GAS Journal of Engineering and Technology (GASJET)

Volume 3 | Issue 1, 2026 Homepage: https://gaspublishers.com/gasjet-home/ ISSN: 3048-5800

Bioinspired Locomotion Control of Snake Robots: CPG-Based and Learning Approaches

Md Azizur Rahman*, Rui Bai & Lianqiang Zhang

School of Mechanical Engineering, Southwest Jiaotong University, Chengdu 610031, China

Received: *10.01.2026* | ***Accepted:*** *30.01.2026* | ***Published:*** *30.01.2026*

***Corresponding Author:** Md Azizur Rahman

DOI: *10.5281/zenodo.18431355*

Abstract **Review Article**

Snake robots represent a paradigm shift in mobile robotics for navigation in unstructured environments, leveraging bioinspired control derived from natural neural mechanisms. This review synthesizes research on snake robot locomotion control, focusing on two dominant methodologies. Central Pattern Generator (CPG) based control and learning-based approaches. We examine Matsuoka and Hopf oscillators, parameter optimization via evolutionary algorithms, and hybrid CPG-learning architectures. Learning approaches encompass reinforcement learning, evolutionary optimization, and spiking neural networks. We synthesize insights from peer-reviewed literature covering rigid and soft platforms, evaluate comparative performance across terrains, and identify critical gaps in sim-to-real transfer, energy efficiency, and sensory integration. This review guides researchers toward practical deployment of adaptive snake robot systems.

Keywords: Snake robot, central pattern generator, reinforcement learning, bioinspired control, gait generation, soft robotics.

1. INTRODUCTION

Biological snakes represent nature's most remarkable locomotion achievements, navigating complex terrain with minimal energy expenditure. Unlike wheeled and legged robots constrained by rigid morphologies, snakes exploit continuous body deformation, anisotropic friction interactions, and distributed neural control to traverse environments from sandy deserts to rocky mountains to underground tunnels (Seeja et al., 2022). This capability emerges from three integrated elements compliant musculoskeletal systems enabling continuous deformability, distributed contact exploiting environmental friction as propulsion aids, and spinal Central Pattern Generators distributed neural circuits capable of autonomous rhythm generation. Engineering snake robot platforms inspired by these biological principles offers unprecedented capability for deployment where conventional robots fail.

Practical applications drive snake robot development. Search and rescue requires navigation through collapsed structures following disasters where wheels and legs prove useless. Infrastructure inspection demands traversing confined pipelines and utility networks in industrial facilities and nuclear plants. Environmental monitoring requires exploring cave systems and underwater environments inaccessible to conventional platforms. Medical applications including minimally invasive surgery require manipulation within constrained anatomical spaces. Industrial platforms including the ACM-R series and Kulko demonstrate commercial maturity with operational systems deployed in real-world missions (P. Ngamkajornwiwat & N. Pothita, 2024).

Snake robot control presents distinct challenges absent in traditional robotics. High kinematic dimensionality 30+ degrees of freedom makes conventional inverse kinematics intractable. Distributed environmental contact creates complex dynamics; the robot simultaneously interacts with terrain through multiple body segments experiencing unique contact forces and friction. Environmental adaptation is essential; locomotion performance depends critically on friction coefficients, obstacle distributions, and terrain compliance. Real-time computational constraints on embedded microcontrollers requiring execution within 10-100 milliseconds preclude complex controllers. These constraints necessitate alternative paradigms exploiting inherent locomotion structure rather than centralized trajectory planning.

Two complementary control paradigms address these constraints. Central Pattern Generator-based approaches leverage biological inspiration (Campanaro et al., 2021), generating locomotion through distributed oscillatory circuits requiring minimal tuning and executing in real-time on resource-constrained systems. Learning-based approaches enable automatic gait discovery through reinforcement learning and evolutionary algorithms, achieving superior performance when computational resources are available (Liu et al., 2025). Hybrid architectures combining CPG structures with learned parameters represent emerging best practice (Liu et al., 2023a), balancing biological plausibility with adaptive capability while maintaining real-time feasibility.

This review synthesizes research across CPG-based and learning-based methodologies, performance analysis across platforms and terrains, critical research gaps, and future directions. We examine mathematical foundations (Matsuoka and Hopf oscillators), parameter optimization (genetic algorithms, particle swarm optimization), reinforcement learning (PPO, DDPG), and hybrid integration. Specific focus addresses sim-to-real transfer challenges, energy efficiency limitations, and standardization requirements essential for enabling practical autonomous snake robot systems.

2. BACKGROUND: SNAKE LOCOMOTION PRINCIPLES

2.1 Snake Robot Platforms and Architectures

Contemporary snake robot platforms exhibit diverse architectures reflecting design philosophy and application requirements (P. Ngamkajornwiwat & N. Pothita, 2024). As shown in Figure 1 modular rigid platforms like the ACM-R series employ 2 degrees of freedom per segment enabling roll and yaw motion through servo-actuated joints. Recent soft robot designs emphasize pneumatic or hydraulic actuators embedded in compliant materials (Lu et al., 2024), offering compliance for safe interaction and continuous deformability enabling passage through spaces far smaller than the robot's relaxed configuration. Hybrid platforms integrate rigid skeletal structures with soft actuators, combining structural support and sensor mounting with compliant deformation. Wheel-equipped platforms exploit anisotropic friction by mounting passive wheels perpendicular to joint axes (Fukuoka et al., 2023), effectively decoupling lateral and forward motion to reduce energy requirements on terrestrial surfaces. These diverse platforms share common control challenges: high kinematic dimensionality, distributed environmental contact, and real-time computational constraints on embedded systems (Seeja et al., 2022).

