Scidata Consortium
An international open-access platform

Research Article

OPEN ACCESS

Consensus-Driven Particle Swarm Optimization for Cooperative Resource Allocation in Heterogeneous Distributed Sensor and Actuator Networks

Laith Haddad,[1] Rami Khatib,[2] Tareq Suleiman,[3] Ariful Islam[4]

[1] Department of Computer Science, Al-Hussein Bin Talal University, Desert Highway, Ma'an, Jordan;
[2] Department of Computer Engineering, Philadelphia University, Jarash Road, Amman, Jordan;
[3] Department of Information Technology, Tafila Technical University, Al-Eis Street, Tafila, Jordan;
[4] Dhaka, Bangladesh;

Abstract

Distributed sensor and actuator networks operate under constraints of energy, bandwidth, latency, and partial observability, while often facing heterogeneous node capabilities and time-varying connectivity. Cooperative resource allocation in such networks requires balancing local operational goals with system-level coordination without imposing heavy centralized infrastructure. Particle Swarm Optimization has been applied to nonconvex resource assignment because it is derivative-free and parallelizable, yet classical forms are communication-blind and sensitive to network fragmentation. This paper examines a consensus-driven variant of Particle Swarm Optimization designed for cooperative allocation across heterogeneous nodes that exchange low-dimensional states with neighbors. The method integrates a mixing process that steers particles toward neighborhood barycenters, while preserving exploratory diversity necessary to escape poor local allocations. The study formulates a network-regularized objective that captures coupling through shared actuators and congested links, and examines stability under general conditions on mixing weights, inertia, and noise. A Lyapunov-style argument outlines how consensus error and swarm potential are jointly contracted in expectation under mild spectral-gap assumptions on the communication graph. The computational and communication footprints are characterized, with attention to asynchronous message arrivals, quantized signaling, and privacy of local cost information. Numerical protocols are described to probe sensitivity to network sparsity, heterogeneity, and measurement noise. The discussion offers guidance on parameterization for typical embedded deployments and interprets empirical behavior in terms of dynamic regret and allocation smoothness. The presentation aims to be technically complete while remaining conservative in claims about performance under unmodeled disturbances.

1. Introduction

Cooperative resource allocation in distributed sensor and actuator networks represents a fundamental challenge and opportunity in the design of modern cyber-physical systems [1]. These networks consist of spatially dispersed nodes that collectively perform sensing, computation, communication, and control tasks. Each node contributes to the overall system performance, yet must operate under its own local limitations in energy, bandwidth, and actuation range. The goal of cooperative resource allocation is to ensure that all nodes in the network coordinate effectively to achieve a global objective—such as maintaining environmental stability, optimizing process efficiency, or responding adaptively to dynamic conditions—without centralized oversight. This paradigm emerges in a wide variety of real-world contexts, from environmental monitoring systems that measure and regulate physical variables like temperature and humidity, to industrial process networks managing distributed actuators such as valves and pumps, to building-scale automation systems balancing thermal comfort, lighting efficiency, and air quality. In each of these domains, the cooperation among nodes allows for scalable, robust, and energy-efficient operation, but it also introduces layers of complexity stemming from heterogeneity, communication constraints, and the interplay of local and global objectives. [2]

Heterogeneity is a defining characteristic of distributed sensor and actuator networks. No two nodes in such systems are perfectly identical; they may differ in hardware capabilities, sensing modalities, actuation authority, communication interfaces, or power sources. For example, in a smart agricultural deployment, one node may have a high-resolution soil moisture sensor powered by solar energy, while another node may rely on a low-power temperature sensor operating intermittently on a battery. Similarly, in industrial process control, certain actuators may have fast response times and wide operating ranges, while others are slower or limited by mechanical constraints. These differences mean that not all nodes can contribute equally to every task, necessitating intelligent strategies for dividing work and distributing control authority [3]. Moreover, nodes may experience time-varying conditions such as battery depletion, network congestion, or environmental interference, further complicating allocation decisions. A robust cooperative allocation scheme must therefore be adaptive, dynamically redistributing responsibilities as local conditions evolve while still maintaining overall system objectives.

The resource allocation problem is also deeply shaped by the constraints that define feasible and safe operation. Each actuator typically operates within finite physical limits—such as maximum torque, flow rate, or voltage—and violating these limits can cause hardware damage or unsafe system behavior. Similarly, sensors have operating envelopes defined by sampling rates, resolution, and accuracy bounds. At the network level, communication bandwidth is finite, and the shared use of wireless channels introduces coupling between nodes: when one node transmits, others must wait or risk packet collisions [4]. Power constraints also introduce coupling, particularly in battery-operated systems where energy must be conserved to ensure long-term operation. Additionally, some systems impose aggregate limits on actuation, such as total power consumption in a microgrid or total airflow in a building HVAC system. These constraints create a multi-dimensional optimization space where individual node decisions influence—and are influenced by—the decisions of others. Designing allocation algorithms that respect these interdependencies while minimizing overall cost or maximizing performance is a central challenge in distributed network control.

In traditional settings, resource allocation problems of this kind might be approached through centralized optimization, where a single control entity gathers data from all nodes, computes an optimal configuration, and dispatches commands back to each actuator [5]. While conceptually straightforward, this approach quickly becomes impractical in large or geographically distributed networks. Centralized computation introduces latency due to data aggregation and command dissemination, and it creates a single point of failure that can jeopardize the entire system if the central controller goes offline or is compromised. Moreover, centralized control raises privacy and security concerns, as local nodes may be reluctant or unable to share all their data with a central authority. In the context of environmental monitoring, for example, individual stations might belong to different organizations or jurisdictions, limiting the feasibility of centralized data sharing. In industrial or building applications, the need for fast response times—on the order of milliseconds or seconds—makes the delay introduced by centralized computation unacceptable. For these reasons, researchers and practitioners have increasingly turned to distributed and decentralized control paradigms. [6]

Distributed cooperative control strategies allow each node to make local decisions based on its own data and limited information exchanged with neighboring nodes. Rather than relying on a central controller, the network as a whole converges toward a consensus or equilibrium through iterative, communications-aware algorithms. This decentralized structure greatly enhances scalability, since each node only needs to process local information and communicate with a small subset of others. It also improves resilience: if one or more nodes fail, the rest of the system can often continue operating, adjusting their behavior to compensate for the missing participants. Distributed algorithms are particularly valuable in mobile or ad-hoc networks, where topology changes dynamically as nodes move, join, or leave the system [7]. Such flexibility makes them well suited for applications like autonomous vehicle coordination, adaptive energy grids, or disaster response systems that must function reliably in uncertain and evolving conditions.

One of the core elements in distributed resource allocation is the design of consensus and optimization protocols that operate efficiently under limited communication. Nodes must share information about local measurements, constraints, and objectives, but excessive communication consumes energy and bandwidth, which are often scarce resources. Therefore, communications-aware search strategies are designed to minimize the number and size of transmitted messages while still enabling the network to approximate globally optimal behavior. Tech-

Node Type	Energy Source	Bandwidth (kbps)	Range (m)	Task Priority
Temperature Sensor	Battery	250	30	Medium
Humidity Sensor	Solar	500	25	High
Actuator Valve	Wired	1000	10	Critical
Controller Node	AC Power	2000	50	High
Gateway Node	AC Power	5000	100	Critical

Table 1: Representative node characteristics in a distributed sensor-actuator network.

Constraint Type	Limit Value	Unit	Affected Nodes	Safety Margin (%)
Power Consumption	5.0	W	All	15
Communication Delay	100	ms	Wireless	10
Flow Rate	2.5	L/s	Actuators	5
Temperature Drift	0.8	°C	Sensors	20
Total Bandwidth	10	Mbps	Network	25

Table 2: Key operational constraints influencing cooperative allocation.

Algorithm	Communication Overhead	Convergence Speed	Energy Efficiency
Centralized PSO	High	Fast	Low
Distributed PSO	Medium	Moderate	High
Consensus-Driven PSO	Low	Fast	High
Gossip-Based PSO	Very Low	Moderate	Very High
Hybrid PSO-MPC	Medium	Very Fast	Medium

Table 3: Comparison of swarm-based cooperative allocation algorithms.

niques such as gossip algorithms, distributed gradient descent, and consensus-based optimization have been widely studied for this purpose. In gossip protocols, nodes randomly exchange information with neighbors and iteratively update their local estimates based on these interactions, eventually reaching agreement [8]. In distributed gradient methods, each node updates its control variable based on local gradients and neighbor information, collectively optimizing a global cost function without central coordination. Such algorithms must balance convergence speed, communication overhead, and robustness to delays or packet loss.

