BIS INFORMATION TECHNOLOGY AND COMPUTER SCIENCE

Volume 3 2026 | DOI: 10.31603/bistycs.477
7th Borobudur International Symposium on Science and Technology (BIS-STE) 2025

UNIMMA PRESS
UNIVERSITAS MUHAMMADIYAH MAGELANG

Swarm-based coordination architecture for humanoid robots: A distributed multi-agent framework with secure rule evolution

Aslan Alwi[1*] and Munirah[1]

[1] Muhammadiyah University of Ponorogo, Ponorogo, Indonesia
[*]Corresponding author email: aslan.alwi@umpo.ac.id

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 distributed 2D swarm simulations 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. The implementation code is publicly available to ensure experimental reproducibility.

Published:
May 04, 2026

Selection and Peer-review under the responsibility of the 7th BIS-STE 2025 Committee

Keywords

Swarm intelligence, Humanoid control, Distributed coordination, Multi-agent systems, Secure consensus.

Introduction

Humanoid robots represent one of the most complex embodiments in robotics due to their high degrees of freedom, nonlinear joint coupling, and dynamic balance requirements. Traditional wholebody control approaches largely rely on centralized architectures, where a global controller computes torque or motion commands based on aggregated system states. While effective in structured environments, centralized schemes suffer from scalability limitations, single-point-of-failure risks, and reduced adaptability under distributed perturbations [9], [11]. As humanoid systems become increasingly modular and adaptive, there is a growing need for distributed coordination paradigms that enable local autonomy while preserving global stability.

Recent advances in distributed robotics and multi-agent systems (MAS) suggest that decentralized coordination can enhance robustness and scalability [12], [15]. In parallel, swarm intelligence has demonstrated how simple local interaction rules among agents can generate emergent global behaviors without centralized supervision [2], [3]. Swarm-based approaches have been successfully applied in multi-robot systems, formation control, and collective exploration tasks [7]. Distributed control approaches have been increasingly adopted in modern humanoid robotics. Classical whole-body frameworks and operational space control [14] 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. [8], demonstrate that locomotion can be achieved through multiple semi-independent controllers operating in coordination. Romualdi et al. [13] demonstrated that humanoid locomotion can be realized through coordinated multi-layer control architectures integrating whole-body position, velocity, and torque control.

More recently, multi-agent reinforcement learning has been explored for legged locomotion [10], 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.

However, their application has predominantly focused on inter-robot coordination rather than intra-humanoid structural coordination. The notion of modeling each joint or limb as an autonomous yet cooperative agent remains underexplored.

Another emerging dimension in distributed systems concerns governance and rule evolution. In adaptive robotic systems, control rules may require dynamic updates in response to environmental changes or learning outcomes. Yet, secure and verifiable rule modification mechanisms are rarely addressed in robotic control architectures. Recent work on blockchain-enabled robotics and secure MAS infrastructures highlights the potential of cryptographic verification and distributed consensus to ensure integrity and trustworthiness in rule propagation [4], [6]. 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. [5] describe blockchain mechanisms that enforce honesty and prevent malicious agents from manipulating shared information. Nevertheless, these approaches are typically designed for networked robotic fleets rather than embedded coordination within a single humanoid morphology.

Despite progress in distributed control, swarm intelligence, and secure multi-agent infrastructures, no existing framework integrates: (1) swarm-based intra-humanoid coordination, (2) distributed joint-level autonomy, and (3) cryptographically verifiable rule evolution within a unified architecture. This gap limits the development of humanoid systems capable of adaptive, scalable, and tamper-resistant coordination.

To address this limitation, this paper proposes a swarm-based coordination architecture for humanoid robots, formulated as a distributed multi-agent framework in which each joint is modeled as an autonomous agent governed by local swarm interaction rules. The architecture introduces a secure rule evolution layer that enables dynamic modification of coordination rules through cryptographic validation and distributed consensus mechanisms. Unlike conventional centralized control systems, the proposed framework distributes computation and decision-making across joint agents, enabling emergent stability and robustness through local interactions. The main contributions of this work are threefold:

1. A formal system model that represents humanoid joints as cooperative agents operating under swarm-based coordination principles.
2. A distributed coordination mechanism that enables emergent stabilization and adaptive behavior without reliance on centralized control.
3. A secure rule evolution protocol that ensures integrity and verifiability of dynamic control updates.

The remainder of this paper is organized as follows. Section II presents the proposed methods, including the system model, swarm coordination mechanism, and secure rule evolution protocol. Section III reports experimental results and discusses stability, scalability, and robustness properties. Finally, Section IV concludes the paper and outlines future research directions.