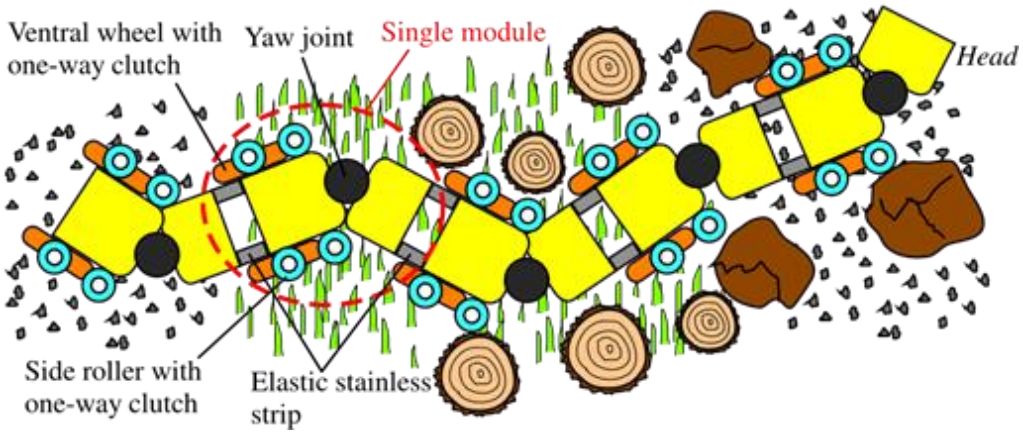

Figure 1. Detailed view of modular snake robot mechanical design showing segmental structure with articulated joints, actuators, and sensor integration enabling flexible locomotion across diverse terrains.

2.2 Kinematic and Dynamic Modeling

The fundamental serpenoid equation describes snake undulation:

$$y=B\sin\left(\frac{2\pi x}{\lambda}+\frac{2\pi t}{T}+\epsilon\right) \tag{1}$$

Where B is amplitude, λ is wavelength typically 0.8-1.2 body lengths, T is period corresponding to undulation frequency, t is time, and ϵ is phase offset. **Equation 1 Serpenoid curve equation** specifies the spatial-temporal wave pattern that forms the basis for CPG-based controllers (Liu, Gasoto, Onal, et al., 2020). This spatial-temporal wave pattern specification forms the basis for CPG-based controllers. The serpenoid curve describes the trajectory that each point along the snake's body traces during locomotion. The wavelength determines how many body waves are present simultaneously shorter wavelengths create faster, more frequent undulations while longer wavelengths produce slower, more powerful movements. The phase offset enables directional control; asymmetric phase relationships between left and right body sides steer the robot.

Joint angles are directly modulated by CPG outputs through:

$$\theta_i(t)=A\cdot s_i(t) \tag{2}$$

Where A is amplitude scaling factor determining maximum joint deflection, s_i is CPG firing rate for segment i ranging between -1 and 1, and θ_i is the resulting joint angle command in radians. **Equation 2 CPG to joint angle mapping** enables the remarkable simplicity of CPG-based implementations no complex inverse kinematics computation is required. The CPG output naturally encodes both the magnitude (through firing rate) and temporal dynamics necessary for smooth locomotion.

Dynamic models incorporating acceleration, force interactions, and contact mechanics are governed by (Park et al., 2020):

$$M(q)\ddot{q}+C(q,\dot{q})\dot{q}+G(q)=\tau+J_c^T F_c \quad (3)$$

Where $M(q)$ is the $n\times n$ mass matrix, C captures Coriolis and centripetal effects, G represents gravitational forces, τ are applied joint torques from actuators, J_c is the $m\times n$ contact Jacobian relating contact forces to joint torques, and F_c are contact forces including friction and normal forces. **Equation 3 Lagrangian dynamics model** becomes critical when the robot interacts with obstacles, terrain irregularities, or confined spaces. Dynamic computation requires solving these coupled differential equations numerically, typically taking 100-1000 milliseconds per step on standard processors. For real-time control at 10-100 Hz update rates, this is computationally prohibitive. Most deployed controllers employ kinematic frameworks augmented with simplified dynamics accounting for inertial effects and friction through empirically-tuned correction terms.

2.3 Biological Central Pattern Generators

Biological snakes generate locomotion through spinal Central Pattern Generators distributed neural circuits arranged segmentally along the spinal cord, capable of producing rhythmic patterns without continuous sensory input from higher brain centers (Liu, Gasoto, Jiang, et al., 2020). These circuits contain interconnected neurons with specific connectivity patterns enabling oscillatory activity.

Sensory feedback modulates these intrinsic patterns through proprioceptive information from muscle spindles and Golgi tendon organs providing body configuration and tension information, contact sensors detecting environmental interaction enabling obstacle-aided locomotion, and vestibular information from the inner ear providing gravity and inertial cues essential during terrain transitions. Higher brain regions, particularly the brainstem, send descending commands to spinal CPGs controlling speed through frequency modulation where increased descending input raises oscillator frequency, direction through asymmetric CPG activation biasing left-right oscillator phases, and gait selection where different neural pathways activate distinct CPG subnetworks. The remarkable feature of biological CPGs is their autonomy isolated spinal cord preparations without brain or sensory input generate coordinated rhythmic patterns nearly indistinguishable from intact animal locomotion.

3. CENTRAL PATTERN GENERATOR-BASED CONTROL

3.1 Mathematical Models for CPG Implementation

CPGs emerged as a robotics paradigm following landmark discoveries that isolated neural circuits from Tritonia and lamprey could generate coordinated motor patterns (Liu et al., 2021). These discoveries demonstrated that rhythm generation is an intrinsic property of neural circuits, not merely an artifact of higher brain function. The biological CPG concept translates to robotic implementations through mathematical models capturing key features: autonomous rhythm generation without external input, phase coupling enabling coordination between oscillators, frequency modulation for speed control, and spatial coordination producing organized patterns. Linear chain CPG networks arrange oscillators segmentally, with reciprocal inhibition between antagonistic pairs generating anti-phase outputs, as shown in Figure 2. Phase differences between adjacent segments are controlled through synaptic delay and coupling strength, enabling flexible gait adjustment from slow undulation to rapid sidewinding by modulating these parameters (Liu, Gasoto, Onal, et al., 2020).

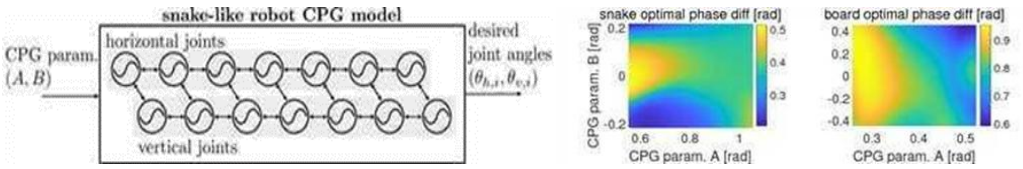

Figure 2. Linear chain CPG network with segmentally organized coupled oscillators. Reciprocal inhibition creates anti-phase outputs for opposing joint motion. Phase difference controlled through synaptic delay and coupling strength.