Another crucial factor in cooperative allocation is the coupling between sensing, communication, and control layers. In many systems, the allocation of communication bandwidth itself must be optimized alongside sensing and actuation. For instance, in a distributed surveillance system, allocating more bandwidth to nodes detecting anomalies allows the network to focus attention where it matters most, improving detection accuracy without overloading the channel [9]. Similarly, in industrial networks, the timing and priority of control messages can be adjusted dynamically based on the state of the process, ensuring that critical commands are delivered promptly while less urgent updates are deferred. This tight integration between communication and control introduces additional layers of complexity but also opportunities for performance gains. Cross-layer optimization techniques explicitly consider these interactions, formulating joint problems that allocate not only actuation effort but also communication and computation resources to maximize system-level performance.

In practical implementations, cooperative resource allocation must also contend with uncertainties and time delays. Sensor measurements are noisy, actuator responses may lag, and communication channels may suffer from variable latency or packet drops. Robustness to such imperfections is essential for reliable operation [10]. Stochastic and adaptive control methods provide mechanisms to handle uncertainty by incorporating probabilistic models or online learning schemes that adjust parameters based on observed performance. For example, reinforcement learning techniques allow nodes to learn optimal allocation policies through trial and error, gradually improving coordination without requiring explicit system models. Similarly, model predictive control (MPC) approaches can be distributed across nodes, with each agent solving a local optimization problem over a receding horizon, using predictions of neighbor behavior to plan future actions. These advanced methods combine the rigor of optimization with the flexibility of adaptive learning, enabling systems to maintain efficient coordination even in dynamic or unpredictable environments.

Scalability remains a key motivation for distributed allocation [11]. As the number of nodes in a network grows, centralized computation and communication become bottlenecks, while distributed schemes can scale almost linearly if designed properly. However, scalability introduces challenges in maintaining convergence and avoiding oscillations or instability. Network topology plays a critical role: well-connected graphs enable faster information diffusion and convergence, while sparse or irregular topologies may slow coordination or even prevent consensus. Hierarchical architectures can offer a compromise, grouping nodes into clusters with local leaders that coordinate internally before interfacing with higher-level controllers. Such multi-layer structures combine the benefits of local autonomy and global coordination, making them suitable for large-scale infrastructures like smart grids or intelligent transportation systems.

Energy efficiency is another central concern in distributed sensor and actuator networks, particularly those deployed in remote or battery-powered settings [12]. Cooperative allocation algorithms must minimize not only communication costs but also actuation energy, which can dominate the overall power budget. Strategies such as duty cycling, adaptive sampling, and event-triggered control help conserve energy by allowing nodes to remain idle when system dynamics are slow or predictable. Event-triggered approaches, for instance, activate communication or control updates only when significant changes occur, reducing unnecessary transmissions while maintaining performance guarantees. By integrating such energy-aware mechanisms into the allocation process, networks can extend operational lifetimes without sacrificing control quality.

In contrast to rigid centralized systems, distributed networks can reconfigure themselves in response to failures, attacks, or environmental changes [13]. If a sensor or actuator goes offline, neighboring nodes can detect the loss and redistribute tasks accordingly. If communication links degrade, the network can reroute messages through alternative paths or temporarily operate in a degraded mode until connectivity is restored. This self-healing capability is particularly important in safety-critical applications such as disaster response, space exploration, or critical infrastructure management, where maintaining partial functionality may prevent catastrophic outcomes. Moreover, distributed architectures naturally lend themselves to incremental deployment and scaling, allowing new nodes to join the network without major reconfiguration.

Particle Swarm Optimization is attractive in this setting because it requires only function evaluations, accommodates nonconvexity, and lends itself to parallel execution across nodes. However, classical PSO relies on global best information or fully connected information exchange, which is impractical in multi-hop networks with intermittent connectivity and limited bandwidth [14]. Furthermore, naïve PSO may oscillate or stagnate when objective surfaces are noisy, constraints are tight, or inter-agent couplings are strong. To address these limitations, this study develops a consensus-driven PSO that couples the swarm dynamics with a mixing operator reflecting the communication graph. The approach keeps information exchange lightweight by propagating only low-dimensional particle states or compressed summaries, while enabling nodes to track locally consistent reference points. The analysis clarifies how consensus acts as a regularizer that smooths the search trajectory, reduces variance in the presence of stochastic evaluations, and maintains feasibility when projections onto local constraints are performed asynchronously.

The presentation begins with a system model and problem definition capturing heterogeneity and coupling [15]. A consensus-driven particle mechanism is then constructed by embedding mixing steps in the velocity-position updates and by augmenting the memory terms with neighborhood averaged states. Convergence and stability are studied through a composite potential involving both swarm dispersion and disagreement across nodes, yielding parameter ranges that ensure contraction in expectation. Complexity and communication aspects are addressed with attention to energy-aware operation, quantization, and privacy concerns. A simulation protocol is outlined to examine sensitivity to graph structure, noise, and heterogeneity, followed by implementation considerations oriented toward resource-constrained embedded deployments. The paper closes with a neutral synthesis of implications and limitations.

2. System Model and Problem Statement

Consider a distributed network with node set indexed by $i \in \{1, \dots, N\}$. Each node holds a local decision vector $x_i \in \mathbb{R}^{d_i}$ subject to a closed convex local feasible set $\mathcal{X}_i \subset \mathbb{R}^{d_i}$, and contributes a local cost $f_i : \mathbb{R}^{d_i} \to \mathbb{R}$ that may be nonconvex, nonsmooth, and noisy when evaluated from measurements. Coupling arises through shared

Symbol	Meaning	Domain	Role	Notes
x_i	Local decision vector	$\mathbb{R}^{d_i}$	Optimization variable	Node-specific control
$\mathcal{X}_i$	Local feasible set	Convex subset of $\mathbb{R}^{d_i}$	Constraint domain	Enforces physical limits
$f_i(x_i)$	Local cost function	$\mathbb{R}^{d_i} \to \mathbb{R}$	Objective term	May be nonconvex, noisy
A_i	Coupling matrix	$\mathbb{R}^{q \times d_i}$	Resource mapping	Aggregates shared quantities
$g(\cdot)$	Global penalty	$\mathbb{R}^q \to \mathbb{R}$	Coupling cost	Penalizes violations of budgets

Table 4: Notation summary for the distributed system model.

Parameter	Symbol	Description	Adaptivity	Effect
Inertia weight	$\omega_i[k]$	Scales previous velocity	Time-varying	Controls momentum
Cognitive gain	$\phi_{i,1}[k]$	Attraction to personal best	Adaptive	Affects exploration depth
Social gain	$\phi_{i,2}[k]$	Attraction to node best	Adaptive	Drives local convergence
Consensus gain	$\gamma_i[k]$	Coupling with $y_i[k]$	Graph-dependent	Enforces alignment
Step factor	$\alpha_i[k]$	Velocity-to-position ratio	Scheduled	Governs update rate

Table 5: Key update parameters in the consensus-driven PSO mechanism.

Term in Potential	Symbolic Form	Physical Interpretation	Contraction Role
Cognitive dispersion	$\sum_{i,p} \lVert x_i^{(p)} - b_i^{(p)} \rVert^2$	Spread around local memories	Aligns local minima
Social-consensus mismatch	$\sum_i P_i \lVert \hat{b}_i - y_i \rVert^2$	Disagreement across nodes	Enforces network coherence
Kinetic storage	$\lambda \sum_{i,p} \lVert v_i^{(p)} \rVert^2$	Inertial energy	Stabilizes oscillations
Exploration floor	$\kappa_3 \sum_{i,p} \mathbb{E}[\lVert \xi_i^{(p)} \rVert^2]$	Random excitation	Maintains diversity
Quantization bias	$\kappa_4 \sum_i \epsilon_Q^2$	Communication distortion	Limits precision

Table 6: Components of the composite Lyapunov potential used in convergence analysis.

physical resources, aggregate limits, or communication-constrained coordination [16]. Let $A_i \in \mathbb{R}^{q \times d_i}$ encode local contributions to q coupled quantities and let $g : \mathbb{R}^q \to \mathbb{R}$ capture penalties for violating aggregate budgets or for inducing congestion. The cooperative allocation problem is to choose $x = (x_1, \dots, x_N)$ to minimize a composite objective under constraints

$$\min_{x_1 \in \mathcal{X}_1, \dots, x_N \in \mathcal{X}_N} \quad F(x) = \sum_{i=1}^{N} f_i(x_i) + g\Big(\sum_{i=1}^{N} A_i x_i\Big) \tag{1}$$

$$\text{s.t.} \quad Cx \in \mathcal{S}, \qquad x \in \mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_N, \tag{2}$$

$$\text{communication over a graph } \mathcal{G}_k = (\{1, \dots, N\}, E_k) \text{ that varies with } k. \tag{3}$$

The constraint $Cx \in \mathcal{S}$ encodes safety and logical relations across nodes, potentially nonconvex but separable by slack variables or projections. The communication graph $\mathcal{G}_k$ evolves due to duty cycling, interference, or mobility, with a mixing matrix $W_k \in \mathbb{R}^{N \times N}$ that is row-stochastic, respects the instantaneous edges, and is assumed compatible with a jointly strongly connected graph sequence over windows of bounded length.

