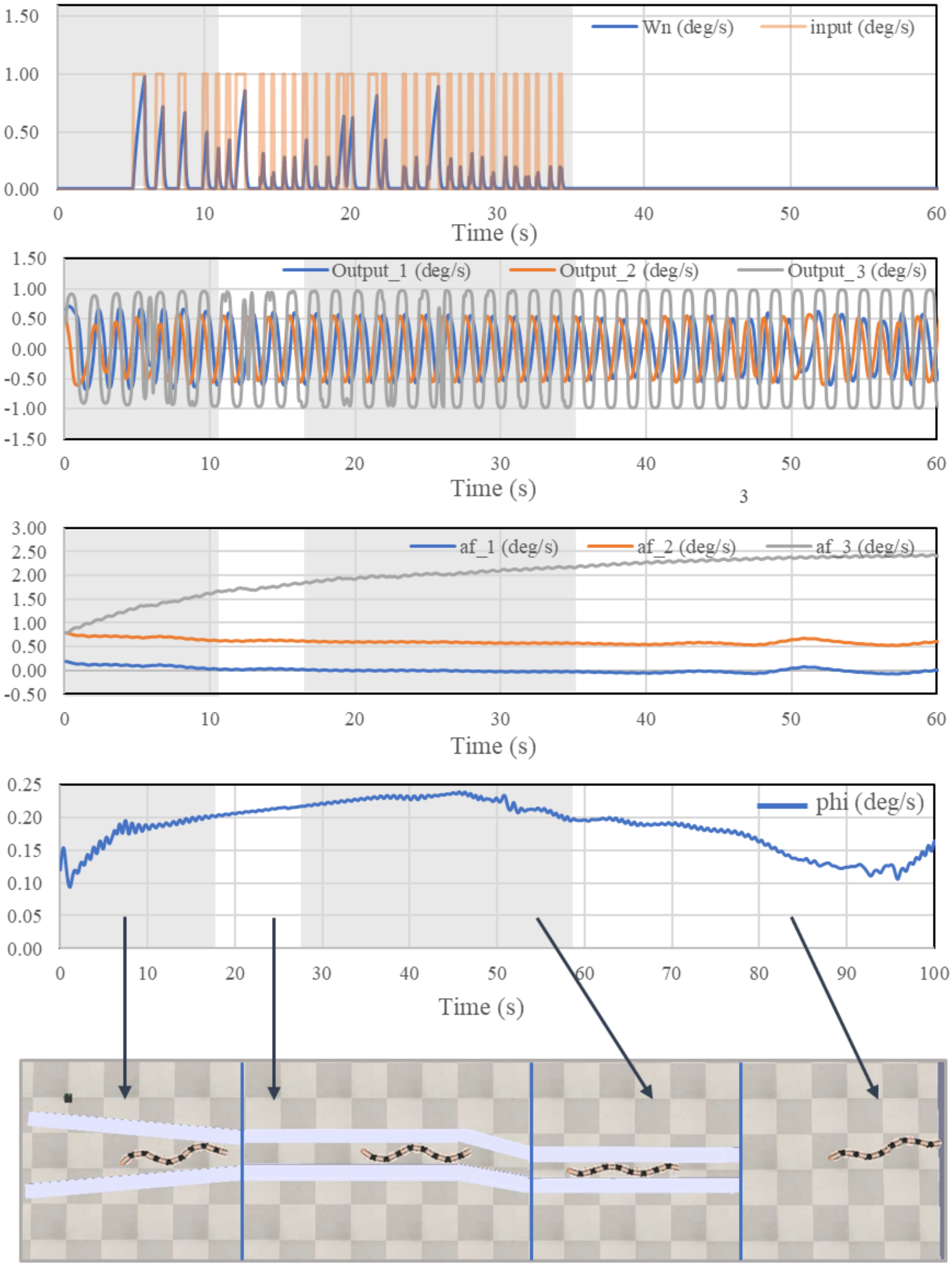


Fig. 10. Response results from the experiment from Fig. 9 with various values presented.

In testing in various simulation areas as shown in Fig. 11, the parameters as shown in Fig. 9 were tested and used in other experiments. It was found that it was possible to pass through other simulated areas and have behavior that can increase and decrease the frequency of movement when encountering objects, the obtained values are shown in Fig. 12, which is a graph that indicates the μ value of the calculated movement and indicates the frequency with which the robot occurs. The simulated area was split into four formats by the experiment: a, b, c, and d. According to Fig. 11.

