Swarm-Based Coordination Architecture for Humanoid Robots: A Distributed Multi-Agent Framework with Secure Rule Evolution

Aslan Alwi*, Munirah, Almudaya Research Institute Team
aslan.alwi@umpo.ac.id, munirah@umpo.ac.id

Independent Researcher at Almudaya Research Institute
Department of Informatics Engineering, Faculty of Engineering, Universitas Muhammadiyah Ponorogo, Indonesia

November 2025

Abstract

Humanoid robots require scalable, adaptive, and fault-tolerant coordination mechanisms to manage the high dimensionality and interdependence of their joint systems. Building on our previous work on distributed multi-agent control with secure blackboard-based coordination, this paper introduces a swarm-based architecture that models the humanoid robot as a collection of interacting joint-agents governed by emergent swarm rules. Each joint operates as an autonomous agent with local perception and actuation, while global motion is achieved through decentralized behaviors such as alignment, cohesion, and stability-seeking interactions.

We propose a hierarchical swarm-control framework that separates rule execution at the joint level from rule evolution at the coordination layer. A secure rule evolution mechanism, inspired by lightweight blockchain validation, ensures that modifications to swarm parameters—such as alignment weights or neighborhood influence scopes—are consistent, safe, and cryptographically verifiable. This enables online adaptation while preventing unsafe or malicious updates to the robot's coordination strategy.

The architecture is validated through simulated experiments demonstrating that stable whole-body behavior can emerge from local swarm rules without requiring centralized trajectory optimization. Results show improved robustness against joint disturbances, faster adaptation to configuration changes, and natural scalability as the number of joints increases. The proposed framework establishes a foundation for future humanoid control systems where autonomy arises from the cooperative dynamics of swarm intelligence under secure, auditable coordination protocols.

Keywords: Swarm Robotics; Humanoid Control; Distributed Multi-Agent Systems; Blockchain Security; Emergent Coordination; Adaptive Rule Evolution; Decentralized Control; Joint-Level Intelligence.

1 Introduction

The growing complexity of humanoid robots—characterized by high degrees of freedom, multimodal perception, and dynamic interaction with human environments—necessitates new approaches to embodied intelligence that exceed the capabilities of classical control architectures. While traditional humanoid controllers rely on centralized optimization, symbolic planning, or deterministic whole-body solvers [7], recent advances in distributed control [4, 5] and multi-agent reinforcement learning for legged systems [8] have demonstrated the potential of decentralizing motor intelligence across many local controllers.

In the companion paper, humanoid robots were modeled as distributed multi-agent systems wherein each joint operates as a semi-autonomous agent, coordinated through an Epistemic Blackboard and secured via an internal blockchain layer. While this architecture ensures coherent and secure decision-making, it still presupposes that most forms of coordination arise from hierarchical or structured agent interactions.

In this second paper, we propose a more radical reconceptualization: viewing the humanoid robot as a *swarm of joint-level agents*. In this paradigm, each joint agent not only performs localized control but also participates in emergent coordination dynamics inspired by swarm intelligence models such as Reynolds' boids, ant colony systems, and decentralized biological motor coordination. Swarm behaviors have been successfully applied in multi-robot systems [1, 3], but their extension into the *internal morphology of a single humanoid robot* remains an unexplored research frontier.

This shift toward swarm-based humanoid control is motivated by several observations:

1. Biological motor systems exhibit emergent coordination arising from millions of semi-autonomous elements (motor neurons, muscle fibers, proprioceptive loops), suggesting that whole-body coherence can arise from local rules.

2. Reflex-based distributed controllers in humanoid robots [2] demonstrate that stability and adaptation can emerge without globally synchronized planning.

3. Reinforcement learning for whole-body humanoid skills [6,9] increasingly favors distributed, modular control structures rather than monolithic policies.

4. Swarm-based algorithms offer robustness, adaptability, and graceful degradation—properties desirable for physical systems interacting with unpredictable environments.

In the proposed architecture, the Epistemic Blackboard serves as a dynamic rule-space in which swarm coordination algorithms are written, modified, or evolved in real time. Joint-level agents read shared "swarm rules" from the blackboard and adapt their behaviors accordingly. These rules govern alignment, cohesion, avoidance, local torque modulation, reactive stabilization, and whole-body adaptation. Because all modifications to swarm rules are secured by the internal blockchain (as introduced in Paper 1), the robot maintains epistemic integrity even when swarm rules evolve dynamically.

The contributions of this paper are:

- A formalization of humanoid joints as a *swarm of local agents* whose collective interactions generate emergent whole-body motion.

- An extension of the Epistemic Blackboard to function as a *live swarm-rule substrate* enabling dynamic, real-time modification of coordination laws.
- A secure blockchain-mediated mechanism for validating swarm rule updates to ensure safety and stability during emergent behavior.
- A demonstration showing how emergent whole-body behaviors such as balance recovery, compliant manipulation, and exploration arise from distributed swarm interactions.