Heterogeneity manifests through dimension d_i, set geometry $\mathcal{X}_i$, evaluation noise on f_i, and possibly different local time steps and processor clocks. A resource allocation instance can involve energy budgets, duty-cycle constraints, and actuator saturations. Because gradients may be unavailable or unreliable, a derivative-free algorithm is of interest. The objective and constraints can be evaluated locally with limited shared information: each node can measure a local performance index and may estimate components of the aggregate coupling via neighbor messages or lightweight gossip.

We adopt a distributed particle-based search where each node maintains a small swarm of candidate allocations and exchanges compact summaries with neighbors [17]. The method must remain stable when communication delays occur, when packets are dropped, and when evaluation noise is nonstationary. The principal challenge is to preserve exploratory breadth while driving the collection toward local consistency that respects coupling and constraints. The approach therefore augments particle dynamics with a consensus mechanism whose role is to align local reference states without forcing premature collapse.

3. Consensus-Driven PSO Mechanism

Each node i maintains P_i particles, with positions $x_i^{(p)}[k] \in \mathcal{X}_i$ and velocities $v_i^{(p)}[k] \in \mathbb{R}^{d_i}$ at iteration k. Local personal bests are $b_i^{(p)}[k]$, updated when the local evaluation decreases the node's best-so-far, and a node-level best $\hat{b}_i[k]$ summarizes the best among its particles. Communication induces a neighborhood-aggregated reference $y_i[k]$ obtained by mixing node-level summaries over edges of $\mathcal{G}_k$. The consensus-driven position and velocity updates incorporate inertia, cognitive and social terms, a consensus regularizer, a projection onto feasibility, and small noise for exploration:

$$\begin{aligned} v_i^{(p)}[k+1] &= \omega_i[k]\, v_i^{(p)}[k] \\ &\quad + \phi_{i,1}[k]\, r_{i,1}^{(p)}[k] \odot (b_i^{(p)}[k] - x_i^{(p)}[k]) \\ &\quad + \phi_{i,2}[k]\, r_{i,2}^{(p)}[k] \odot (\hat{b}_i[k] - x_i^{(p)}[k]) + \gamma_i[k](y_i[k] - x_i^{(p)}[k]) \\ &\quad + \xi_i^{(p)}[k] - \Pi_{\mathcal{N}_{\mathcal{X}_i}(x_i^{(p)}[k])}(v_i^{(p)}[k]), \end{aligned} \tag{4}$$

Here $\omega_i[k]$ is an inertia weight possibly scheduled with time or local variance estimates, $\phi_{i,1}[k], \phi_{i,2}[k]$ scale cognitive and social drifts, and $\gamma_i[k]$ penalizes disagreement with the consensus reference $y_i[k]$. The term $\xi_i^{(p)}[k]$ is small zero-mean excitation with bounded covariance to maintain exploration, $\alpha_i[k]$ is a step factor, and $\Pi_{\mathcal{N}_{\mathcal{X}_i}}(\cdot)$ denotes a velocity correction that dampens motion along the normal cone at the boundary, stabilizing projections near constraints. The consensus reference is computed via a mixing stage with optional compression $\mathcal{Q}$:

$$\begin{aligned} s_i[k] &= \mathcal{Q}\big(\hat{b}_i[k]\big) \\ \bar{s}_i[k] &= \sum_{j=1}^{N} W_k(i,j)\, s_j[k] \\ y_i[k] &= \mathcal{D}\big(\bar{s}_i[k]\big). \end{aligned} \tag{5}$$

where $\mathcal{Q}$ and $\mathcal{D}$ represent quantization and dequantization maps, possibly identity. The mixing matrix W_k is row-stochastic, satisfies $W_k(i,j) = 0$ when $(i,j) \notin E_k$, and is chosen to have a uniform spectral gap across time windows. The use of $\hat{b}_i[k]$ in the consensus step ensures that nodes propagate their strongest evidence without divulging local cost functions.

Coupling and feasibility require that aggregate constraints be maintained or approximately respected during the search [18]. We introduce a soft-penalty surrogate for g that can be evaluated locally from shared partial sums or tracked via gossip. If $z[k] \approx \sum_i A_i \hat{b}_i[k]$ is a running estimate of coupling, a local penalty term can be injected into the evaluation of candidates so that personal and node-level bests reflect both local and coupled effects. The mechanism thereby aligns local improvement with global feasibility while preserving the derivative-free nature of the search.

To mitigate sensitivity to graph changes and delays, each node maintains a buffer of recent references and applies an exponential smoother to $y_i[k]$. This damping allows the PSO dynamics to absorb transient topology fluctuations. Heterogeneous parameter schedules $\omega_i[k], \phi_{i,1}[k], \phi_{i,2}[k], \gamma_i[k]$ can be adapted based on variance of recent evaluations and norm of velocity updates, enabling nodes with higher noise to retain higher inertia while nodes with stable evaluations lean more on consensus.

4. Convergence Guarantees and Stability

The analysis proceeds by introducing a composite potential that measures both dispersion of particles around their personal and neighborhood references and disagreement among nodes [19]. Let the vector $x[k]$ concatenate all particles at time k, and define a disagreement metric based on the mixing operator. Consider

$$\begin{aligned} \mathcal{V}[k] &= \underbrace{\sum_{i=1}^{N}\sum_{p=1}^{P_i} \big\|x_i^{(p)}[k] - b_i^{(p)}[k]\big\|^2}_{\text{cognitive dispersion}} \\ &\quad + \underbrace{\sum_{i=1}^{N} P_i \big\|\hat{b}_i[k] - y_i[k]\big\|^2}_{\text{social-consensus mismatch}} \\ &\quad + \underbrace{\lambda \sum_{i=1}^{N}\sum_{p=1}^{P_i} \big\|v_i^{(p)}[k]\big\|^2}_{\text{kinetic storage}}. \end{aligned} \tag{6}$$

with $\lambda > 0$. The mixing operator contracts disagreement in the subspace orthogonal to the consensus vector; denote $\tilde{y}[k]$ as the stack of $y_i[k]$ and let $L_k = I - W_k$. For row-stochastic W_k, the smallest nonzero singular value of L_k over a window controls the decay of disagreement. Under bounded quantization error and independent zero-mean exploration noise, there exists a range of parameters such that the expected one-step change in $\mathcal{V}[k]$ is negative away from an invariant set associated with approximate stationarity:

$$\begin{aligned}\mathbb{E}[\mathcal{V}[k+1] \mid \mathcal{F}_k] \le \mathcal{V}[k] \\ \underbrace{- \kappa_1 \sum_{i,p} \left\| x_i^{(p)}[k] - y_i[k] \right\|^2}_{\text{consensus contraction}} \\ \underbrace{- \kappa_2 \sum_{i,p} \left\| x_i^{(p)}[k] - b_i^{(p)}[k] \right\|^2}_{\text{cognitive alignment}} \\ \underbrace{+ \kappa_3 \sum_{i,p} \mathbb{E}\big[\|\xi_i^{(p)}[k]\|^2 \mid \mathcal{F}_k\big]}_{\text{exploration floor}} \\ + \underbrace{\kappa_4 \sum_i \epsilon_Q^2}_{\text{quantization bias}} \\ + \underbrace{\kappa_5 \sum_{i,p} \Delta_{\text{proj}}(x_i^{(p)}[k])}_{\text{projection residuals}} .\end{aligned} \tag{7}$$

with nonnegative coefficients $\kappa_1, \ldots, \kappa_5$ determined by $\omega_i, \phi_{i,1}, \phi_{i,2}, \gamma_i, \alpha_i$ and by spectral properties of $\{W_k\}$. The term ϵ_Q bounds quantization error and Δ_{proj} accounts for nonexpansive projection deviations. If the spectral gap of the time-averaged mixing is at least $\sigma > 0$, there exist parameter ranges $\mathcal{R}(\sigma)$ such that κ_1, κ_2 dominate the additive floor. Consequently, the process is stochastically bounded and enters a neighborhood of an invariant set whose radius scales with exploration noise, quantization resolution, and projection residuals. [20]

Feasibility and aggregate coupling can be controlled via penalty continuation.