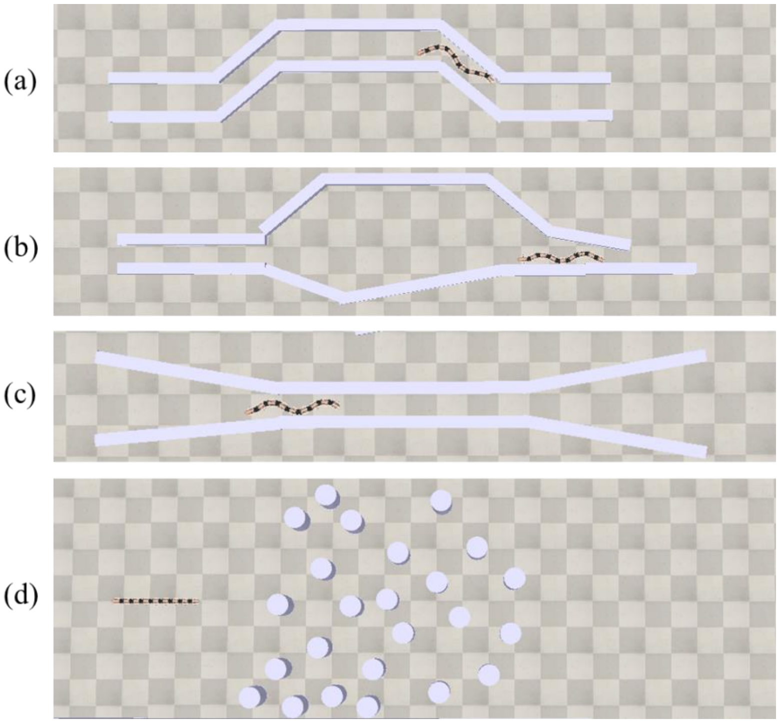


Fig. 11. Various Simulation areas.

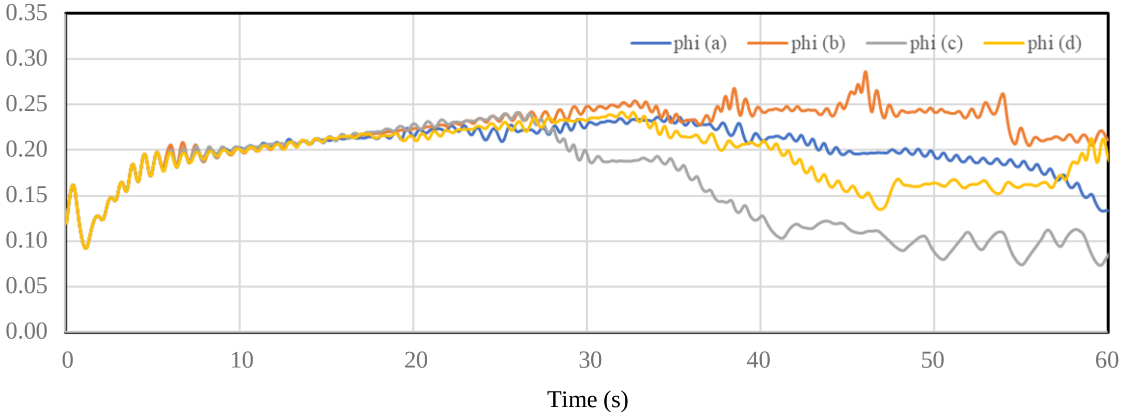


Fig. 12. Various Simulation areas. As shown in Fig. 11, show the frequency changes that are transformed based on the simulated environment (a), (b), (c), and (d). It should be able to exit an area with barriers or simulated channels as the intended mobility result. There will be a reduced frequency by increasing the high frequency when the region is discovered to be a narrow area and decreasing the frequency when leaving the area and being an open space without identifying items.

5. Conclusion

This study has contributed significantly to the realm of robotics, shedding light on the potential advancements in snake robot management through the implementation of adaptive Central Pattern Generators (CPGs) coupled with switch signal input. The capacity of snake robots to navigate through intricate spaces and negotiate obstacles is greatly enhanced by this novel approach.

The central objective of this research was to delve into the optimization of hyperparameters within the CPG equations, a crucial step towards accurately characterizing the robot's behavior. This investigation employed three key indices: one based on movement duration, another on movement frequency, and the third on the path of movement.