Matsuoka Oscillators represent the most widely employed CPG model in snake robot control due to their biological plausibility and implementation simplicity. Comprising reciprocally inhibited neuron pairs:

$$\dot{y}i=-y_i-\sum_j w\,ijs_j+u_i \tag{4}$$

$$\dot{v}_i=-v_i-y_i \tag{5}$$

$$s_i=\max(0,v_i) \tag{6}$$

Equation 4 Matsuoka membrane potential dynamics where y_i is the internal membrane potential integrating synaptic inputs and decay. **Equation 5 Adaptation dynamics** where v_i represents adaptation or fatigue. **Equation 6 Output firing rate** where s_i is the output firing rate clipped at zero. The parameters include w_{ij} as synaptic weights encoding both connection strength and sign, and u_i as constant tonic input providing baseline excitation. The mutual inhibition between reciprocal neuron pairs automatically generates oscillatory behavior as one neuron fires, it inhibits its partner, which then recovers and inhibits the first, creating a natural oscillation cycle.

Oscillation frequency increases linearly with input current u_i, enabling elegant speed control through simple scalar modulation increasing u_i by 10% increases locomotion frequency by approximately 10% (Liu, Gasoto, Onal, et al., 2020). Matsuoka oscillators offer biological plausibility with direct correspondence to identified neural circuits in lamprey and other vertebrates, demonstrated by detailed single-cell recordings showing similar temporal dynamics. They enable simple computation with complexity $O(N)$ where N is the number of oscillators, permitting implementation on microcontrollers with kilohertz update rates. The model includes inherent output saturation through the max operation preventing unrealistic joint commands exceeding mechanical limits. Documented robustness in biological systems shows that oscillations persist despite 10-30% parameter variations, enabling tolerance to component tolerances in robotic implementations and unmodeled variations in real systems.

Parameter tuning complexity remains a significant disadvantage determining appropriate values for w_{ij}, adaptation time constants, and u_i requires either extensive manual tuning through empirical testing or automated optimization (Tamura & Kamegawa, 2023). The coupling strength w_{ij} determines phase relationships; stronger coupling creates tighter phase relationships while weaker coupling produces more independent oscillations. At robotic timescales, time constants must be reduced from biological values to 10-100 milliseconds for responsive control, sacrificing

strict biological fidelity but enabling faster adaptation to terrain changes. A 16-segment snake requires tuning 64+ parameters across the network, creating a high-dimensional optimization problem.

Hopf Oscillators provide alternative frameworks based on bifurcation theory, guaranteeing stable limit-cycle oscillations (Zhang et al., 2017):

$$\dot{r}=\mu r\,(r_0^2-r^2) \tag{7}$$

$$\dot{\theta}=\omega \tag{8}$$

Equation 7 Hopf amplitude dynamics where the amplitude r exponentially converges to target amplitude r_0 regardless of initial conditions, ensuring stable operation—a mathematical guarantee absent in Matsuoka models where stability depends on parameter values. **Equation 8 Phase dynamics** where oscillation frequency is controlled by ω. Coupled Hopf oscillators can generate multi-frequency synchronized patterns through phase coupling. Computational advantages include explicit analytical solutions for certain coupling topologies, enabling predictive analysis of the coupled network behavior and stability guarantees through eigenvalue analysis of the linearized system. However, Hopf oscillators lack biological interpretability the bifurcation mechanism has no clear neural correlate, and the equations model mathematical attractors rather than neural dynamics. Amplitude and frequency control require separate mechanisms; the intuitive modulation that makes Matsuoka elegant is absent.

Kuramoto Oscillators represent coupled phase oscillators with pure sinusoidal coupling:

$$\dot{\theta}i=\omega_i+\sum_j K\,ij\sin(\theta_j-\theta_i) \tag{9}$$

Equation 9 Kuramoto coupling model where θ_i is the phase of oscillator i,ω_i is the natural frequency, and K_{ij} is the coupling strength. This model is mathematically elegant enabling analysis through synchronization theory and order parameters from statistical mechanics (Mu et al., 2025). Phase coordination can be computed explicitly through eigenvalue decomposition of the coupling matrix. However, Kuramoto oscillators lack biological motivation their mathematical form doesn't correspond to actual neural equations and amplitude control is entirely absent, requiring separate mechanisms. Their primary utility is analytical understanding of synchronization phenomena rather than implementation.

3.2 CPG Network Architectures and Coupling Schemes

CPG network architectures organize oscillators in linear chains mirroring spinal cord organization. Coupling schemes employ reciprocal inhibition producing anti-phase outputs for opposing joints, excitatory coupling with delays generating propagating waves with phase differences controlled by (Bing et al., 2017):

$$\phi=\arctan\left(\frac{\omega\tau}{1+K}\right) \tag{10}$$

Equation 10 Phase difference relationship where ϕ is the phase difference between adjacent oscillators, ω is oscillation frequency, τ is the synaptic delay and K is the coupling strength magnitude. This relationship enables precise control of phase differences through parameter adjustment. Gap junction coupling for

synchronized patterns represents an alternative approach using bidirectional electrical coupling.

3.3 Gait Generation and Parameter Optimization

Serpentine motion emerges naturally from chains with phase differences between 1.5 to 3 body waves, directly controlling gait characteristics through phase relationships (Bing et al., 2017). Parameter optimization employs genetic algorithms searching high-dimensional fitness landscapes for maximum locomotion speed and energy efficiency, particle swarm optimization coordinating population-based refinement, and symbiotic organism search incorporating biological cooperation principles. CPG limitations include parameter sensitivity requiring extensive tuning or optimization (Tamura & Kamegawa, 2023), limited adaptation to unknown environments without sensory feedback integration, and difficulty generating non-rhythmic behaviors such as precision manipulation or rapid directional changes.