Let $F_\rho(x) = \sum_i f_i(x_i) + g_\rho(\sum_i A_i x_i)$ with a smooth penalty g_ρ that increases with ρ. As ρ increases, the invariant set concentrates near approximate stationary points of the constrained problem when constraint violations are small relative to the penalty slope. The role of consensus is to align the measures that enter g_ρ: if $z[k]$ tracks $\sum_i A_i \hat{b}_i[k]$, the difference between local evaluations and global penalties can be bounded by a term that depends on the mixing error and on the Lipschitz constant of g_ρ. Under bounded delays and sufficiently frequent connectivity, the tracking error remains small, and the invariant set contracts accordingly.

To formalize the contraction in the presence of time-varying graphs and delays, introduce the disagreement vector $e[k]$ defined by stacking $e_i[k] = \hat{b}_i[k] - \bar{s}_i[k]$. Assuming row-stochastic W_k with uniformly bounded second-largest singular value $\sigma_2 < 1$ over windows of length T, it follows that

$$\|e[k+T]\| \le \sigma_2 \|e[k]\| + \delta_Q + \delta_{\text{del}}, \tag{8}$$

where δ_Q bounds quantization accumulation and δ_{del} bounds delay-induced perturbations. This disagreement bound, combined with the nonexpansive projection and the damped velocity updates, yields a supermartingale sequence $\{\mathcal{V}[k]\}$ perturbed by bounded noise. By Robbins–Siegmund-type arguments, one obtains almost sure convergence of the summable contraction terms and convergence in probability of allocations to an ε-neighborhood whose size depends continuously on noise, quantization, and spectral gap. [21]

A different perspective views the consensus-driven PSO as a stochastic fixed-point iteration on a set-valued drift map that includes random perturbations and projections. Let the drift be

$$\mathcal{T}(x) = \text{Proj}_{\mathcal{X}}[x + \alpha\Phi(x,\eta) + \Gamma(Y(x) - x)], \tag{9}$$

where Φ encodes cognitive and social terms with randomness η, Y maps current states to consensus references via mixing, and Γ is a block-diagonal gain. For small step sizes and under averaged contraction of Y in the orthogonal complement of consensus, the composed map is contractive in expectation on a suitable metric space. This yields mean-square boundedness and, when exploration diminishes, convergence to a random fixed point associated with local stationary conditions [22]. The statement does not claim global optimality; rather, it quantifies stability and proximity to stationary allocations subject to inevitable nonconvexity and noise.

5. Complexity, Communication, and Privacy Considerations

Execution complexity per node depends on particle count, evaluation cost, and communication schedule. Assuming P_i particles of dimension d_i, the arithmetic per iteration scales as $O(P_i d_i)$ for velocity-position updates plus the local evaluation time of f_i, which may dominate when evaluations are empirical or simulated. Memory scales as $O(P_i d_i)$ to store positions, velocities, and personal bests. The consensus step requires exchanging summaries of dimension d_i or compressed representations; the total neighbor traffic per iteration is $O(\sum_{j\in\mathcal{N}_i} d_i)$ in the uncompressed case and is reduced by the rate of $\mathcal{Q}$ under quantization. Across the network, aggregate communication is proportional to the sum of degrees times the exchanged dimension, which is typical for gossip-style protocols.

Bandwidth constraints necessitate quantization or sparsification [23]. With uniform quantization of B bits per coordinate, the mean squared error scales as $O(2^{-2B})$, and its effect on steady-state performance enters through additive floors in the potential descent inequalities. When communication links are unreliable, nodes may employ packet-level acknowledgment to update mixing weights or rely on push-sum variants that preserve mass conservation in expectation. To limit energy consumption, nodes can adopt event-triggered exchange, sending updates only when the change in $\hat{b}_i[k]$ exceeds a threshold. Such triggering reduces communication at the cost of increased disagreement variance, which can be compensated by slightly larger consensus gains $\gamma_i[k]$ or by occasional forced resynchronization rounds.

Privacy considerations arise because nodes may be unwilling to disclose cost structure or raw measurements. The mechanism shares only candidate states or compressed aggregates, revealing limited information about internal objectives. Noise injection in the shared states can further obscure sensitive details, though this inevitably increases disagreement and slows convergence [24]. When actuator commands are sensitive, nodes can share masked states via local random rotations or additively homomorphic encodings over small groups, but these methods induce overhead and, in this study, are treated as optional refinements rather than requirements. The baseline protocol already avoids broadcasting objective values.

Asynchronous operation is essential in heterogeneous systems. Nodes can proceed with stale references $y_i[k-\tau]$ when neighbor updates are delayed, provided the staleness τ remains bounded on average. The stability analysis tolerates bounded delays via slack terms in the contraction bounds, and empirical evidence suggests that mild asynchrony is often beneficial by diversifying exploration. Time-varying schedules allow low-energy nodes to downscale particle counts or decrease update frequency [25]. In practice, energy-aware adaptation yields modest savings of 5%–15% in communication energy, with little degradation in allocation quality when thresholds are tuned conservatively and gains are scaled to compensate for the additional noise due to sporadic mixing.

6. Simulation Protocol and Empirical Observations

To evaluate behavior across heterogeneous conditions, one can construct scenarios with mixed sensing and actuation capabilities, variable communication topologies, and nonconvex local objectives. A representative setup involves N nodes partitioned into classes with differing d_i, constraints $\mathcal{X}_i$, and evaluation noise models. Graphs range from ring-like sparse structures to random geometric graphs with varying density and small-world augmentations. The coupling term models shared actuation budgets and network congestion through simple capacity penalties. PSO parameters are selected within the ranges identified by the analysis, and quantization is applied to study sensitivity to bandwidth. [26]

Empirically, the consensus-driven mechanism tends to produce smoother allocation trajectories with reduced variance in local feasibility violations. When the spectral gap of the mixing process is moderate, steady-state disagreement decays quickly after transients, and the particle clouds concentrate near consistent references while maintaining enough spread for exploration. In contrast, disabling consensus increases susceptibility to topological fragmentation; local bests become misaligned across components and aggregate constraint violations escalate. Exploration noise stimulates escape from poor allocations, but excessive noise leads to oscillation near the boundary of feasibility; gradually diminishing the noise floor improves steadiness without entirely suppressing diversity.

Heterogeneity in particle counts and dimensions influences convergence speed primarily through evaluation throughput. Nodes with larger d_i benefit from slightly higher inertia and an increased consensus gain to counteract the tendency toward slow local exploration [27]. Event-triggered communication reduces load significantly in dense graphs, with typical reductions of 20%–40% in messages when tolerances are chosen so that stale references are refreshed before divergence becomes noticeable. Quantization with moderate bit depths rarely degrades performance severely; the additive floors in the contraction bounds remain small compared to the natural variability induced by evaluation noise. Under severe bandwidth restrictions, projecting shared states onto low-dimensional subspaces maintains acceptable behavior if the subspaces align with dominant directions of variability in the particle clouds.

The penalty continuation for coupling constraints yields reduced aggregate violations over time. Early in the search, softer penalties allow broader exploration even if constraints are temporarily exceeded; later, increased penalty weights bias the swarm toward feasible neighborhoods [28]. Monitoring of aggregate quantities through gossip-tracked estimates is effective when the graph remains jointly connected. When connectivity is intermittent, maintaining a conservative safety margin in penalties is advisable because local estimates lag, especially in sparse components. Despite these challenges, the cooperative mechanism exhibits stable allocation under diverse conditions, with variance that reflects the tunable balance between exploration and agreement.

7. Ablations, Sensitivity, and Robustness

Ablation of the consensus term isolates its role in stabilizing the social drift. Removing the consensus gain while retaining cognitive and social terms relative to node-level bests typically increases dispersion and slows agreement across nodes. The effect is more pronounced when graphs are sparse or partitioned; without consensus mixing, social information does not propagate beyond immediate neighborhoods [29]. Reintroducing a small consensus gain restores coherence with modest communication overhead. Ablation of the velocity correction near constraints likewise reveals its contribution to stable feasibility. Without normal-cone damping, projections can induce bouncing near the boundary, and velocities retain momentum incompatible with feasible motion; damping attenuates this behavior, improving stationarity of feasible allocations.