By integrating distributed epistemic control with swarm intelligence, this paper aims to lay the theoretical foundation for a new class of adaptive, self-organizing humanoid robots whose intelligence arises not solely from hierarchical planning, but from emergent embodied computation.

2 Background and Related Works

Swarm-based humanoid control integrates three major research domains: (1) swarm intelligence, (2) distributed and embodied humanoid control, and (3) shared-memory cognitive substrates such as blackboard architectures. This section reviews key literature from each domain and highlights how existing contributions motivate the proposed synthesis.

2.1 Swarm Intelligence and Emergent Coordination

Swarm intelligence studies how large populations of simple agents produce complex and coordinated behaviors through local interactions. Classical models include Reynolds' boids for flocking, ant colony systems for collective foraging, and distributed consensus rules in social insects. These systems demonstrate that global organization can emerge from localized rules without centralized control.

Although swarm intelligence has been applied extensively to multi-robot systems and distributed sensing networks, its application within the morphology of a *single robot* remains rare. However, prior works in reflex-based humanoid control [2] suggest that local behaviors can give rise to coordinated motor responses, providing a bridge between swarm principles and humanoid embodiment.

Swarm algorithms offer several desirable properties for humanoid control:

- robustness to individual agent failure,
- scalability with increasing numbers of degrees of freedom,
- adaptability through decentralized coordination,
- emergent stabilization and motor synergy.

These properties align with the needs of humanoid robots as they become more complex and operate in unstructured environments.

2.2 Distributed and Embodied Humanoid Control

Distributed control approaches have been increasingly adopted in modern humanoid robotics. Classical whole-body frameworks and operational space control [7] rely on structured optimization across all joints simultaneously. However, recent trends emphasize decentralization.

Distributed walking and balance controllers, such as those in Herdt et al. [4], demonstrate that locomotion can be achieved through multiple semi-independent controllers operating in coordination. Koolen et al. [5] further implemented a real-time distributed architecture for humanoid robots capable of whole-body tasks under computational constraints.

More recently, multi-agent reinforcement learning has been explored for legged locomotion [8], showing that decentralized controllers can learn coordinated behaviors without global optimization. These results indicate that humanoid control can emerge from multiple agents interacting through shared constraints, which aligns closely with swarm-based interpretations.

2.3 Epistemic Blackboard Architectures

Blackboard architectures provide a shared representational substrate through which distributed agents contribute partial knowledge or intermediate conclusions. Originating in early AI, blackboards have been adapted for robot navigation, sensor fusion, and planning.

The key advantage of a blackboard architecture is its ability to:

- unify knowledge across heterogeneous agents,
- support asynchronous contributions,
- maintain hierarchical and semantic coherence.

In the context of swarm-based humanoid control, the Epistemic Blackboard becomes a dynamic, real-time substrate where swarm coordination laws are:

1. written,
2. modified,
3. evolved, and
4. validated.

This design transforms the blackboard into a live "genomic space" for the swarm of joint agents.

2.4 Blockchain in Swarm and Multi-Agent Systems

Blockchain has been studied for securing multi-agent coordination, ensuring tamper resistance, and enabling mutual trust in decentralized systems. Afanasyev et al. [1] and Dorri et al. [3] describe blockchain mechanisms that enforce honesty and prevent malicious agents from manipulating shared information.

While existing works mostly focus on distributed robots or IoT systems, the integration of blockchain as an *internal* epistemic security layer for humanoid robots was introduced in the companion paper. In the context of swarm-based humanoid intelligence, blockchain validation ensures that modifications to swarm rules:

- cannot be spoofed by faulty agents,
- are authenticated cryptographically,
- are serialized consistently,
- preserve safety constraints.

This provides a novel foundation for "secure emergent intelligence," where swarm behaviors evolve dynamically yet remain safe under physical constraints.

2.5 Positioning the Proposed Work

Bringing these strands together, the proposed swarm-based humanoid architecture extends prior work by:

- applying swarm intelligence principles *within a single embodied humanoid system*,
- treating each joint or actuator as a member of an internal swarm,
- using an Epistemic Blackboard as a dynamic rule substrate for swarm coordination,
- leveraging blockchain to secure swarm-rule evolution in real time,
- enabling emergent whole-body behaviors without centralized optimization.

To the best of our knowledge, no prior framework unifies swarm intelligence, cognitive blackboard architectures, and blockchain-mediated epistemic security within a humanoid control paradigm.

3 Conceptual Architecture of Swarm-Based Humanoid Control

This section introduces the conceptual architecture through which a humanoid robot is modeled as a swarm of distributed joint-level agents. Each agent participates in emergent coordination governed by swarm rules dynamically stored in the Epistemic Blackboard and cryptographically validated through an internal blockchain layer.