Methods

System representation

The humanoid robot is modeled as a distributed collection of N joint-agents. Each joint-agent is represented by a 2-dimensional state vector:

$$x_i(t) \in \mathrm{R}^2$$

No rigid-body dynamics or physical torque equations are used. The state vector represents an abstract joint configuration embedding used to evaluate decentralized coordination behavior.

Swarm-based local update rule

At each discrete timestep, every joint-agent updates its state using three local interaction components: (1) Alignment term; (2) Cohesion term; and (3) Stability term. The update rule is:

$$x_i(t+1) = x_i(t) + \alpha A_i(t) + \beta C_i(t) + \gamma S_i(t)$$

where:

Alignment:

$$A_i(t) = \frac{1}{N}\sum_{j=1}^{N}(x_j(t) - x_i(t))$$

Cohesion:

$$C_i(t) = \bar{x}(t) - x_i(t)$$

with

$$\bar{x}(t) = \frac{1}{N}\sum_{j=1}^{N} x_j(t)$$

Stability:

$$S_i(t) = -x_i(t)$$

The coordination parameters used in all experiments are:

$$\alpha = 0.05, \qquad \beta = 0.02, \qquad \gamma = 0.01$$

All joint-agents update synchronously.

Centralized baseline controller

For comparison, a centralized averaging controller is implemented:

$$x_i(t+1) = x_i(t) + 0.1(\bar{x}(t) - x_i(t))$$

This controller represents a global coordination scheme without decentralized swarm interaction.

Disturbance protocol

To evaluate robustness, a disturbance is injected at timestep $t = 50$ by perturbing a single joint agent:

$$x_1(t) \leftarrow x_1(t) + [5,-5]$$

The simulation runs for $T = 200$ timesteps.

Evaluation metrics

1. Stability Index. System stability is quantified using the Stability Index (SI):

$$SI = \frac{1}{T}\sum_{t=1}^{T}\left(\frac{1}{N}\sum_{i=1}^{N}\|x_i(t)\|\right.!$$

Lower SI indicates improved global stabilization.

2. Recovery Time. Recovery time is defined as the number of timesteps required after disturbance for the average norm:

$$\frac{1}{N}\sum_{i=1}^{N}\|x_i(t)\|$$

to fall below 20% of its initial value.

3. Scalability Evaluation. The experiment is repeated for:

$$N \in \{10, 20, 50, 100\}$$

to evaluate scalability of decentralized coordination.

Secure rule evolution mechanism

A lightweight rule validation protocol is implemented to govern parameter updates. A candidate parameter set:

$$(\alpha', \beta', \gamma')$$

is encoded and hashed using SHA-256:

$$h = \mathrm{SHA256}(\alpha', \beta', \gamma')$$

Each joint-agent participates in a distributed approval vote. A rule update is accepted if the approval ratio exceeds a 70% consensus threshold:

$$\text{Approval Ratio} \geq 0.7$$

Malicious parameter proposals (e.g., $\alpha = 1, \beta = 1, \gamma = 1$) are tested to evaluate rejection behavior.

Results and Discussion

The proposed swarm-based humanoid coordination architecture was validated through a 2D distributed simulation implemented in Python (Google Colab). Each joint was modeled as a 2D state vector updated via decentralized swarm interaction rules without rigid-body dynamics or centralized trajectory optimization.

Experimental setup

The simulation was conducted with varying numbers of joint-agents $N \in \{10, 20, 50, 100\}$. Each joint updated its state according to local alignment, cohesion, and stability rules:

$$x_i(t+1) = x_i(t) + \alpha A_i + \beta C_i + \gamma S_i$$

where $\alpha = 0.05$, $\beta = 0.02$, and $\gamma = 0.01$. At timestep $t = 50$, a disturbance was injected into a single joint to evaluate recovery behavior.

Stability and recovery performance

Stability was measured using the Stability Index (SI):

$$SI = \frac{1}{T}\sum_{t=1}^{T}\frac{1}{N}\sum_{i=1}^{N}\|x_i(t)\|$$

Recovery time was defined as the number of steps required for the average norm to return below 20% of its initial value after disturbance. Scalabillity and stability comparison presented in Table 1.

Table 1. Scalability and stability comparison

N	SI (Swarm)	SI (Centralized)	Recovery (Swarm)	Recovery (Centralized)
10	0.3698	0.6243	87	150
20	0.2502	0.4188	36	150
50	0.1969	0.3664	15	150
100	0.1297	0.2194	0	6

The results show that the swarm-based architecture consistently achieves lower Stability Index values compared to centralized control. Notably, recovery time decreases as the number of joint agents increases, indicating natural scalability of decentralized coordination. In contrast, the centralized controller exhibits persistent recovery plateaus, particularly for small-to-medium system sizes.