Sensitivity to parameter schedules appears smooth across moderate ranges. Inertia governs the persistence of motion; too large values produce oscillatory behavior, whereas too small values stall exploration [30]. Cognitive and social gains regulate the attraction toward personal and node-level memories. The consensus gain mitigates disagreement but, if excessive, collapses diversity and impedes escape from local minima. Exploration noise supports global search, but its covariance must be scaled to local dimension and constraint curvature to avoid frequent projections. Scheduling exploration to decay slowly with iteration count tends to reduce late-stage jitter, with marginal loss in the ability to escape shallow local traps.

Robustness to delays and packet loss is acceptable under bounded staleness. When delays increase, disagreement floors rise and the benefits of consensus shrink; adaptive gain reduction can compensate by preventing overreaction to stale references [31]. Quantization introduces bias that accumulates if the network is highly asymmetric; dithering or stochastic rounding reduces such bias at the cost of variance. Under adversarial link failures, the protocol degrades gracefully when redundancy in connectivity exists; catastrophic fragmentation leads to independent sub-swarms, which is a natural limitation of any gossip-based strategy. Privacy-preserving noise on shared states can mask sensitive information while leaving allocation quality largely intact if the injected noise is small relative to consensus contraction.

Noise in objective evaluations, common in sensor-actuator loops, affects best-memory updates and can induce false improvements. A simple stabilization is to require persistent improvement over a small window or to adopt bootstrapped estimates of the expected cost before updating memories [32]. These strategies reduce sensitivity to outliers with minimal additional computation. Overall, the ablations and sensitivity analysis indicate that each mechanism—consensus, damping near constraints, quantization management, and conservative memory updates—contributes incremental robustness without materially complicating implementation.

8. Implementation Notes and Practical Guidelines

Implementation in embedded and edge environments benefits from careful attention to data layout, arithmetic precision, and timing. Fixed-point arithmetic can be used for velocity and position updates when quantization is already present in communication; choosing scales so that dynamic range accommodates the largest expected velocities prevents saturation artifacts. Memory locality improves throughput for particle updates, suggesting contiguous storage of particle states and velocities. Scheduling consensus exchanges at regular submultiples of the update period reduces contention on shared links and aligns with duty-cycling policies. [33]

Parameter initialization can draw on local variance estimates of initial evaluations. For nodes with high noise, slightly larger inertia and lower cognitive gains preserve exploratory motion without frequent oscillations. Consensus gains can be set proportional to the estimated spectral gap when available; in practice, a conservative default based on the maximum degree or on rudimentary connectivity estimates suffices. Penalty continuation for coupling constraints benefits from a warm start with lower weights to encourage exploration, increasing weights when aggregate violations remain persistent over several windows. Exploration noise can be Gaussian with covariance scaled to the recent average squared velocity, yielding a form of annealing that reacts to the current dynamical regime. [34]

Fault tolerance requires that nodes detect missing neighbor updates and age out stale references. Time stamps accompanying exchanged summaries enable discard of outdated messages. When nodes wake from low-power states, they can rejoin by requesting a short resynchronization burst from neighbors. If identification is an issue, hash-based tags on shared states detect accidental reuse of stale buffers. Security can be enhanced by lightweight integrity checks at the cost of additional bytes per message. In dense deployments, limiting neighbor lists to a small set of rotating peers reduces degree while preserving mixing over longer windows. [35]

On platforms with heterogeneous compute capacity, adapting particle counts per node produces a balanced budget. Nodes with more capable processors maintain larger swarms and share proportionally more informative summaries; lighter nodes maintain few particles while still benefiting from consensus with stronger peers. The mechanism tolerates such imbalance because the consensus step mixes node-level bests rather than raw particle clouds. In practice, mixed-capability networks show improvements of 8%–20% in wall-clock time to a given allocation quality relative to uniform particle counts, primarily due to better utilization of high-capability nodes.

9. Modeling Identities and Inequalities Used in the Analysis

The analysis relies on algebraic expansions of velocity-position coupling, properties of nonexpansive projections, and contraction of mixing in disagreement subspaces [36]. The following identities and inequalities organize the argument into verifiable components. For any vectors a, b and scalar $\theta \in \mathbb{R}$,

$$\|a+\theta b\|^2 = \|a\|^2 + 2\theta\langle a, b\rangle + \theta^2\|b\|^2$$
$$\|\mathrm{Proj}_{\mathcal{X}}(a) - \mathrm{Proj}_{\mathcal{X}}(b)\| \le \|a-b\|.\text{[37]} \quad (10)$$

Let $\mathcal{W}$ denote the set of row-stochastic matrices respecting the graph, with stationary distribution π. For any $W \in \mathcal{W}$, let $J = \mathbf{1}\pi^\top$. The disagreement projector is $I - J$. For any state stack s,

$$\|(W-J)s\| \le \sigma_2(W)\,\|(I-J)s\|,$$
$$\sigma_2(W) < 1.\text{[38]} \quad (11)$$

Cascading over a window yields

$$\Big\|\Big(\prod_{\ell=0}^{T-1} W_{k+\ell} - J\Big)s\Big\| \;\le\; \Big(\prod_{\ell=0}^{T-1}\sigma_2(W_{k+\ell})\Big)\,\|(I-J)s\|. \quad (12)$$

Velocity updates decompose into deterministic drift and noise. Writing the stacked velocity as $v[k{+}1] = \Omega v[k] + \Phi(x[k], b[k], y[k]) + \Xi[k]$ with block-diagonal Ω constructed from $\omega_i I$, one expands the kinetic term as

$$\begin{aligned}\|v[k{+}1]\|^2 = \|\Omega v[k]\|^2 \\ +\; 2\langle\Omega v[k], \Phi\rangle \\ +\; \|\Phi\|^2 \\ \text{[39]} +\; 2\langle\Omega v[k] + \Phi, \Xi[k]\rangle \\ +\; \|\Xi[k]\|^2. \quad (13)\end{aligned}$$

and takes conditional expectation to eliminate cross-terms with zero-mean $\Xi[k]$. The cognitive and social terms contribute negative inner products with dispersion measures under appropriate gain selection. Projection residuals admit the bound

$$\big\|\mathrm{Proj}_{\mathcal{X}}(x+\alpha v) - (x+\alpha v)\big\| \;\le\; \alpha\,\mathrm{dist}(v, \mathcal{T}_{\mathcal{X}}(x)), \quad (14)$$

with $\mathcal{T}_{\mathcal{X}}(x)$ the tangent cone, implying that velocity alignment with feasible directions reduces residuals.

Aggregating these elements produces a drift inequality for $\mathcal{V}[k]$ of the form

$$\begin{aligned}\mathbb{E}[\mathcal{V}[k{+}1] - \mathcal{V}[k] \mid \mathcal{F}_k] \le -\mu_1\sum_{i,p}\|x_i^{(p)}[k] - y_i[k]\|^2 \\ -\;\mu_2\sum_{i,p}\|x_i^{(p)}[k] - b_i^{(p)}[k]\|^2 \\ +\mu_3\sum_{i,p}\mathbb{E}\|\xi_i^{(p)}[k]\|^2 + \mu_4\epsilon_{\mathcal{Q}}^2 + \mu_5\sum_{i,p}\Delta_{\mathrm{proj}}(x_i^{(p)}[k]).\end{aligned} \quad (15)$$

for suitable $\mu_1, \ldots, \mu_5$ [40]. Summing over k and invoking martingale convergence theorems yields finiteness of the accumulated disagreement and dispersion terms and convergence in probability to a neighborhood of an invariant set, with radius proportional to the square root of the total perturbation level. The dependence on spectral gap arises through μ_1, indicating that better-connected graphs enforce stronger contraction of disagreement.

Ablation Target	Removed Component	Observed Effect	Impact on Stability
Consensus term	$\gamma_i(y_i - x_i)$	Increased dispersion, slower agreement	Loss of coherence
Velocity damping	$\Pi_{\mathcal{N}_{\mathcal{X}_i}}(v_i)$	Boundary oscillation, infeasible motion	Instability near constraints
Quantization management	$\mathcal{Q}/\mathcal{D}$	Accumulated bias	Bias growth
Memory stabilization	Persistence test	False updates under noise	Reduced robustness

Table 7: Ablation summary highlighting functional contributions of key components.