The architecture integrates three tiers of interaction:

1. **Local Joint Agents**: each joint behaves as an autonomous swarm entity.
2. **Swarm-Rule Substrate**: the Epistemic Blackboard provides a shared rule-space.
3. **Secure Evolution Layer**: blockchain validation mediates rule evolution and ensures safety.

These components jointly enable emergent whole-body behaviors that are adaptable, fault-tolerant, and semantically grounded.

3.1 Joint-Level Agents as Swarm Entities

In contrast to hierarchical control in classical humanoid robotics [7], the proposed architecture reinterprets each joint agent as an autonomous swarm element with the following properties:

- **Local State** s_i: joint angle, velocity, torque, proprioception, tactile feedback.
- **Local Behavior Policy** π_i: a neural or rule-based controller incorporating swarm rules.
- **Neighborhood Perception**: each agent observes kinematic neighbors (e.g., parent/child joints).
- **Local Objectives**: stability, torque minimization, compliance, task alignment.
- **Blackboard Access**: read swarm rules + write local signals.

This parallels swarm robotics approaches in which each agent follows simple rules while contributing to global emergent behavior [1].

Formally, each joint-agent updates its control policy at time t as:

$$u_i(t) = \pi_i\big(s_i(t), \mathcal{N}_i(t), R(t)\big)$$

where:

- $\mathcal{N}_i(t)$ = set of neighbors of joint i,
- $R(t)$ = swarm rules at time t stored in the blackboard.

3.2 Swarm Rules: Alignment, Cohesion, Avoidance, and Task Coupling

Swarm coordination arises from local rules analogous to those found in biological systems (e.g., Reynolds' boids). We define four classes of rules:

3.2.1 1) Alignment Rule

Each joint aligns its local motion or torque trajectory with that of its neighbors:

$$r_{\text{align}} : \quad \Delta u_i \propto \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} \big(u_j - u_i\big)$$

This encourages fluid whole-body coordination.

3.2.2 2) Cohesion Rule

Joints converge toward a shared posture or center-of-motion reference:

$$r_{\text{cohesion}} : \quad \Delta s_i \propto \big(s_{\text{group}} - s_i\big)$$

This is analogous to "motor synergies" described in biological motor control literature.

3.2.3 3) Avoidance Rule

To prevent unsafe configurations:

$$r_{\text{avoid}}: \quad \Delta u_i \propto -\nabla V_{\text{safety}}(s_i)$$

This enforces joint limits, self-collision constraints, and stability envelopes.

3.2.4 4) Task-Coupling Rule

Task-level intent is broadcast through the blackboard and incorporated as:

$$r_{\text{task}}: \quad \Delta u_i \propto \Phi_{\text{task}}(s_i, g)$$

where g is the global goal and Φ maps task semantics into local influence signals.

These four rule families are not static: they can be rewritten or tuned dynamically.

3.3 Epistemic Blackboard as a Swarm-Rule Substrate

The Epistemic Blackboard stores:

- current swarm rule parameters $R(t)$,
- emergent signals (e.g., momentum flow, posture consensus),
- global goals and semantic intent,
- conflict resolution markers,
- hierarchical modulation factors.

Unlike traditional blackboard systems, the blackboard here functions as a *dynamic genome* of swarm behavior. All joint-agents read rules from the blackboard and update their policies accordingly.

3.4 Blockchain-Secured Swarm Rule Evolution

Because swarm rules influence the entire embodiment, any modification must be validated to prevent dangerous emergent behaviors. Following the blockchain-secured architecture described in the companion paper, we introduce:

- **Swarm-Rule Proposal**: generated by a task agent or learning module.
- **Validator Set**: composed of whole-body, limb-level, and safety-monitor agents.
- **Consensus Round**: validators check consistency with:
  - physical safety constraints,
  - kinematic feasibility,

- previously committed swarm rules,
  - agent authorization.
- **Commit**: rule becomes part of the robot's internal blockchain.

This ensures that emergent behaviors remain safe and cognitively interpretable.

3.5 Resulting Architecture Overview

The overall architecture is summarized as:

$$\text{Swarm Humanoid} = \big(A_1, \ldots, A_n; R(t); \text{Blackboard}; \text{Blockchain}\big)$$

where:

- A_i = joint-level swarm agents,
- $R(t)$ = live swarm rule set,
- **Blackboard** = shared epistemic and rule substrate,
- **Blockchain** = secure validation and evolution layer.

This architecture yields a humanoid control system capable of emergent, adaptive, and secure whole-body coordination.