Our findings highlight the substantial impact of parameter μ, as evidenced by its role in shaping the snake robot's behavior in confined spaces. An optimal μ value of 0.8 emerged as the most effective, leading to reduced passage time, minimal energy expenditure, and shorter travel distances within the experimental area.

The implications of this research reverberate across the field of robotics, particularly in scenarios demanding adaptability to rapidly changing environments. The integration of switch signal input into CPG-based control systems holds great promise for addressing the challenges posed by dynamic terrains. As we chart our course forward, it is imperative to conduct further research to tackle the scalability issues associated with increasing the number of CPG models and switch signal inputs. A comprehensive exploration of the entire spectrum of capabilities and constraints of this approach remains a vital avenue of inquiry.

In summary, this study opens doors to the development of highly adaptable snake robots, poised to excel in diverse applications ranging from search and rescue missions to exploratory endeavors in arduous terrains. The fusion of biological inspiration with cutting-edge technology continues to forge new frontiers in the domain of robotics.

Nomenclature

CPG	Central Pattern Generator,
C_1	CPG 1,
C_2	CPG 2,
C_3	CPG 3,
O_1	Output from C_1,
O_2	Output from C_2,
O_3	Output from C_3,
Prox	Proximity sensor,
P	perturbation,
Wn	The equation used to convert the response signal of the input signal,
phi (φ)	frequency that occurred is represented by this value,
μ	Learning Rate,
A	Weight of impact on CPG equation,
B	Weight of impact on CPG equation,
FA	Fourier analysis

Acknowledgments

This work is supported by Panyapiwat Institute of Management and Innovation Center for Robotics and Automation Systems (iCRAS).

References

[1] Manzoor S, Choi Y. A unified neural oscillator model for various rhythmic locomotions of snake-like robot. Neurocomputing. 2016;173:1112-1123.

[2] Blanchard D, Aihara K, Levi T. Snake robot controlled by biomimetic CPGs. J Robot Netw Artif Life. 2019;5(4):253-256.

[3] Hu WJ, Zhao LY, Yan JL, Li JX, Wang ZL. Locomotion patterns based on a CPG model of snake robot. Proceedings of the 3rd Annual International Conference on Mechanics and Mechanical Engineering (MME 2016); 2016 Dec 16-18; Chengdu, China. Amsterdam: Atlantis Press; 2017. p. 588-593.

[4] Haghshenas-Jaryani M. Dynamics and computed-muscle-force control of a planar muscle-driven snake robot. Actuators. 2022;11(7):194.

[5] Baysal YA, Altas IH. A fast non-dominated sorting multi-objective symbiotic organism search algorithm for energy efficient locomotion of snake robot. Comput Sci Inf Syst. 2022;19(1):353-378.

[6] Szadkowski R, Prágr M, Faigl J. Self-learning event mistiming detector based on central pattern generator. Front Neurorobot. 2021;15:629652.

[7] Tang C, Shu X, Meng D, Zhou G. Arboreal concertina locomotion of snake robots on cylinders. Int J Adv Robot Syst. 2017;14(6):1-13.

[8] Wang G, Chen X, Han SK. Central pattern generator and feedforward neural network-based self-adaptive gait control for a crab-like robot locomoting on complex terrain under two reflex mechanisms. Int J Adv Robot Syst. 2017;14(4):1-13.

[9] Wang M, Yu J, Tan M. CPG-based sensory feedback control for bio-inspired multimodal swimming. Int J Adv Robot Syst. 2014;11(10):1-11.

[10] Nachstedt T, Wörgötter F, Manoonpong P, Ariizumi R, Ambe Y, Matsuno F. Adaptive neural oscillators with synaptic plasticity for locomotion control of a snake-like robot with screw-drive mechanism. 2013 IEEE International Conference on Robotics and Automation (IEEE ICRA); 2013 May 6-10; Karlsruhe, Germany. USA: IEEE; 2013. p. 3389-3395.

[11] Transeth AA, Pettersen KY, Liljebäck P. A survey on snake robot modeling and locomotion. Robotica. 2009;27:999-1015.

[12] Tamura H, Kamegawa T. Parameter search of a CPG network using a genetic algorithm for a snake robot with tactile sensors moving on a soft floor. Front Robot AI. 2023;10:1138019.