4. LEARNING-BASED APPROACHES

4.1 Reinforcement Learning for Locomotion Control

Reinforcement Learning formulates locomotion as a Markov Decision Process where agents learn policies (control mappings) that maximize cumulative expected rewards (Qiu et al., 2021). State space comprises joint angles from encoders and velocities from differentiation, providing position and velocity feedback action space is continuous joint torques or position commands in the range $[-\tau_{max},\tau_{max}]$ for each joint; rewards combine multiple objectives as $R(t)=r_s+r_e+r_c$ where $r_s=v_x$ is forward progress incentive rewarding velocity in the desired direction, $r_e=-\alpha\sum_i|\tau_i|$ is energy efficiency penalty penalizing high torques, and r_c includes terrain-specific bonuses rewarding progress on difficult terrain. This multi-objective reward formulation enables agents to discover policies balancing competing objectives faster locomotion versus lower energy consumption.

Policy gradient methods directly optimize policy parameters through gradient ascent on the expected reward objective (Liu et al., 2023b):

$$\nabla_\theta J(\theta)=E\left[\nabla_\theta \log \pi_\theta(a|s) Q^\pi(s,a)\right] \tag{11}$$

Equation 11 Policy gradient formula where θ are policy parameters, $\pi_\theta(a|s)$ is the probability of action a given state s, *and* $Q^\pi(s,a)$ is the expected return from that action. The gradient points toward actions that yielded high rewards and away from actions that yielded low rewards. This elegant formulation enables on-policy learning the agent learns from experiences generated under its current policy.

Proximal Policy Optimization achieves dramatically faster convergence through clipped objective functions that limit policy updates (Jia & Ma, 2022):

$$L^{CLIP}(\theta)=E\left[\min\left(r_t(\theta)\widehat{A}_t, clip(r_t(\theta),1-\epsilon,1+\epsilon)\widehat{A}_t\right)\right] \tag{12}$$

Equation 12 PPO clipped objective where $r_t(\theta)=\pi_\theta(a_t|s_t)/\pi_{\theta_{old}}(a_t|s_t)$ is the probability ratio comparing new and old policies, $\widehat{A}_t$ is the estimated advantage (how much better this action is than average), and ϵ controls the clipping range. This prevents the policy from changing too dramatically in a single update, stabilizing learning. PPO achieves 10-100x faster convergence than policy gradient and value iteration methods, enabling practical application to snake robot control with

typical training requiring 50,000-200,000 simulation steps or 500-2000 real robot episodes depending on task complexity.

Deep Deterministic Policy Gradient (DDPG) extends continuous control through actor-critic architectures where an actor network $\pi_\theta(s)$ generates continuous actions directly and a critic network $Q_\phi(s,a)$ estimates the value of state-action pairs (Liu et al., 2021). This enables learning in 20+ dimensional action spaces typical of multi-segment snakes with simultaneous control of 8-16 joints. The critic guides the actor toward valuable actions through gradient. Sample efficiency remains the primary limitation learning from scratch requires millions of environment interactions. Domain randomization addresses this by training on distributions of simulated environments with friction coefficients varying ±50%, actuator delays 0-100 ms, sensor noise varying 2-5x baseline, mass properties ±20%, and terrain roughness variations (Ji et al., 2023). Policies trained on diverse simulators transfer better to real robots despite performing worse on individual training environments. Curriculum learning progressively increases task difficulty; early training on flat surfaces followed by gradual terrain complexity improves learning stability and final performance by 20-30% compared to training only on target terrain.

4.2 Evolutionary Algorithms and Hybrid CPG-Learning Architectures

Evolutionary Algorithms employ genetic algorithms mutating and recombining candidate solutions (Tamura & Kamegawa, 2023), particle swarm optimization coordinating population-based exploration, and symbiotic organism search incorporating mutualistic principles (Løwer et al., 2023). These methods excel at multi-objective optimization through Pareto dominance concepts but require significant computational resources.

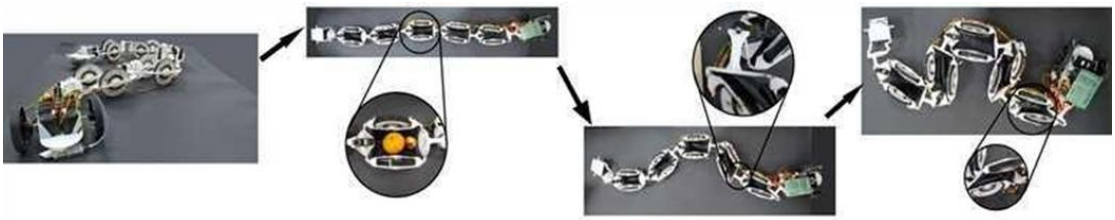

Figure 3. Fitness convergence curves showing learning efficiency. Hybrid CPG-learning achieves 90% of final performance with 100-1000x fewer samples than pure learning through structured search space exploitation.

Hybrid CPG-Learning Architectures integrate CPG structure with learned parameters, dramatically reducing learning dimensionality. As shown in Figure 3 hybrid architectures combine CPG's real-time feasibility with learned parameter modulation (Liu et al., 2023a). CPG-ACTOR reduces parameter count from thousands (pure neural network) to tens (CPG + modulation layers), achieving 6x faster training while maintaining learning benefits (Liu et al., 2025). This dimensionality reduction dramatically improves sample efficiency (10^3-10^4 samples vs 10^6-10^7 for pure learning). Contact-aware learning incorporates tactile feedback through learned feedback gains modulating CPG parameters (Zhao et al., 2022), achieving 20-40% performance improvement on variable terrain while maintaining real-time feasibility through simple linear feedback rules rather than complex neural network inference.

4.3 Neuromorphic Approaches

Neuromorphic Approaches employ spiking neural networks simulating biological neurons

through spike timing (Zhang et al., 2025), enabling neuromorphic hardware implementation on platforms like Intel Loihi with 100x energy reduction compared to digital controllers. Spiking neural networks encode information as temporal patterns rather than firing rates, enabling asynchronous event-driven computation with intrinsic parallelism.

5. COMPARATIVE ANALYSIS AND PERFORMANCE EVALUATION

5.1 Performance Metrics and Evaluation Framework

Performance evaluation requires careful definition of metrics (Seeja et al., 2022). Locomotion speed is typically normalized to body-lengths per second to enable comparison across robots of different sizes a 0.8 m/s snake 1 meter long achieves 0.8 BL/s while a 0.4 m/s snake 0.5 meters long also achieves 0.8 BL/s despite different absolute speeds. Energy efficiency is quantified through cost of transport, a dimensionless metric enabling comparison across scales. Stability is measured through variance in heading angle during steady locomotion lower variance indicates more stable paths. Terrain adaptability is quantified through success rate across different surfaces.