4 Mathematical Formulation of Swarm Interaction Dynamics and Safety Constraints

This section formalizes the mathematical foundations of swarm-based humanoid control. Each joint is modeled as a swarm agent whose behavior emerges from local interaction rules (alignment, cohesion, avoidance, and task coupling) and global safety constraints. The resulting whole-body motion arises from decentralized computations rather than centralized optimization, extending prior formulations used in reflex-based humanoid control [2] and distributed locomotion [4].

4.1 State Representation of Joint Agents

Let the humanoid consist of n actuated joints. Each joint is represented as an agent:

$$A_i = \{s_i(t), u_i(t), \mathcal{N}_i(t)\}$$

where:

- $s_i(t)$ is the joint state (angle, velocity, torque),
- $u_i(t)$ is the control input (torque command),

- $\mathcal{N}_i(t)$ is the neighborhood set (kinematic chain neighbors or dynamically assigned neighbors).

The joint dynamics follow:

$$\dot{s}_i(t) = f(s_i(t), u_i(t))$$

where $f(\cdot)$ is the actuator-specific or learned dynamic model.

4.2 Swarm Interaction Model

The total swarm influence on a joint i is defined as:

$$U_i(t) = \alpha\, U_{\text{align}}(i) + \beta\, U_{\text{cohesion}}(i) + \gamma\, U_{\text{avoid}}(i) + \delta\, U_{\text{task}}(i)$$

where $\alpha, \beta, \gamma, \delta$ are adaptive influence weights stored on the Epistemic Blackboard.

4.2.1 1) Alignment Term

Following classical swarm flocking behavior:

$$U_{\text{align}}(i) = \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} (u_j - u_i)$$

This term encourages synchronization of local joint efforts, analogous to biological motor synergies.

4.2.2 2) Cohesion Term

The cohesion term drives joints toward a shared limb- or body-level reference:

$$U_{\text{cohesion}}(i) = (s_{\text{centroid}} - s_i)$$

where:

$$s_{\text{centroid}} = \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} s_j$$

This captures emergent stabilization similar to distributed posture control in humanoid systems [7].

4.2.3 3) Avoidance Term (Safety Potential)

We model safety avoidance as the gradient of a safety potential:

$$U_{\text{avoid}}(i) = -\nabla V_{\text{safety}}(s_i)$$

The potential V_{safety} penalizes:

- approaching joint limits,

- self-collision risks,
- excessive torque or velocity,
- instability indicators from whole-body balance models.

A common form is:

$$V_{\text{safety}}(s_i) = \sum_k \frac{1}{(d_{ik} - d_{\min})^2}$$

where d_{ik} is distance to constraint k. This formulation follows potential-field safety models in legged locomotion [8].

4.2.4 4) Task-Coupling Term

The task coupling term projects high-level goals into local influence fields:

$$U_{\text{task}}(i) = \Phi_{\text{task}}\big(s_i(t), g(t)\big)$$

where Φ_{task} may represent:

- whole-body momentum shaping,
- hand/foot target tracking,
- balance objective distributions,
- learned latent action fields [9].

4.3 Swarm-Controlled Joint Dynamics

The updated control input for joint i is:

$$u_i(t + \Delta t) = u_i(t) + \eta\, U_i(t)$$

where η is a swarm responsiveness factor.

This decentralized update rule replaces classical global optimization with emergent coordination, analogous to distributed control in biological motor systems.

4.4 Stability Through Distributed Consensus

Let:

$$x(t) = \begin{bmatrix} s_1(t) \\ \vdots \\ s_n(t) \end{bmatrix}$$

be the stacked state vector.

For stability, swarm interactions must satisfy:

$$\|x(t+1) - x^*\| \le \lambda \|x(t) - x^*\|$$

with $0 < \lambda < 1$, where x^* is the coordinated desired posture.

This corresponds to standard convergence in distributed consensus theory [4].

4.5 Blockchain-Secured Constraints

Each swarm rule update must satisfy:

$$\mathcal{V}(R_t) = \begin{cases} \text{valid}, & \text{if rule satisfies safety, stability, authorization constraints} \\ \text{invalid}, & \text{otherwise} \end{cases}$$

Validators check:

- collision avoidance guarantees,
- joint-limit preservation,
- Lyapunov-based stability bounds,
- agent identity and permissions [3].

Approved swarm-rule modifications are committed to the internal blockchain as immutable transitions, ensuring epistemic safety.

4.6 Safety Envelope and Hard Constraints

The final allowed control input for joint i is:

$$u_i^{\text{safe}} = \mathrm{Proj}_{\mathcal{C}}\big(u_i(t + \Delta t)\big)$$

where $\mathcal{C}$ is the feasible control set defined by:

- joint torque limits,
- velocity limits,
- balance and ZMP stability conditions,
- environmental contact constraints.

This projection step aligns with whole-body safety envelopes used in humanoid stability frameworks.