Disturbance recovery behavior

Figure 1 illustrates convergence of the swarm-based architecture before and after disturbance injection. The system demonstrates rapid redistribution of local state discrepancies through neighborhood interactions. Figure 2 shows centralized control behavior, where global averaging produces slower convergence and residual error.

Secure rule evolution validation

To validate the secure rule evolution mechanism, an extreme parameter update proposal ($\alpha = 1, \beta = 1, \gamma = 1$) was introduced. A distributed voting process with a 70% consensus threshold was applied. Secure rule evolution test presented in Table 2.

Table 2. Secure rule evolution test

Proposal	Approval Ratio	Accepted
Extreme Update	0.20	No

The malicious rule update was rejected due to insufficient consensus approval. This demonstrates that the lightweight cryptographic validation layer prevents unsafe modifications while preserving decentralized governance.

Discussion

The experimental results confirm that stable global behavior can emerge purely from local swarm rules without centralized optimization. As the number of joints increases, emergent stabilization improves rather than degrades, highlighting the scalability of the architecture. Furthermore, the secure rule evolution mechanism enables adaptive parameter modification while ensuring system integrity. Together, these findings support the feasibility of hierarchical swarm-based humanoid control under secure decentralized coordination.

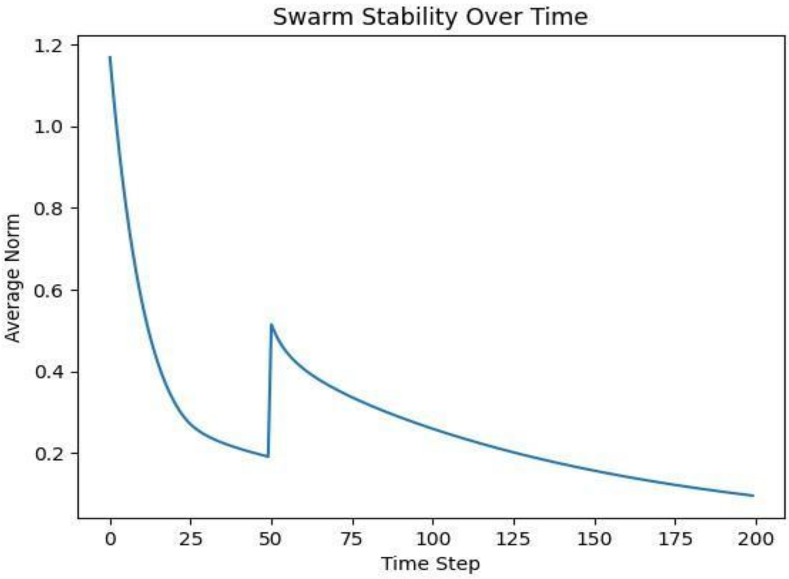

Figure 1. Swarm stability convergence over time. Disturbance injected at $t = 50$.

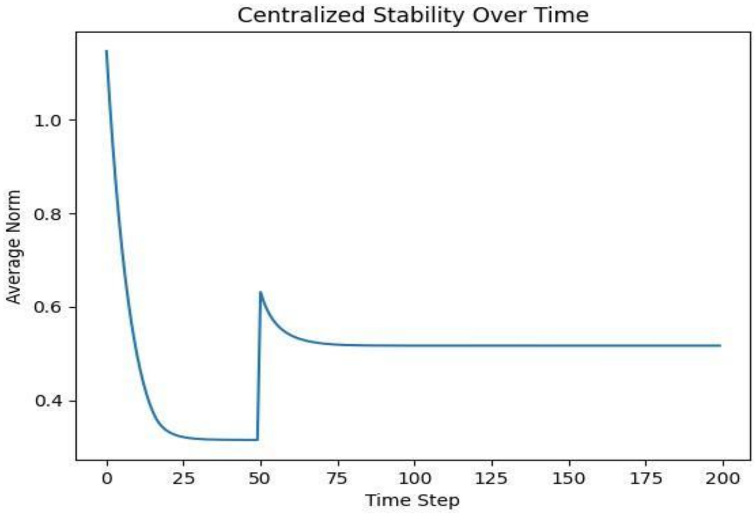

Figure 2. Centralized stability convergence over time.

Conclusion

This paper introduced a swarm-based coordination architecture for humanoid robots in which each joint operates as an autonomous agent governed by decentralized alignment, cohesion, and stability rules. Unlike conventional centralized control schemes, the proposed framework does not rely on trajectory optimization or rigid-body dynamic modeling, but instead demonstrates emergent stabilization through local interaction rules.