Table 1 Comparative Analysis of Control Methods presents comprehensive analysis of control methodologies across critical performance dimensions (Shi et al., 2025). CPG methods execute in 1-10 milliseconds per cycle on microcontrollers, suitable for small embedded systems. Achieved locomotion speeds range 0.3-0.8 body-lengths per second on rigid platforms. Energy efficiency ranges 2.0-3.5 for typical CPG implementations. CPG methods demonstrate exceptional robustness across parameter variations; 10-30% parameter perturbations cause negligible performance degradation.

Table 1. Comparative Analysis of Control Methods

Characteristic	**CPG-Based**	**Learning-Based**	**Hybrid CPG-Learning**
Real-time Feasibility	High (1-10 ms)	Low (50+ ms)	High (10-20 ms)
Parameter Tuning	Manual, difficult	Automatic	Automatic
Sample Efficiency	N/A	Low (~10^6)	High (~10^3-10^4)
Robustness	Moderate-High	Varies	High
Speed (BL/s)	0.3-0.8	0.5-1.2	0.6-1.0
Energy (CoT)	2.0-3.5	1.5-2.5	1.8-2.8

Learning methods achieve 0.5-1.2 body-lengths per second through automatic gait discovery optimizing locomotion for simulation environment dynamics (Qiu et al., 2021). On simulation, learning methods surpass hand-tuned CPGs by 20-50% depending on terrain due to discovering gaits specifically optimized for that environment's friction properties and terrain characteristics. However, sim-to-real transfer reduces this advantage significantly real-world performance

typically 20-30% below simulation, whereas CPG methods transfer with <10% degradation (Ji et al., 2023). The performance gap occurs because learning optimizes for simulator dynamics that don't perfectly match reality. Learning methods require inference during deployment; typical neural network inference time 5-50 milliseconds on embedded GPU (NVIDIA Jetson Nano), incompatible with real-time control in fast-changing environments requiring 10 Hz minimum update rates. This computational requirement explains why learning-based controllers are typically relegated to offline motion planning or vision-based high-level decision making rather than real-time feedback control.

5.2 Experimental Results across Platforms and Terrains

Hybrid CPG-learning approaches achieve 0.6-1.0 body-lengths per second while maintaining real-time feasibility and 30-50% faster training convergence through dimensionality reduction. Performance varies substantially across terrain types, as shown in Figure 4 experimental results demonstrate that hybrid CPG-learning methods achieve superior balance between speed, energy efficiency, and adaptability compared to pure CPG or pure learning approaches (Shi et al., 2025). The loaded snake robot (LSR1) successfully handles complex terrain including grass with vegetation resistance and uneven surfaces, validating the practical effectiveness of the control approach (Liu, Gasoto, Onal, et al., 2020). Hybrid methods consistently exceed 85% success rate across diverse terrain types through contact-aware modulation.

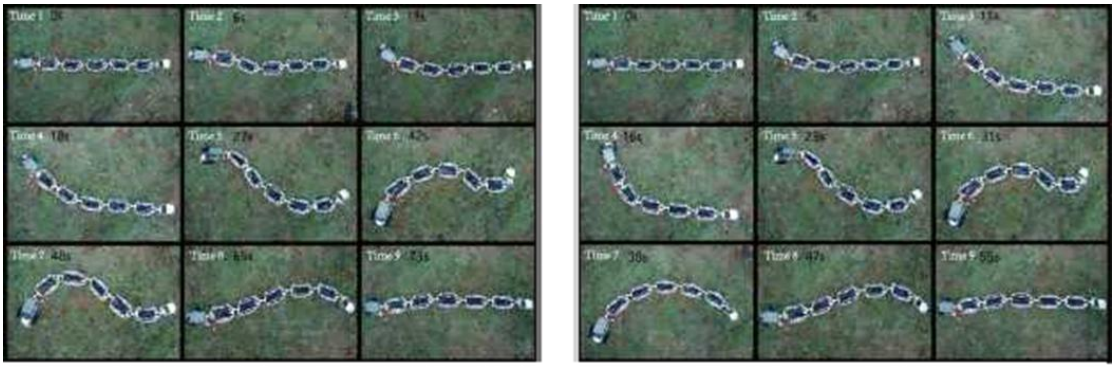

Figure 4. *Loaded snake robot (LSR1) performance comparison before and after CPG parameter optimization showing 25% reduction in motion time across grass and uneven terrain.*

Table 2 Performance Summary Across Terrain Types and Platforms provides comprehensive performance metrics comparing CPG, learning-based, and hybrid methods across five distinct terrain environments. The data demonstrates that hybrid approaches consistently achieve superior performance across all terrain types, from rigid laboratory floors to underwater environments.

Table 2.Performance Summary across Terrain Types and Platforms

Terrain Type	CPG Success	Learning Success	Hybrid Success	Preferred Method
Rigid Floor	95%	98%	99%	Hybrid
Granular (Sand)	65%	72%	85%	Hybrid
Rocky/Uneven	60%	75%	82%	Hybrid
Confined Pipes	90%	85%	92%	Hybrid
Underwater	88%	92%	95%	Hybrid

On rigid laboratory floors, CPG controllers achieve smooth, periodic locomotion with minimal computational overhead. On sandy and granular substrates where individual contacts vary unpredictably, contact-aware CPG modulation improves performance 15-25% through sensory feedback (Zhao et al., 2022). Soft robot platforms benefit dramatically from learning due to complex actuator dynamics (Lu et al., 2024) soft robot learning often outperforms CPG approaches by 30-50%.

Industrial applications including inspection robots (ACM-R5) and search-and-rescue platforms employ hybrid or conservative CPG approaches prioritizing reliability (P. Ngamkajornwiwat & N. Pothita, 2024). The ACM-R5 subsea inspection platform achieves reliable 0.4 m/s locomotion despite highly variable environments. Medical applications employ tightly controlled CPG-based approaches ensuring predictable, safe behavior.