5 Integration With the Epistemic Blackboard and Blockchain: Dynamic Rule Evolution

In swarm-based humanoid control, the Epistemic Blackboard functions as the shared cognitive substrate that stores swarm coordination rules, rule parameters, emergent signals, and collective constraints. The internal blockchain ensures that modifications to these rules are cryptographically validated and remain safe throughout the robot's embodied operation. Together, these components enable real-time evolution of swarm behavior while guaranteeing epistemic integrity.

5.1 Blackboard as a Swarm-Rule Substrate

The Epistemic Blackboard stores the set of swarm rules $R(t)$, consisting of alignment, cohesion, avoidance, and task-coupling parameters:

$$R(t) = \{r_{\text{align}}, r_{\text{cohesion}}, r_{\text{avoid}}, r_{\text{task}}, \alpha(t), \beta(t), \gamma(t), \delta(t)\}.$$

Unlike classical blackboard architectures, which store symbolic knowledge or intermediate problem-solving results, the proposed architecture extends the blackboard into a real-time swarm-rule substrate with the following capabilities:

- **Rule Storage**: persistent representation of swarm rules.
- **Rule Propagation**: real-time broadcasting of updated rules to all joint-level agents.
- **Emergent-State Aggregation**: collection of distributed signals (e.g., posture consensus, torque gradients).
- **Constraint Enforcement**: embedding safety envelopes and feasibility constraints.

Because swarm coordination depends directly on the rules, ensuring correctness of $R(t)$ is critical for safe behavior.

5.2 Rule Update Cycle

At each control cycle, swarm rules may be:

1. updated,
2. replaced,
3. tuned,
4. or augmented

based on:

- task intent,
- whole-body performance,
- sensory feedback,
- learned policy adjustments,
- predictive models.

Let $R'(t)$ represent a proposed modification. The rule update cycle proceeds as follows:

1. **Proposal Generation**: $R'(t)$ is generated by a higher-level controller, learning module, or predictive swarm optimizer.

2. **Validation**: validators inspect the rule for safety, consistency, and feasibility.
3. **Consensus**: validators vote using a lightweight BFT-style consensus [3].
4. **Commit**: if accepted, $R'(t)$ is written to the internal blockchain.
5. **Propagation**: the blackboard updates to $R(t+1) = R'(t)$, and all joint agents incorporate the new rules.

This process supports dynamic adaptation while preserving safety-critical constraints.

5.3 Blockchain as the Epistemic Gatekeeper

Following the architecture introduced in the companion paper, each proposed rule update $R'(t)$ is encoded as a blockchain transaction:

$$B_t = \{\text{timestamp}, \text{rule\_update}, \text{agent\_id}, \text{signature}, \text{prev\_hash}\}.$$

Validators examine:

- **Authorization**: is the proposing agent allowed to modify this part of the rule-space?
- **Safety Compliance**: does the rule violate joint limits, collision envelopes, or stability criteria?
- **Consistency**: is the rule compatible with other committed rules?
- **Physical Feasibility**: can the swarm dynamics converge under the new parameters?
- **Semantic Validity**: if the rule propagates task intent, does it match the goal representation?

If validated, the rule becomes the new authoritative state for the robot.
This blockchain-mediated rule evolution ensures:

- no malformed rule can propagate,
- no malicious agent can alter global behavior,
- every rule has a traceable provenance,
- joint agents always operate with a consistent rule set.

5.4 Joint Agent Synchronization With Updated Rules

After a rule is committed, the blackboard triggers a synchronization event:

$$\text{Notify}(A_i) : R(t+1)$$

Each joint updates its local controller:

$$u_i(t) \leftarrow \pi_i\big(s_i(t), \mathcal{N}_i(t), R(t+1)\big)$$

This step ensures that:

- all swarm agents immediately adopt the new rules,
- alignment and cohesion terms reflect the updated parameters,
- avoidance and safety fields are recalculated consistently,
- task coupling reflects updated semantic intent.

5.5 Dynamic Adaptation and Evolutionary Learning

Beyond fixed rule sets, the architecture supports evolutionary rule adaptation inspired by swarm learning:

$$R'(t) = \Psi\big(R(t), E(t)\big)$$

where:

- Ψ is an evolutionary operator (mutation, crossover, gradient update),
- $E(t)$ is emergent swarm performance (e.g., stability residuals, energy use).

This aligns with contemporary work on adaptive multi-agent control and emergent RL [6, 9].

Because evolutionary operators modify core swarm behavior, blockchain validation guarantees that:

- unsafe mutations are rejected,
- destabilizing rule changes never reach the blackboard,
- only feasible emergent behaviors propagate through the swarm.