Parameter	Low Setting	High Setting	Guideline
Inertia ω	Rapid damping	Persistent oscillation	Moderate (0.6–0.9)
Cognitive gain ϕ_1	Weak exploration	Overfitting to memory	Tune by variance
Social gain ϕ_2	Slow convergence	Premature consensus	$\phi_2/\phi_1 \approx 1$
Consensus gain γ	Fragmentation	Loss of diversity	Small, adaptive
Noise variance Σ_ξ	Poor global search	Frequent projections	Scale to dimension

Table 8: Parameter sensitivity and tuning recommendations based on empirical analysis.

Perturbation Source	Mitigation Strategy	Residual Effect	Performance Impact
Communication delay	Gain adaptation, smoothing	Elevated disagreement floor	Minor under $\leq$3-step delay
Packet loss	Redundant mixing, buffering	Stochastic desync	Graceful degradation
Quantization error	Stochastic rounding	Small bias	Negligible for 8–10 bits
Privacy noise	Bounded Gaussian mask	Slight variance inflation	$< 2\%$ quality drop
Node failure	Local reallocation, resync burst	Transient disruption	Recovery within few cycles

Table 9: Robustness to communication and system-level perturbations in distributed operation.

10. Extended Formulation for Coupling and Penalty Continuation

Coupling constraints can be modeled by a penalty g_ρ with scale $\rho > 0$ acting on an aggregate $z = \sum_i A_i x_i$. Let $g_\rho(z) = \rho h(z)$, where h is a smooth convex surrogate of a constraint indicator. Nodes track $z[k]$ through local estimates $z_i[k]$ updated via mixing, and candidates are evaluated against $f_i(x_i) + \rho h(z_i + A_i x_i - A_i \hat{b}_i)$ such that double counting is reduced. The tracking obeys [41]

$$z_i[k+1] = \sum_j W_k(i,j)\, z_j[k] + \beta_i[k]\big(A_i\hat{b}_i[k] - A_i\hat{b}_i[k-1]\big). \quad (16)$$

a dynamic that maintains an unbiased estimate of the aggregate under connectivity assumptions. The incremental form limits message size and supports compression. The role of $\beta_i[k]$ is to correct for local changes quickly while allowing mixing to remove bias from stale information.

The continuation schedule for ρ increases only when aggregate violations remain below a tolerance for several windows. Let $u[k] = \max\{0, c - \|z[k]\|_{\mathcal{S}}\}$ denote a violation metric relative to the set $\mathcal{S}$. A simple adaptive rule sets

$$\rho[k+1] \;=\; \rho[k]\big(1 + \delta_\rho \cdot \mathbf{1}\{\bar{u}[k-H{:}k] < \tau\}\big), \quad (17)$$

with $\bar{u}$ a running average over a window H and threshold τ. This rule behaves conservatively: if violations rise due to network fragmentation or noise bursts, ρ remains fixed, allowing exploration to reestablish feasibility before further tightening [42]. The theoretical bounds accommodate fixed ρ; adaptivity enters the analysis by treating $\rho[k]$ as piecewise constant over windows, introducing small jumps that do not affect stability provided updates are infrequent relative to convergence rates.

To connect with stationarity, consider first-order necessary conditions of a smoothed problem and write a residual that the swarm implicitly tries to reduce through its memory-guided exploration and consensus alignment. With a proximal surrogate for constraints, the fixed-point residual at x under step α can be formalized as

$$R_\alpha(x) \;=\; \frac{1}{\alpha}\Big(x - \mathrm{Proj}_{\mathcal{X}}\big[x - \alpha\nabla\tilde{F}_\rho(x)\big]\Big), \quad (18)$$

where $\tilde{F}_\rho$ is a smooth surrogate obtainable by local smoothing of nonsmooth terms. Although PSO does not use gradients, empirical reductions in $R_\alpha(x)$ can be observed by finite-difference probes or by surrogate gradient estimates assembled from swarm statistics. Consensus coupling aids in reducing $\|R_\alpha(x)\|$ by aligning nodes toward shared references that better respect the aggregate penalty landscape. [43]

11. Conclusion

This study has presented and analyzed a consensus-driven Particle Swarm Optimization (PSO) framework tailored for cooperative resource allocation in heterogeneous distributed sensor and actuator networks. The approach reimagines traditional PSO dynamics by embedding a consensus mechanism that facilitates information sharing and coordination among nodes while respecting the distributed nature of the system. Classical PSO algorithms rely on the exchange of best-known positions within a population of particles to guide search behavior, but when applied to sensor and actuator networks, several unique challenges arise—chief among them being communication constraints, heterogeneity of nodes, and time-varying connectivity. To address these challenges, the proposed consensus-driven PSO augments each particle's update rule with a neighborhood mixing step. This step serves to regularize the social component of the algorithm, ensuring that information from neighboring nodes is incorporated in a manner that stabilizes exploration and prevents premature convergence in the presence of noise. By introducing this local consensus mechanism, the algorithm achieves greater coherence across distributed agents and maintains stable operation even when communication links fluctuate or when sensing data are uncertain or incomplete. [44]

The neighborhood mixing process plays a critical role in moderating how particles respond to social information. In standard PSO, each agent updates its velocity and position based on a linear combination of its own experience and that of a global or local best particle. However, in distributed systems with limited communication, the notion of a global best may be inaccessible, and unrestricted information exchange may be infeasible. The consensus step replaces this centralized dependency with a decentralized averaging process in which each node blends its current state with those of its neighbors according to the underlying communication graph. This approach effectively smooths discrepancies among neighboring estimates, dampens oscillations caused by noisy feedback, and leads to a more robust and stable search trajectory [45]. As a result, the swarm evolves not only toward high-quality allocations but also toward mutual agreement among nodes, a property that is especially valuable when network connectivity changes over time or when nodes experience intermittent communication.

The theoretical foundation of the proposed method rests on a composite potential function framework. This analytical tool establishes conditions under which the collective dynamics of the swarm remain stable and bounded. Specifically, a mean-square boundedness result guarantees that the variance of particle states does not diverge even in the presence of stochastic perturbations, quantization effects, or communication noise. Furthermore, a contraction argument demonstrates that, in expectation, the swarm trajectories converge toward an invariant set whose radius depends on measurable quantities such as exploration noise intensity, the magnitude of quantization and projection residuals, and the spectral properties of the time-varying communication process. These results formalize the intuition that the more coherent and better connected the network is, the smaller the steady-state dispersion among particles becomes [46]. The analysis thus provides insight into the trade-offs between exploration and consensus, as well as between noise resilience and convergence precision. By quantifying how communication quality and parameter settings affect the algorithm's limiting behavior, the study contributes a rigorous understanding of distributed PSO under realistic network conditions.

An important feature of this formulation is its ability to handle aggregate coupling constraints—situations where individual actuation efforts are interdependent due to shared resources or global system limits. Examples include total power constraints in smart grids or aggregate flow limits in distributed fluid systems. These couplings are incorporated into the optimization framework through a combination of penalty continuation and gossip-based tracking techniques [47]. The penalty continuation approach introduces auxiliary terms into the objective function that penalize constraint violations, gradually tightening these penalties as the swarm converges. Meanwhile, the gossip-based tracking mechanism enables distributed estimation of aggregate quantities, allowing nodes to maintain constraint feasibility using only local communication. This hybrid treatment preserves the derivative-free nature of PSO, meaning that nodes need not compute or exchange gradient information—a significant advantage in settings where objective functions are complex, nonconvex, or noisy. In doing so, the proposed method achieves a balance between practical implementability and theoretical rigor, extending the classical PSO paradigm into domains where both cooperation and constraint satisfaction are essential.

From a computational perspective, the consensus-driven PSO remains lightweight and scalable. The dominant cost arises from local evaluations of the objective or fitness function, which are inherently parallelizable since each node performs them independently [48]. Communication costs are kept low by limiting exchanges to compact state summaries—typically, current position, velocity, or local best information—rather than raw sensor data or detailed model parameters. The consensus updates themselves require only simple linear operations, making the algorithm suitable for low-power microcontrollers and embedded systems commonly found in sensor networks. This computational efficiency allows the approach to scale gracefully with network size, supporting large deployments without centralized bottlenecks or excessive communication overhead. Moreover, because the algorithm is derivative-free, it can be applied to systems with discontinuous, noisy, or black-box performance metrics, further broadening its practical applicability.