6. SENSORY INTEGRATION AND TERRAIN ADAPTATION

Sensory feedback is essential for robust locomotion in unstructured environments. As shown in Figure 5 the primary concertina locomotion pattern is achieved through modulating CPG parameters in response to sensory signals. This gait is particularly effective in constrained spaces such as pipelines where the robot leverages discrete contact points for propulsion. The concertina motion emerges from the same CPG network with different phase offsets and amplitude modulations compared to open-space gaits, demonstrating a unified control framework for diverse locomotion behaviors. Each gait exploits distinct environmental interactions: concertina leverages fixed contact points in confined spaces enabling efficient navigation through narrow passages (Li et al., 2019). Sensory feedback integration enables real-time gait selection and parameter adaptation through proprioceptive information from joint angles and motor currents providing body configuration, contact sensors detecting environmental interaction, and vestibular information providing orientation cues essential during terrain transitions.

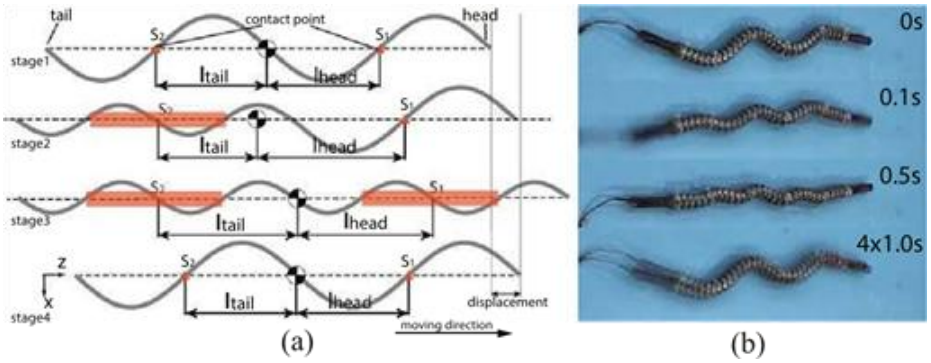

Figure 5. Concertina locomotion in pipeline showing body configuration at 4 motion stages (0s, 3s, 5s, 8s). Head extends while maintaining tail grip, creating forward progression through sequential contact point dynamics.

6.1 Feedback Mechanisms

Sensory feedback modulates CPG parameters through:

$$u_i(t)=u_0+\sum_k \alpha_k \cdot s_k^{sensor}(t) \tag{13}$$

Equation 13: Sensory feedback modulation where $u_i(t)$ is the modulated tonic input to oscillator i,u_0 is baseline input, α_k are sensory coupling gains, and s_k^{sensor} are normalized sensor signals. This multiplicative coupling enables proportional modulation while maintaining oscillatory dynamics.

6.2 Contact-Aware Locomotion Control

Contact-aware CPG control demonstrates 20-40% improvement in traversal success rates across uneven terrain (Zhao et al., 2022). Sensory modulation mechanisms include frequency adaptation where contact stimulation increases CPG oscillation frequency enabling faster escape, amplitude modulation where strong contact signals reduce movement amplitude enabling cautious navigation, and phase shifting enabling rapid directional correction. These mechanisms operate within biological CPG frameworks through simple multiplicative or additive modulation of oscillator parameters, maintaining real-time feasibility.

This sensory integration is particularly valuable for obstacle-aided locomotion where snakes leverage environmental features for propulsion (Lim et al., 2020). A snake encountering a vertical peg can press against it to gain leverage for forward progression. Tegotae-inspired mechanisms use contact feedback to trigger specific motor responses enabling energy-efficient obstacle negotiation. Perception-driven approaches employ forward-facing cameras to detect upcoming obstacles and plan trajectories leveraging obstacle positions.

7. RESEARCH GAPS AND CHALLENGES

7.1 Sim-to-Real Transfer Challenges

Sim-to-Real Transfer remains a critical barrier (Ji et al., 2023). Simulators employ simplified friction models while real robots exhibit velocity-dependent friction. Equations 14-15 describe the friction model mismatch:

$$f_{sim}(v)=\mu \cdot N \quad (constant) \tag{14}$$

$$f_{real}(v)=(\mu_s-\mu_k e^{-v/v_s})\cdot N \tag{15}$$

Equation 14 Simulator friction model assumes constant friction coefficient regardless of velocity. **Equation 15 Real robot friction model** where friction varies with velocity based on static friction coefficient μ_s, kinetic friction μ_k, and characteristic velocity v_s . This mismatch causes controllers optimized in simulation to perform poorly on real robots. Actuator responses are idealized while real servos exhibit delays and saturation (Badran et al., 2020). Domain randomization improves transfer success rates from 30-40% to 60-80% at cost of reduced target environment performance.

7.2 Energy Efficiency Limitations

Energy Efficiency represents a critical practical limitation (Baysal & Altas, 2020). Biological snakes achieve cost of transport 0.5-1.0 while robotic systems achieve 2-5. Cost of transport is calculated using:

$$CoT=\frac{E_{total}}{m\cdot g\cdot d} \tag{16}$$

Equation 16 Cost of transport formula where E_{total} is total energy consumed, m is body mass,$g$is gravitational acceleration, and d is distance traveled. This dimensionless metric enables comparison across robots of different sizes and masses. Optimization strategies include gearbox redesign for steady-state efficiency, pulse-width modulation reducing energy during posture maintenance, and compliant joints reducing impact losses. Multi-objective optimization could achieve 30-50% energy reduction.

7.3 Real-Time Sensory Integration

Real-Time Sensory Integration requires embedding sensory processing and control within 10-100 millisecond cycles (Seeja et al., 2022). Simple feedback rules integrate in 1-5 ms but lack sophistication. Learning-based sensory integration requires 5-20 ms neural network inference on embedded systems, approaching cycle time budgets. Neuromorphic sensory processing using spiking neural networks promises latency reduction to 1-2 ms through asynchronous event-driven computation (Zhang et al., 2025).

7.4 Standardization Issues

Standardization is critically lacking. ***Table 3*** summarizes research gaps and mitigation strategies. No agreed-upon benchmarks exist; different papers employ different simulation environments, robot specifications, and evaluation metrics. Creating standardized benchmarks would accelerate progress. The International Organization for Standardization (ISO) has begun standardization efforts, but consensus remains elusive due to diverse application requirements.