Experimental validation was conducted using a distributed 2D simulation with varying numbers of joint-agents ($N = 10,20,50,100$). Across all scales, the swarm-based architecture achieved consistently lower Stability Index (SI) values compared to centralized control. For example, at $N = 20$, the SI was reduced from 0.4188 (centralized) to 0.2502 (swarm), and at $N = 100$, from 0.2194 to 0.1297.

Recovery time following disturbance injection also improved significantly under decentralized coordination. At $N = 50$, recovery time decreased from 150 steps (centralized plateau) to 15 steps under swarm control. As the number of agents increased, recovery time further decreased, reaching immediate stabilization behavior at $N = 100$.

In addition, the secure rule evolution mechanism successfully rejected malicious parameter updates that failed to reach the 70% consensus threshold, demonstrating tamper-resistant governance of coordination parameters without introducing centralized authority.

These results confirm that stable global behavior can emerge from purely local swarm interactions, and that system robustness improves with scale rather than degrading. The proposed hierarchical separation between rule execution and secure rule evolution establishes a foundation for scalable, fault-tolerant humanoid coordination architectures grounded in decentralized swarm intelligence.

Acknowledgments

The author expresses sincere gratitude to Universitas Muhammadiyah Ponorogo for its financial support for conference participation and for the valuable contributions of its researchers to this work. The author also acknowledges the constructive insights provided by the Almudaya Research Institute.

A reproducibility and code availability

The distributed swarm-based coordination experiments presented in this paper were implemented in Python and executed in a Google Colab environment. The implementation includes:

1. Distributed 2D joint-agent simulation
2. Swarm-based local update rules (alignment, cohesion, stability)
3. Centralized baseline controller
4. Disturbance injection protocol
5. Stability Index and recovery time computation
6. Secure rule evolution mechanism with SHA-256 validation and consensus voting

The full source code used to generate all experimental results and tables in this paper is publicly available at: https://github.com/elangbijak4/Swarm-Humanoid-Control-PoC. All numerical results reported in Tables 1 and 2 can be reproduced by executing the provided notebook with the specified random seed configuration.

References

1. Igor Afanasyev, Roman Kolotov, and Dzmitry Tsetserukou. Blockchain solutions for multiagent robotic systems: Related work and open questions. *arXiv preprint arXiv:1903.11041*, 2019.
2. E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems*.

Oxford University Press, 1999.
3. M. Dorigo and E. S¸ahin. Swarm robotics special issue editorial. *Autonomous Robots*, 17(23):111–113, 2004.
4. Dorri, S. S. Kanhere, and R. Jurdak. Blockchain in internet of things: Challenges and solutions. *arXiv preprint arXiv:1608.05187*, 2016.
5. Ali Dorri, Feng Luo, Salil S. Kanhere, and Raja Jurdak. Blockchain in multi-agent systems: Applications, challenges, and open issues. *arXiv preprint arXiv:1912.01606*, 2019.
6. M. A. Ferrag and L. Shu. The performance evaluation of blockchain-based security and privacy systems for the internet of things: A tutorial. *IEEE Internet of Things Journal*, 8(24), 2021. Tutorial on performance evaluation of blockchain for IoT security and privacy.
7. H. Hamann. Swarm robotics: A formal approach. *Springer*, 2018.
8. 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.
9. S. Kajita, K. Kaneko, and F. Kanehiro. Biped walking pattern generation by using preview control of zero-moment point. *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1620–1626, 2003.
10. Rezwan Al Islam Khan, Chenyun Zhang, Zhongxiao Deng, Anzheng Zhang, Yuzhen Pan, Xuan Zhao, Huiliang Shang, and Ruijiao Li. Multi-agent reinforcement learning tracking control of a bionic wheel-legged quadruped. *Machines*, 12(12):902, 2024.
11. D. E. Orin and A. Goswami. Centroidal momentum matrix of a humanoid robot: Structure and properties. *IEEE/ASME Transactions on Mechatronics*, 13(6):767–775, 2008.
12. L. E. Parker. Distributed intelligence: Overview of the field and its application in multi-robot systems. *Journal of Physical Agents*, 2(1):5–14, 2008.
13. Giulio Romualdi, Stefano Dafarra, Yue Hu, Prashanth Ramadoss, Francisco Javier Andrade Chavez, Silvio Traversaro, and Daniele Pucci. A benchmarking of dcm-based architectures for position, velocity and torque-controlled humanoid robots. *International Journal of Humanoid Robotics*, 17(1), 2020. Discusses real-time whole-body control architectures for humanoid robots using Divergent Component of Motion (DCM) frameworks.
14. Luis Sentis and Oussama Khatib. Synthesis of whole-body behaviors through hierarchical control of behavioral primitives. Technical report, Stanford Robotics Lab, 2005.
15. M. Wooldridge. An introduction to multiagent systems. *John Wiley & Sons*, 2009.