5.6 Summary of the Integration Architecture

The combined architecture can be expressed as:

$$R(t+1) = \text{Commit}\Big(\text{Validate}\big(R'(t)\big)\Big)$$

$$u_i(t+1) = \pi_i\big(s_i(t), \mathcal{N}_i(t), R(t+1)\big)$$

Together, the Epistemic Blackboard and the blockchain create a secure, adaptive substrate for swarm coordination, ensuring that emergent whole-body behaviors remain safe, stable, and semantically consistent.

6 Case Study: Emergent Behavior in Swarm-Based Humanoid Control

This section demonstrates how swarm-based humanoid control produces complex whole-body behaviors through local interactions among joint-level agents. Unlike hierarchical approaches [7], the behaviors emerge from distributed swarm rules dynamically stored in the Epistemic Blackboard and validated by the internal blockchain.

We present three scenarios:

1. emergent balance recovery under perturbation,
2. compliant whole-body reaching,
3. adaptive posture stabilization.

Each scenario illustrates how joint-level agents use swarm rules to coordinate motion without explicit global planning.

6.1 Scenario 1: Emergent Balance Recovery

Consider a humanoid robot subjected to a lateral perturbation at the torso. Traditionally, balance recovery requires whole-body optimization or centralized feedback control [4]. In a swarm-based architecture, each joint responds according to local rules and shared emergent signals.

Step 1: Local Disturbance Detection

Joint agents in the hip, ankle, and knee detect abnormal acceleration or torque deviation:

$$u_i^{\text{reflex}} \propto \nabla s_i(t)$$

This mirrors reflex-based control in prior humanoid studies [2].

Step 2: Swarm Rule Activation from Blackboard

The blackboard broadcasts a disturbance tag:

$$R(t) \ni \texttt{DisturbanceFlag}(D, M)$$

which triggers:

- increased alignment weight α,
- increased cohesion weight β,
- increased safety potential strength γ.

All joint agents update their local policies:

$$u_i(t+1) = \pi_i(s_i, \mathcal{N}_i, R(t))$$

Step 3: Emergent Whole-Body Coordination

Joints collectively shift in opposite direction to the perturbation due to alignment and cohesion terms:

$$U_{\text{align}}(i),\ U_{\text{cohesion}}(i)\ \Rightarrow\ \text{group momentum compensation}$$

The emergent behavior resembles:

- ankle-hip synergy,
- coordinated arm counter-rotation,
- global center-of-mass redirection.

Step 4: Stability Restoration

Once balance is recovered, the blackboard clears the disturbance flag and resets swarm parameters.

This behavior is not pre-scripted; it results from swarm dynamics under safety potentials and emergent consensus.

6.2 Scenario 2: Emergent Compliant Whole-Body Reaching

In traditional humanoid control, whole-body reaching requires solving complex nonlinear optimization problems. In the swarm architecture, compliance and coordination arise from swarm interactions.

Step 1: Task-Coupling Rule Insertion

A task-level module writes a rule to the blackboard:

$$R(t) \ni r_{\text{task}} = \Phi_{\text{task}}(g(t))$$

representing reaching intent.

Step 2: Blockchain Validation

Validators check:

- self-collision safety,
- torque-limit feasibility,
- stability envelope (ZMP or momentum-based),
- consistency with previously committed rules.

This prevents unsafe emergent behaviors.

Step 3: Swarm Execution

Joint agents receive local task influence:

$$U_{\text{task}}(i) = \Phi_{\text{task}}(s_i, g)$$

Combined with cohesion and alignment, the swarm forms an emergent whole-body reaching motion:

- torso leans,
- hips rotate,
- knees adjust balance,
- free arm counter-balances.

6.3 Scenario 3: Adaptive Posture Stabilization

Consider slow drift in posture due to sensor noise or external micro-disturbances.

Step 1: Emergent Drift Detection

Joint agents detect small deviations:

$$\epsilon_i = s_i - s_i^{\text{desired}}$$

Step 2: Swarm Synchronization

The blackboard triggers a global drift compensation rule.
Swarm agents apply:

$$U_{\text{align}}(i) \quad \text{and} \quad U_{\text{cohesion}}(i)$$

causing:

- joint micro-adjustments,
- distributed drift correction,
- whole-body stiffening in a compliant way.

Step 3: Stability Through Consensus

The robot converges to a stable posture via swarm consensus:

$$x(t+1) = Wx(t)$$

where W is a doubly-stochastic swarm interaction matrix.

6.4 Discussion of Emergent Behaviors

Across these scenarios, we observe that:

- **coordination emerges** from swarm rules rather than explicit trajectories,
- **safety is enforced** by avoidance potentials and blockchain validation,
- **adaptation arises dynamically** via rule modulation,
- **robustness is inherent** because failure of a single agent does not break global behavior,
- **behaviors are flexible** and generalize across tasks.