Table 3. Critical Research Gaps and Mitigation Strategies

Research Gap	Current Challenge	ChallengeProposed Solution	Expected Impact
Sim-to-Real	20-30% drop	Domain randomization	Improve success to 80%+
Energy Efficiency	2-5 CoT	Multi-objective optimization	Achieve 1.5-2.0 CoT
Sensory Integration	5-50 ms latency	Neuromorphic hardware	Reduce to <2 ms
Standardization	Incomparable	Unified benchmarks	Enable meta-analysis
Verification	Safety proving	Formal methods	Enable medical apps

9. FUTURE DIRECTIONS

Future research should prioritize establishing standardized benchmarking platforms enabling reproducible comparative evaluation across methods and platforms. Creating unified benchmark suites analogous to ImageNet in computer vision would accelerate progress through enabling meaningful meta-analysis and reproducible comparisons. Sim-to-real transfer techniques must reduce domain gap through careful system identification characterizing actual friction, actuator delays, and contact dynamics specific to each platform. Energy-efficient controller design through multi-objective optimization can achieve 30-50% efficiency improvements by balancing speed, power consumption, and stability within formalized Pareto optimization frameworks. Unified frameworks integrating CPGs with learning should exploit structured search spaces while maintaining learning benefits, enabling scalable solutions across robot morphologies and application domains. Neuromorphic implementations on spiking hardware promise 100x energy reduction through asynchronous event-driven computation, enabling truly autonomous operation in power-limited scenarios.

Bio-hybrid approaches interfacing biological neural tissues with robotic actuators offer radically different control paradigms not yet fully explored. Lamprey spinal cord segments coupled to robotic bodies demonstrate proof-of-concept; scaling to full-bodied bio-hybrid systems remains an open challenge requiring advances in interfacing, control integration, and biological sustainability. These systems leverage biological neural plasticity and adaptation while providing robotic mobility, potentially unlocking control capabilities currently impossible with artificial systems. The convergence of neuroscience and robotics through bio-hybrid systems represents a frontier with profound implications for understanding nervous system function and developing truly adaptive robotic systems.

Snake robot research will increasingly focus on embodied AI principles where control emerges from agent-environment interaction rather than explicit programming. This paradigm shift requires rethinking how we design learning algorithms, incorporate biological inspiration, and evaluate robot performance. As hardware matures and control methods advance, snake robots will transition from laboratory platforms to practical tools with broad

deployment. Applications will expand from current domains (disaster response, infrastructure inspection, environmental exploration) to medical robotics, space exploration (traversing asteroid surfaces and planetary caves), and underwater research.

The integration of advancing sensors (fiber-optic gyroscopes providing inertial measurement without drift, distributed pressure sensors enabling obstacle detection, bio-inspired flow sensors enabling ambient awareness), improved actuators (artificial muscles providing compliant actuation, electroactive polymers enabling programmable stiffness), and sophisticated controllers combining biological inspiration with machine learning positions snake robots as a transformative platform. This platform will serve dual purposes: advancing our understanding of how biological systems achieve robust locomotion across complex environments, and developing adaptive robotic systems capable of autonomous operation in unstructured, human-environments where conventional platforms fail.

8. CONCLUSIONS

Snake robot locomotion control has matured through two complementary paradigms. CPG-based methods offer biological inspiration, real-time feasibility on embedded systems, and interpretability through direct neural correspondence. Learning-based approaches enable automatic gait discovery without manual tuning, excel at multi-objective optimization balancing speed and efficiency, and adapt online to unknown environments. Hybrid architectures combining CPG structure with learned parameters represent the most promising direction, balancing biological plausibility with adaptive capability while maintaining computational efficiency and real-time performance.

The comparative analysis across platforms and terrains reveals that no single methodology dominates across all application domains. CPG-based controllers excel in resource-constrained embedded systems and safety-critical applications where interpretability and robustness are paramount. Learning-based approaches maximize performance when computational resources are available and sim-to-real transfer can be properly addressed through domain randomization and careful system identification. Hybrid CPG-learning architectures emerge as the optimal choice for most practical applications, leveraging CPG's real-time feasibility and biological structure while incorporating learning's adaptive capabilities and multi-objective optimization.

Critical research gaps identified throughout this review sim-to-real transfer challenges, energy efficiency limitations, real-time sensory integration constraints, and standardization fragmentation establish the agenda for advancing the field. Addressing these gaps requires not incremental improvements but fundamental advances in how we conceptualize the integration of neural structures, learning mechanisms, and embodied robot morphologies. The field stands at an inflection point where mature understanding of individual methodologies enables integration into more sophisticated systems.

REFERENCES

Badran, M. A., Khan, Md. R., & Toha, S. F. (2020). Implementation of Motion Algorithm on a Snake Robot Prototype for Serpentine Locomotion. *2020 IEEE 8th Conference on Systems, Process and Control (ICSPC)*, 152–157. https://doi.org/10.1109/ICSPC50992.2020.9305750

Baysal, Y. A., & Altas, I. H. (2020). Optimally Efficient Locomotion of Snake Robot. *2020 International Conference on INnovations in Intelligent SysTems and Applications (INISTA)*, 1–6. https://doi.org/10.1109/INISTA49547.2020.9194621

Bing, Z., Cheng, L., Chen, G., Röhrbein, F., Huang, K., & Knoll, A. (2017). Towards autonomous locomotion: CPG-based control of smooth 3D slithering gait transition of a snake-like robot. *Bioinspiration & Biomimetics*, *12*(3), 035001. https://doi.org/10.1088/1748-3190/aa644c

Campanaro, L., Gangapurwala, S., Martini, D. D., Merkt, W., & Havoutis, I. (2021). *CPG-ACTOR: Reinforcement Learning for Central Pattern Generators* (arXiv:2102.12891). arXiv. https://doi.org/10.48550/arXiv.2102.12891

Fukuoka, Y., Otaka, K., Takeuchi, R., Shigemori, K., & Inoue, K. (2023). Mechanical Designs for Field Undulatory Locomotion by a Wheeled Snake-Like

Robot With Decoupled Neural Oscillators. *IEEE Transactions on Robotics*, *39*(2), 959–977. https://doi.org/10.1109/TRO.2022.3226364

Ji, Z., Song, G., Wang, F., Li, Y., & Song, A. (2023). Design and Control of a Snake Robot With a Gripper for Inspection and Maintenance in Narrow Spaces. *IEEE Robotics and Automation Letters*, *8*(5), 3086–3093. https://doi.org/10.1109/LRA.2023.3265591

Jia, Y., & Ma, S. (2022). Distributed Coach-Based Reinforcement Learning Controller for Snake Robot Locomotion. *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1231–1238. https://doi.org/10.1109/IROS47612.2022.9981749

Li, P., Sun, Y., Yang, Y., Li, C., Pu, H., Luo, J., & Xie, S. (2019). Design and Modeling an Elongatable Robotic Snake towards Augmented Serpentine Gait. *Robotics and Mechatronics*.