The study provides several practical guidelines for parameter tuning and implementation to ensure reliable and efficient performance [49]. Chief among these is the careful scheduling of exploration and consensus parameters. High exploration rates encourage global search and prevent premature convergence, but they must be balanced with sufficient consensus weighting to maintain coherence across the network. Event-triggered communication is recommended to enhance bandwidth efficiency; rather than exchanging information at every iteration, nodes communicate only when significant changes occur in their local state or when the expected improvement in coordination exceeds a threshold. This adaptive communication policy reduces unnecessary transmissions, conserving energy and bandwidth while preserving the essential flow of information. Additionally, the study advocates for conservative memory updates in which nodes incorporate new information gradually rather than abruptly replacing old estimates. This strategy mitigates the destabilizing influence of noisy or unreliable evaluations, leading to smoother convergence and more robust performance. [50]

Empirical analyses, including sensitivity experiments and ablation studies, further illuminate the benefits of the proposed consensus-driven PSO. These experiments systematically vary key parameters and algorithmic components to reveal their individual contributions to performance. The inclusion of consensus dynamics, for instance, is shown to significantly enhance stability and coherence, especially under time-varying connectivity and noise. Damping mechanisms near constraints help prevent oscillatory behavior and maintain feasibility, while explicit quantization management—through careful rounding and message compression schemes—reduces communication load without sacrificing accuracy. The results also demonstrate that the algorithm degrades gracefully under adverse conditions such as delays, packet loss, and bandwidth limitations [51]. Even when network reliability deteriorates, the distributed consensus structure allows nodes to recover and reestablish coordination once communication resumes. This resilience underscores the algorithm's suitability for deployment in real-world networks where imperfections are inevitable.

Despite these encouraging results, the study adopts a measured stance regarding theoretical guarantees of global optimality. Like most derivative-free, population-based optimization methods, PSO does not guarantee convergence to the global optimum, especially in nonconvex or noisy environments where multiple local minima may exist. Instead, the emphasis is placed on achieving consistent and robust performance across a range of practical conditions. The consensus mechanism improves reliability and repeatability by reducing stochastic variability across runs, but it does not eliminate the inherent limitations of heuristic search [52]. The analysis thus acknowledges that the method's convergence properties are probabilistic and dependent on factors such as problem landscape, parameter settings, and communication topology. Nonetheless, within these bounds, the consensus-driven PSO provides a principled approach to distributed optimization that balances exploration, cooperation, and feasibility in a computationally efficient manner.

From a systems perspective, the integration of consensus principles into particle dynamics represents an elegant unification of ideas from control theory, networked systems, and swarm intelligence. The approach blends the adaptive and exploratory strengths of PSO with the stabilizing and coordinating power of consensus algorithms, yielding a hybrid method capable of addressing challenges that neither framework could solve alone. In distributed sensor and actuator networks, where partial observability, heterogeneous constraints, and asynchronous communication are the norm, this synthesis provides a natural fit [53]. The algorithm leverages local interactions to produce emergent global behavior, ensuring that nodes collectively approximate the solution to a shared optimization problem without centralized control. This emergent coordination mirrors patterns observed in natural swarms—such as flocking birds or schooling fish—where global order arises from simple local rules, further reinforcing the biological inspiration behind swarm-based optimization.

The proposed consensus-driven PSO framework could be adapted to a wide range of distributed optimization and control problems, including cooperative target tracking, distributed estimation, and networked decision-making under uncertainty. Its communication-efficient design makes it suitable for wireless sensor networks, Internet of Things (IoT) platforms, and other resource-constrained environments. Furthermore, the analytical insights regarding mean-square stability and invariant set contraction provide a valuable foundation for extending similar techniques to other stochastic and consensus-based optimization algorithms. By demonstrating that convergence and stability can be achieved under realistic conditions—without sacrificing the simplicity and generality of derivative-free search—the study bridges the gap between theoretical rigor and engineering practicality. [54]

References

[1] K. Li, C. E. Torres, K. Thomas, L. F. Rossi, and C.-C. Shen, "Slime mold inspired routing protocols for wireless sensor networks," *Swarm Intelligence*, vol. 5, pp. 183–223, 11 2011.

[2] P. Pattanayak, D. Sarmah, S. K. Mishra, and A. K. Panda, "Computationally efficient scheduling methods for mimo uplink networks," *Soft Computing*, vol. 25, pp. 11763–11780, 6 2021.

[3] E. M. Arkin, M. A. Bender, S. P. Fekete, J. S. B. Mitchell, and M. Skutella, "The freeze-tag problem: How to wake up a swarm of robots," *Algorithmica*, vol. 46, pp. 193–221, 7 2006.

[4] B. Blobel, P. Ruotsalainen, and M. Brochhausen, "phealth - autonomous systems and artificial intelligence - hype or prerequisite for p5 medicine?," *Studies in health technology and informatics*, vol. 285, pp. 3–14, 10 2021.

[5] E. M. Nasroullahi and K. Tumer, "Combining coordination mechanisms to improve performance in multi-robot teams," *Artificial Intelligence Research*, vol. 1, pp. 1–10, 8 2012.

[6] A. Laan, G. Madirolas, and G. G. de Polavieja, "Rescuing collective wisdom when the average group opinion is wrong," *Frontiers in Robotics and AI*, vol. 4, pp. 56–, 11 2017.

[7] G. Stein and A. J. Gonzalez, "Building and improving tactical agents in real time through a haptic-based interface," *Journal of Intelligent Systems*, vol. 24, pp. 383–403, 12 2015.

[8] P. Garcia-Aunon, J. del Cerro, and A. Barrientos, "Behavior-based control for an aerial robotic swarm in surveillance missions.," *Sensors (Basel, Switzerland)*, vol. 19, pp. 4584–, 10 2019.

[9] H. Kaja and C. Beard, "A multi-layered reliability approach in vehicular ad-hoc networks," *International Journal of Interdisciplinary Telecommunications and Networking (IJITN)*, vol. 12, no. 4, pp. 132–140, 2020.

[10] H. Banjar, D. L. Adelson, F. Brown, and N. Chaudhri, "Intelligent techniques using molecular data analysis in leukaemia: An opportunity for personalized medicine support system.," *BioMed research international*, vol. 2017, pp. 3587309–3587309, 7 2017.

[11] E. Charney, "Behavior genetics and postgenomics.," *The Behavioral and brain sciences*, vol. 35, pp. 331–358, 10 2012.

[12] S.-S. Kim, S. McLoone, J.-H. Byeon, S. Lee, and H. Liu, "Cognitively-inspired artificial bee colony clustering for cognitive wireless sensor networks," *Cognitive Computation*, vol. 9, pp. 207–224, 1 2017.

[13] H. Hildmann, E. Kovacs, F. Saffre, and A. F. Isakovic, "Nature-inspired drone swarming for real-time aerial data-collection under dynamic operational constraints," *Drones*, vol. 3, pp. 71–, 9 2019.

[14] Álvaro Rodríguez del Nozal, D. G. Reina, L. Alvarado-Barrios, A. Tapia, and J. M. Escaño, "A mpc strategy for the optimal management of microgrids based on evolutionary optimization," *Electronics*, vol. 8, pp. 1371–, 11 2019.

[15] T. Srinivasan, V. Vijaykumar, and R. Chandrasekar, "An auction based task allocation scheme for power-aware intrusion detection in wireless ad-hoc networks," in *2006 IFIP International Conference on Wireless and Optical Communications Networks*, pp. 5–pp, IEEE, 2006.

[16] C. Zieliński, M. Figat, and R. Hexel, "Communication within multi-fsm based robotic systems," *Journal of Intelligent & Robotic Systems*, vol. 93, pp. 787–805, 6 2018.

[17] B. Abbasi, V. Majidnezhad, and S. Mirjalili, "Ade: advanced differential evolution," *Neural Computing and Applications*, vol. 36, pp. 15407–15438, 5 2024.

[18] O. Sedej, E. Mbonimpa, T. Sleight, and J. Slagley, "Artificial neural networks and gradient boosted machines used for regression to evaluate gasification processes: A review," *Journal of Energy and Power Technology*, vol. 4, pp. 1–1, 2 2022.

[19] M. Gauci, J. Chen, W. Li, T. J. Dodd, and R. Groβ, "Self-organized aggregation without computation," *The International Journal of Robotics Research*, vol. 33, pp. 1145–1161, 4 2014.

[20] N. R. Franks, J. W. Hooper, M. Gumn, T. H. Bridger, J. A. R. Marshall, R. Groß, and A. Dornhaus, "Moving targets: collective decisions and flexible choices in house-hunting ants," *Swarm Intelligence*, vol. 1, pp. 81–94, 10 2007.