These examples show the potential of swarm-based control to replace or augment classical optimization frameworks in humanoid robotics.

7 Discussion

The proposed swarm-based humanoid control architecture represents a shift from traditional centralized or hierarchical control approaches toward a distributed, emergent, and evolutionarily adaptable framework. This section analyzes the theoretical implications, benefits, limitations, and connections to both biological motor systems and recent developments in machine learning.

7.1 Comparison With Classical Humanoid Control

Traditional humanoid control frameworks rely heavily on:

- centralized optimization (e.g., whole-body QP solvers),
- operational space control [7],
- hierarchical task structures,
- model-based planning and control pipelines.

These approaches are effective in structured settings but can become brittle under:

- model inaccuracies,
- external disturbances,
- complex contact interactions,
- sensor noise,
- unstructured environments.

By contrast, swarm-based control distributes computation among joint-level agents, enabling:

- greater robustness,

- natural redundancy,
- local adaptation,
- emergent resilience,
- real-time reorganization.

Rather than solving one global optimization, coordination emerges from local rules and dynamic interactions [4].

7.2 The Role of Emergence in Humanoid Control

Emergence is central to swarm coordination. In the proposed framework:

$$\text{Whole-body behavior} = \text{Emergent}(A_1, \ldots, A_n, R(t))$$

This aligns with:

- biological motor synergies,
- distributed reflex arcs [2],
- consensus mechanisms in multi-agent robotics,
- evolutionary learning of motor behaviors [9].

Emergent control offers:

- flexible behavior adaptation,
- smooth interpolation between tasks,
- fault-tolerant motion,
- scalable coordination across many DoF.

Rather than optimizing trajectories explicitly, behaviors self-organize under local rules.

7.3 Benefits of the Swarm-Based Architecture

The architecture provides several benefits:

1. Adaptability

Swarm agents can rapidly adjust to new tasks, disturbances, or constraints by updating rule parameters without requiring global replanning.

2. Fault Tolerance

Failure of one or several joint agents does not cause systemic collapse. Redundancy emerges naturally through swarm interactions.

3. Real-Time Flexibility

Agents perform local computations at high frequency, enabling reflex-like responses comparable to biological motor control systems.

4. Scalability With Robotic Morphology

As robots evolve with more degrees of freedom, swarm interactions scale linearly, whereas classical whole-body optimization scales poorly.

5. Blockchain-Secured Rule Evolution

The use of blockchain ensures that emergent behaviors remain safe and cognitively interpretable:

- no unsafe mutation of swarm rules,
- cryptographically validated rule transitions,
- complete traceability of internal decision-making,
- protection against compromised modules.

This provides unprecedented epistemic safety for emergent robotic intelligence.

7.4 Limitations

Despite its promise, the architecture has several limitations:

1. Parameter Sensitivity

Swarm rules require careful selection of:

- alignment strength (α),
- cohesion strength (β),
- avoidance potential gains (γ),
- task coupling parameters (δ).

Improper values may cause oscillations or poor convergence.

2. Emergence Unpredictability

While emergence is powerful, it can make:

- verification harder,
- behavior less predictable,
- debugging more complex.

3. Computational Requirements

Swarm control is decentralized, but:

- maintaining a blackboard,
- running blockchain validation,
- computing potential fields,

introduce computational overhead.

4. Need for Safety Guarantees

Although safety is enforced via blockchain and potentials, formal proofs and robust bounds are needed for deployment in real-world humanoids.

7.5 Relation to Biological Motor Control

The architecture aligns with findings in neuroscience:

- motor synergies arise from collective activation of distributed groups,
- spinal cord reflexes operate as distributed, semi-autonomous circuits,
- the central nervous system modulates local rules rather than computing full-body trajectories.

Thus, the swarm-based approach may provide a more biologically plausible model than centralized optimization.

7.6 Relation to Learning-Based Control

Swarm-based humanoid control complements modern learning-based methods:

- hierarchical RL [6],
- whole-body skill learning [9],
- multi-agent RL [8].

Swarm rules can be:

- learned,
- optimized,
- evolved,
- or discovered through exploration.

Meanwhile, blockchain validation ensures that only safe updates propagate to the swarm.

7.7 Implications for Future Humanoid Design

Swarm-based architectures may lead to:

- highly modular humanoids with joint-level autonomy,
- morphologies optimized for distributed control,
- robots capable of emergent adaptation,
- shared-rule ecosystems enabling multi-humanoid cooperation.

These implications suggest a new paradigm for designing future embodied AI systems.

8 Conclusion and Future Work

This paper introduced a novel swarm-based humanoid control architecture in which each joint is modeled as an autonomous agent participating in emergent whole-body coordination through locally defined swarm rules. These rules—alignment, cohesion, avoidance, and task coupling—are dynamically stored in the Epistemic Blackboard and are cryptographically validated through an internal blockchain layer to ensure epistemic safety, physical feasibility, and rule integrity.