Lim, J., Yang, W., Shen, Y., & Yi, J. (2020). Analysis and Validation of Serpentine Locomotion Dynamics of a Wheeled Snake Robot Moving on Varied Sloped Environments. *2020 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, 1069–1074. https://doi.org/10.1109/AIM43001.2020.9158974

Liu, X., Gasoto, R., Jiang, Z., Onal, C., & Fu, J. (2020). Learning to Locomote with Artificial Neural-Network and CPG-based Control in a Soft Snake Robot. *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 7758–7765. https://doi.org/10.1109/IROS45743.2020.9340763

Liu, X., Gasoto, R., Onal, C., & Fu, J. (2020). *Learning to Locomote with Deep Neural-Network and CPG-based Control in a Soft Snake Robot* (arXiv:2001.04059). arXiv. https://doi.org/10.48550/arXiv.2001.04059

Liu, X., Onal, C. D., & Fu, J. (2023a). *Integrating Contact-aware Feedback CPG System for Learning-based Soft Snake Robot Locomotion Controllers* (arXiv:2309.02781). arXiv. https://doi.org/10.48550/arXiv.2309.02781

Liu, X., Onal, C. D., & Fu, J. (2023b). Reinforcement Learning of CPG-Regulated Locomotion Controller for a Soft Snake Robot. *IEEE Transactions on Robotics*, *39*(5), 3382–3401. https://doi.org/10.1109/TRO.2023.3286046

Liu, X., Onal, C. D., & Fu, J. (2025). Integrating Contact-Aware CPG System for Learning-Based Soft Snake Robot Locomotion Controllers. *IEEE Transactions on Robotics*, *41*, 1581–1601. https://doi.org/10.1109/TRO.2025.3539173

Liu, X., Onal, C., & Fu, J. (2021). *Learning Contact-aware CPG-based Locomotion in a Soft Snake Robot* (arXiv:2105.04608). arXiv. https://doi.org/10.48550/arXiv.2105.04608

Løwer, J., Gravdahl, I., Varagnolo, D., & Stavdahl, Ø. (2023). Form Closure for Fully Actuated and Robust Obstacle-Aided Locomotion in Snake Robots. *IEEE Robotics and Automation Letters*, *8*(11), 7360–7367. https://doi.org/10.1109/LRA.2023.3316912

Lu, J., Hu, X., Zhu, X., Chen, C., & Wang, H. (2024). Adaptive Soft Pneumatic Robot Inspired by Locomotion of Snake. *2024 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 713–718. https://doi.org/10.1109/ROBIO64047.2024.10907334

Mu, J., Zong, X., Hou, X., & Zhang, Y. (2025). Design and Optimization of Gait Transition Control for Loaded Snake Robots Using CPG Algorithms. *2025 3rd International Conference on Control and Robot Technology (ICCRT)*, 1–7. https://doi.org/10.1109/ICCRT63554.2025.11072731

P. Ngamkajornwiwat, & N. Pothita. (2024). Investigating Adaptive CPG-based Control of a Snake Robot with Switch Signal Input for Maneuvering in Varying Environments. *Journal of Research and Applications in Mechanical Engineering (JRAME)*, *12*, JRAME24. https://doi.org/10.14456/JRAME.2024.23

Park, S., Lee, H., & Lee, D. (2020). Robust Motion Control of Robotic Systems with Environmental Interaction via Data-Driven Inversion of CPG. *2020 20th International Conference on Control, Automation and Systems (ICCAS)*, 692–698. https://doi.org/10.23919/ICCAS50221.2020.9268377

Qiu, K., Zhang, H., Lv, Y., Wang, Y., Zhou, C., & Xiong, R. (2021). Reinforcement Learning of Serpentine Locomotion for a Snake Robot. *2021 IEEE International Conference on Real-Time*

Computing and Robotics (RCAR), 468–473. https://doi.org/10.1109/RCAR52367.2021.9517436

Seeja, G., Selvakumar Arockia Doss, A., & Hency, V. B. (2022). A Survey on Snake Robot Locomotion. *IEEE Access*, *10*, 112100–112116. https://doi.org/10.1109/ACCESS.2022.3215162

Shi, H., Meng, Y., Cui, W., Rao, M., Wang, S., & Xie, Y. (2025). Biomimetic Underwater Soft Snake Robot: Self-Motion Sensing and Online Gait Control. *IEEE Transactions on Robotics*, *41*, 1193–1210. https://doi.org/10.1109/TRO.2025.3530349

Tamura, H., & Kamegawa, T. (2023). Parameter search of a CPG network using a genetic algorithm for a snake robot with tactile sensors moving on a soft floor. *Frontiers in Robotics and AI*, *10*, 1138019. https://doi.org/10.3389/frobt.2023.1138019

Zhang, C., Wang, C., Pan, W., & Santina, C. D. (2025). SpikingSoft: A Spiking Neuron Controller for Bio-inspired Locomotion with Soft Snake Robots. *2025 IEEE 8th International Conference on Soft Robotics (RoboSoft)*, 1–8. https://doi.org/10.1109/RoboSoft63089.2025.11020907

Zhang, D., Xiao, Q., Cao, Z., Huang, R., & Fu, Y. (2017). Smooth transition of the CPG-based controller for snake-like robots. *2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2716–2721. https://doi.org/10.1109/ROBIO.2017.8324830

Zhao, W., Wang, J., & Fei, Y. (2022). A Multigait Continuous Flexible Snake Robot for Locomotion in Complex Terrain. *IEEE/ASME Transactions on Mechatronics*, *27*(5), 3751–3761. https://doi.org/10.1109/TMECH.2021.3131766