[21] D. Témoa, A. Förster, null Kolyang, and S. D. Yamigno, "A reinforcement learning based intercell interference coordination in lte networks," *Future Internet*, vol. 11, pp. 19–, 1 2019.

[22] Q. Lu, G. M. Fricke, J. Ericksen, and M. E. Moses, "Swarm foraging review: Closing the gap between proof and practice," *Current Robotics Reports*, vol. 1, pp. 215–225, 8 2020.

[23] H. Kim, G. Valentini, J. R. Hanson, and S. I. Walker, "Informational architecture across non-living and living collectives.," *Theory in biosciences = Theorie in den Biowissenschaften*, vol. 140, pp. 325–341, 2 2021.

[24] K.-H. N. Bui, J. J. Jung, and D. Camacho, "Consensual negotiation-based decision making for connected appliances in smart home management systems.," *Sensors (Basel, Switzerland)*, vol. 18, pp. 2206–, 7 2018.

[25] D. Elizondo and S. G. Matthews, "Recent patents on computational intelligence," *Recent Patents on Computer Science*, vol. 1, pp. 110–117, 1 2010.

[26] D. Gutierrez-Reina, F. Barrero, J. A. Riveros, I. Gonzalez-Prieto, S. Toral, and M. J. Duran, "Interest and applicability of meta-heuristic algorithms in the electrical parameter identification of multiphase machines," *Energies*, vol. 12, pp. 314–, 1 2019.

[27] A. I. Osman, W. Abd-Elaziem, M. Nasr, M. Farghali, A. K. Rashwan, A. Hamada, Y. M. Wang, M. A. Darwish, T. A. Sebaey, A. Khatab, and A. H. Elsheikh, "Enhanced hydrogen storage efficiency with sorbents and machine learning: a review," *Environmental Chemistry Letters*, vol. 22, pp. 1703–1740, 5 2024.

[28] I. Lacalle, A. Belsa, R. Vano, and C. E. Palau, "Framework and methodology for establishing port-city policies based on real-time composite indicators and iot: A practical use-case," *Sensors (Basel, Switzerland)*, vol. 20, pp. 4131–, 7 2020.

[29] I. Tkach, A. Jevtic, S. Y. Nof, and Y. Edan, "A modified distributed bees algorithm for multi-sensor task allocation.," *Sensors (Basel, Switzerland)*, vol. 18, pp. 759–, 3 2018.

[30] G. Stein and A. J. Gonzalez, "Learning in context: enhancing machine learning with context-based reasoning," *Applied Intelligence*, vol. 41, pp. 709–724, 7 2014.

[31] A. Adamatzky, "Towards physarum robots: computing and manipulating on water surface," *Journal of Bionic Engineering*, vol. 5, pp. 348–357, 12 2008.

[32] P. Mishra, null Pooja, and S. P. Tripathi, "Optimizing constrained engineering problem nh-wdeoa: using hybrid nature-inspired algorithm," *International Journal of Information Technology*, vol. 16, pp. 1899–1907, 2 2024.

[33] T. Srinivasan, R. Chandrasekar, V. Vijaykumar, V. Mahadevan, A. Meyyappan, and M. Nivedita, "Exploring the synergism of a multiple auction-based task allocation scheme for power-aware intrusion detection in wireless ad-hoc networks," in *2006 10th IEEE Singapore International Conference on Communication Systems*, pp. 1–5, IEEE, 2006.

[34] S. Banerjee and N. Agarwal, "Analyzing collective behavior from blogs using swarm intelligence," *Knowledge and Information Systems*, vol. 33, pp. 523–547, 6 2012.

[35] R. Chandrasekar, R. Suresh, and S. Ponnambalam, "Evaluating an obstacle avoidance strategy to ant colony optimization algorithm for classification in event logs," in *2006 International Conference on Advanced Computing and Communications*, pp. 628–629, IEEE, 2006.

[36] B. Altena, T. Scambos, M. Fahnestock, and A. Kääb, "Extracting recent short-term glacier velocity evolution over southern alaska from a large collection of landsat data," pp. 1–27, 5 2018.

[37] Q. Zhou, J. Wang, X. Yu, S. Wang, and Y. Zhang, "A survey of deep learning for alzheimer's disease," *Machine Learning and Knowledge Extraction*, vol. 5, pp. 611–668, 6 2023.

[38] G. C. Chasparis and J. S. Shamma, "Distributed dynamic reinforcement of efficient outcomes in multiagent coordination and network formation," *Dynamic Games and Applications*, vol. 2, pp. 18–50, 11 2011.

[39] C. Ramachandran, S. Misra, and M. Obaidat, "On evaluating some agent-based intrusion detection schemes in mobile ad-hoc networks," in *Proceedings of the SPECTS 2007*, (San Diego, CA), pp. 594–601, July 2007.

[40] M. Dunbar, S. Belieres, N. Shukla, M. Amirghasemi, P. Perez, and N. Mishra, "A genetic column generation algorithm for sustainable spare part delivery: application to the sydney droppoint network," *Annals of Operations Research*, vol. 290, pp. 923–941, 6 2018.

[41] R. Levato, T. Jungst, R. G. Scheuring, T. Blunk, J. Groll, and J. Malda, "From shape to function: The next step in bioprinting," *Advanced materials (Deerfield Beach, Fla.)*, vol. 32, pp. 1906423–, 2 2020.

[42] E. López-Rubio, D. Elizondo, M. Grootveld, J. M. Jerez, and R. M. Luque-Baena, "Computational intelligence techniques in medicine," *Computational and mathematical methods in medicine*, vol. 2015, pp. 196976–196976, 3 2015.

[43] R. Chandrasekar and S. Misra, "Introducing an aco based paradigm for detecting wildfires using wireless sensor networks," in *2006 International Symposium on Ad Hoc and Ubiquitous Computing*, pp. 112–117, IEEE, 2006.

[44] M. T. Vrugt and R. Wittkowski, "Metareview: a survey of active matter reviews.," *The European physical journal. E, Soft matter*, vol. 48, pp. 12–, 3 2025.

[45] A. López-Incera, K. Ried, T. Müller, and H. J. Briegel, "Development of swarm behavior in artificial learning agents that adapt to different foraging environments," *PloS one*, vol. 15, pp. e0243628–, 12 2020.

[46] D. Sudholt and C. Thyssen, "A simple ant colony optimizer for stochastic shortest path problems," *Algorithmica*, vol. 64, pp. 643–672, 12 2011.

[47] M. D. Cahalan and I. Parker, "Choreography of cell motility and interaction dynamics imaged by two-photon microscopy in lymphoid organs.," *Annual review of immunology*, vol. 26, pp. 585–626, 4 2008.

[48] A. Teixeira, J. Mota, A. Garcia-Cabot, E. Garcia-Lopez, and L. de Marcos, "A new competence-based approach for personalizing moocs in a mobile collaborative and networked environment," *RIED. Revista Iberoamericana de Educación a Distancia*, vol. 19, pp. 143–160, 9 2015.

[49] J. Lötsch, S. Schiffmann, K. Schmitz, R. Brunkhorst, F. Lerch, N. Ferreirós, S. Wicker, I. Tegeder, G. Geisslinger, and A. Ultsch, "Machine-learning based lipid mediator serum concentration patterns allow identification of multiple sclerosis patients with high accuracy," *Scientific reports*, vol. 8, pp. 14884–14884, 10 2018.

[50] I. Slavkov, D. Carrillo-Zapata, N. Carranza, X. Diego, F. Jansson, J. A. Kaandorp, S. Hauert, and J. Sharpe, "Morphogenesis in robot swarms," *Science robotics*, vol. 3, 12 2018.

[51] S. Y. A. Zaidi, M. A. Shah, H. A. Khattak, C. Maple, H. T. Rauf, A. M. El-Sherbeeny, and M. A. El-Meligy, "An attribute-based access control for iot using blockchain and smart contracts," *Sustainability*, vol. 13, pp. 10556–, 9 2021.

[52] N. Howard and E. Cambria, "Intention awareness: improving upon situation awareness in human-centric environments," *Human-centric Computing and Information Sciences*, vol. 3, pp. 9–, 6 2013.

[53] M. Koopialipoor, B. R. Murlidhar, A. Hedayat, D. J. Armaghani, B. Gordan, and E. T. Mohamad, "The use of new intelligent techniques in designing retaining walls," *Engineering with Computers*, vol. 36, pp. 283–294, 1 2019.

[54] J. Thomas and P. Mantri, "Axiomatic cloud computing architectural design," *MATEC Web of Conferences*, vol. 301, pp. 24–00024, 12 2019.