The proposed architecture represents a substantial shift from traditional humanoid robotics, which relies on centralized optimization, rigid hierarchical control structures, or fixed reflex loops [7]. By distributing control among joint-level agents, the humanoid gains:

- robustness against disturbances and local failures,
- real-time adaptability to environmental changes,
- fluid and compliant whole-body behaviors,
- scalable coordination across many degrees of freedom,
- traceable and safe rule evolution via blockchain validation.

Through case studies on balance recovery, compliant whole-body reaching, and adaptive posture stabilization, we demonstrated how global behaviors can emerge from simple local interactions, extending earlier concepts in distributed humanoid control [4] and reflex-based motor behavior [2]. This emergent approach aligns with biological insights into motor synergies and distributed neural control and complements modern learning-based methods such as hierarchical reinforcement learning and whole-body skill acquisition [6, 9].

8.1 Future Work

Several directions for future research are envisioned:

1. Formal Stability Analysis

Although preliminary consensus-based stability guarantees were introduced, a comprehensive Lyapunov or passivity-based analysis for emergent swarm dynamics is needed to ensure safe deployment on physical humanoid systems.

2. Learning Swarm Rules

Future work may incorporate:

- reinforcement learning for discovering optimal swarm parameters,
- evolutionary algorithms for rule adaptation,
- meta-learning for task-driven swarm behavior shaping.

3. Multi-Modal Sensory Integration

Integrating vision, tactile sensing, and inertial feedback into swarm rule dynamics may enable richer emergent behaviors and situational awareness.

4. Hardware Implementation

Testing on real humanoid platforms will reveal:

- latency tradeoffs,
- blackboard bandwidth constraints,
- blockchain consensus scalability,
- real-world safety limits.

5. Multi-Humanoid Swarm Coordination

An exciting extension involves multiple humanoid robots sharing:

- a distributed blackboard,
- shared swarm rules,
- cooperative emergent behaviors.

This could enable coordinated tasks, such as collaborative manipulation or group locomotion, based on shared swarm intelligence.

8.2 Final Remarks

This work contributes a new conceptual foundation for humanoid control—one where intelligence is not imposed from the top down, but emerges from the bottom up through distributed interactions among local agents. By unifying swarm dynamics, epistemic blackboard architectures, and blockchain-secured rule evolution, this framework opens the door to a new class of adaptive, resilient, and semantically grounded humanoid systems.

The swarm-based approach invites a paradigm shift: humanoid robots may no longer require monolithic controllers, but instead can rely on the emergent coordination of many simple components, much like biological organisms.

Acknowledgments

The author acknowledges the contributions of all researchers at the Almudaya Research Institute and Universitas Muhammadiyah Ponorogo.

References

[1] Igor Afanasyev, Roman Kolotov, and Dzmitry Tsetserukou. Blockchain solutions for multi-agent robotic systems: Related work and open questions. *arXiv preprint arXiv:1903.11041*, 2019.

[2] Mohammad Ajallooeian et al. A reflex-based approach to humanoid locomotion control. In *IEEE Conference Proceedings*, 2013. IEEE Xplore page; access may require subscription.

[3] Ali Dorri, Feng Luo, Salil S. Kanhere, and Raja Jurdak. Blockchain in multi-agent systems: Applications, challenges, and open issues. *arXiv preprint arXiv:1908.10761*, 2019.

[4] Andrei Herdt, Holger Diedam, Pierre-Brice Wieber, Dimitar Dimitrov, Katja Mombaur, and Moritz Diehl. Online walking motion generation with automatic footstep placement. *Advanced Robotics*, 24(5–6):719–737, 2010.

[5] Twan Koolen et al. Distributed real-time whole-body control for humanoid robots. In *IEEE Conference Proceedings*, 2016. IEEE Xplore page; access may require subscription.

[6] Josh Merel, Saran Tunyasuvunakool, Jonas Rothfuss, and Nicolas Heess. Hierarchical control of complex tasks with reinforcement learning. *arXiv preprint arXiv:1811.09656*, 2019.

[7] Luis Sentis and Oussama Khatib. Synthesis of whole-body behaviors through hierarchical control of behavioral primitives. Technical report, Stanford Robotics Lab, 2005.

[8] Z. Xiong et al. Multi-agent reinforcement learning for locomotion control of humanoid robots. In *IEEE Conference Proceedings*, 2023. IEEE Xplore page; access may require subscription.

[9] Zhenjia Zhang, Yuan Ye, and Sehoon Ha. Learning whole-body motor skills for humanoid robots with reinforcement learning. *arXiv preprint arXiv:2303.03363*, 2023.