B.H. : Pages:1–21

Original Research

Collective Optimization Strategies for Global Parts Management Using Multi-Agent Data Analytics

Tharindu Jayasekara[1], Ishara Nanayakkara[2] and Malithi Abeykoon[3]

[1]Ceylon Institute of Technology, Computer Science Department, Sirimangalaya Road, Kurunegala, Sri Lanka.
[2]Ruhunu College of Computing, Computer Science Department, Nelumpokuna Road, Galle, Sri Lanka.
[3]Northern Highlands University, Computer Science Department, Perumal Kovil Road, Jaffna, Sri Lanka.

Abstract
Global production and service networks operate through complex interactions among suppliers, warehouses, plants, and distribution centers that jointly manage large portfolios of parts. The resulting coordination problem is shaped by dispersed inventories, heterogeneous lead times, and demand uncertainty that varies across regions and time. Traditional centralized planning architectures encode these interactions in large optimization models that are difficult to solve and maintain as networks grow in size and complexity. At the same time, increased data availability from transactional systems and tracking technologies has created opportunities for more responsive and distributed decision processes. This paper examines collective optimization strategies for global parts management based on a multi-agent data analytics view. The study considers a setting where decision responsibilities are distributed across agents associated with locations and transportation resources. Each agent observes local states and data streams, learns predictive models for demand and lead times, and solves local optimization problems subject to shared constraints. The global problem is modeled through linear structures that capture conservation of flow, capacity limits, and service-level requirements. Coordination emerges through iterative mechanisms that exchange dual variables, price-like signals, or low-dimensional summaries of forecasts and uncertainty sets. The analysis focuses on how decomposition structures, learning architectures, and communication patterns influence cost, service, and scalability properties. The discussion emphasizes trade-offs between centralization and decentralization and outlines conditions under which collective strategies approximate centrally computed policies while accommodating modularity, heterogeneous information, and evolving data-driven models in large global parts networks.

Table 1: Global parts network overview used in the multi-agent optimization experiments

Region	# Suppliers	Avg. lead time (days)	Disruption risk index
North America	42	8.3	0.21
Europe	35	7.5	0.18
Asia-Pacific	57	13.2	0.34
Latin America	19	16.1	0.29
Middle East & Africa	14	18.7	0.37

1. Introduction

Global parts management involves the systematic coordination of material flows for many distinct part numbers across a geographically dispersed network of suppliers, consolidation hubs, regional warehouses, plants, and customer-serving locations [1]. Each part follows a trajectory that may span several

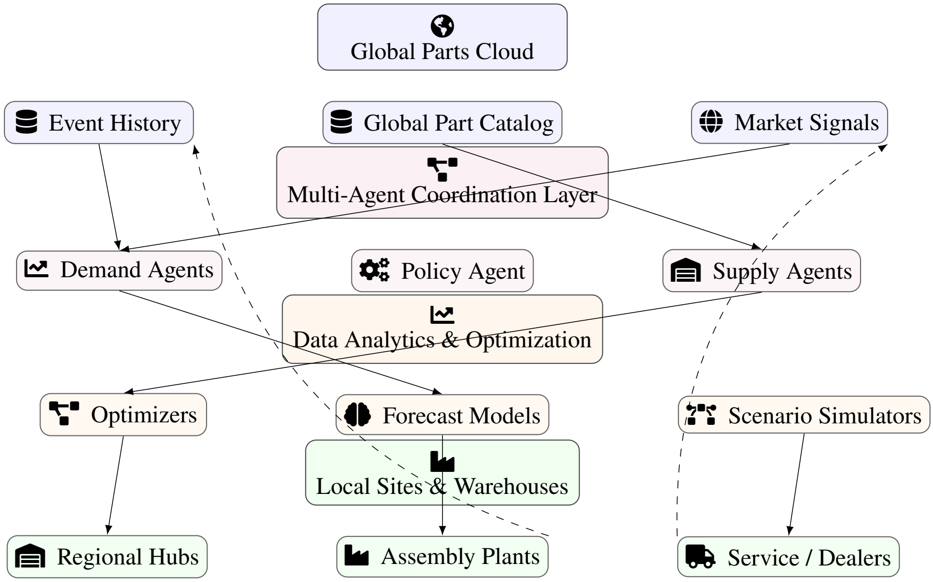


Figure 1: End-to-end architecture for collective optimization in global parts management. Cloud-based data repositories aggregate historical events, market signals, and catalog information. Multi-agent coordination orchestrates demand and supply agents, while analytics components perform forecasting, optimization, and scenario simulation. Decisions are propagated to plants, regional hubs, and dealer networks with feedback channels continuously updating the shared data layer.

Table 2: Core agent types and decision responsibilities in the collective optimization framework

Agent type	Primary objective	Decision horizon	Action frequency
Demand planner	Forecast accuracy	4–12 weeks	Weekly
Inventory allocator	Service level vs. cost	1–8 weeks	Daily
Procurement agent	Cost and risk hedging	4–24 weeks	Bi-weekly
Logistics scheduler	Lead time reliability	1–4 weeks	Daily
Capacity manager	Utilization balance	4–26 weeks	Monthly

Table 3: Data layers used for multi-agent analytics and their representative features

Layer	Source examples	Time granularity	Key features (illustrative)
Transactional	ERP, WMS	Daily	Orders, receipts, backorders, returns
Operational	TMS, MES	Hourly	Shipments, capacity, disruptions, delays
Market	POS, web	Daily	Sell-through, promotions, price indices
Risk	External	Weekly	Weather alerts, political risk, outages
Master data	PLM, MDM	Static	BOMs, sourcing rules, part criticality

echelons and transportation modes before reaching its point of use. Decision makers must determine replenishment quantities, safety stock levels, allocation rules under scarcity, and possible lateral transshipment actions, all under constraints on storage capacity, transportation resources, and contractual lead times. The decision problem is high dimensional because it couples temporal, spatial, and part-specific indices, and it is further complicated by stochastic demand, variable lead times, and occasional disruptions.

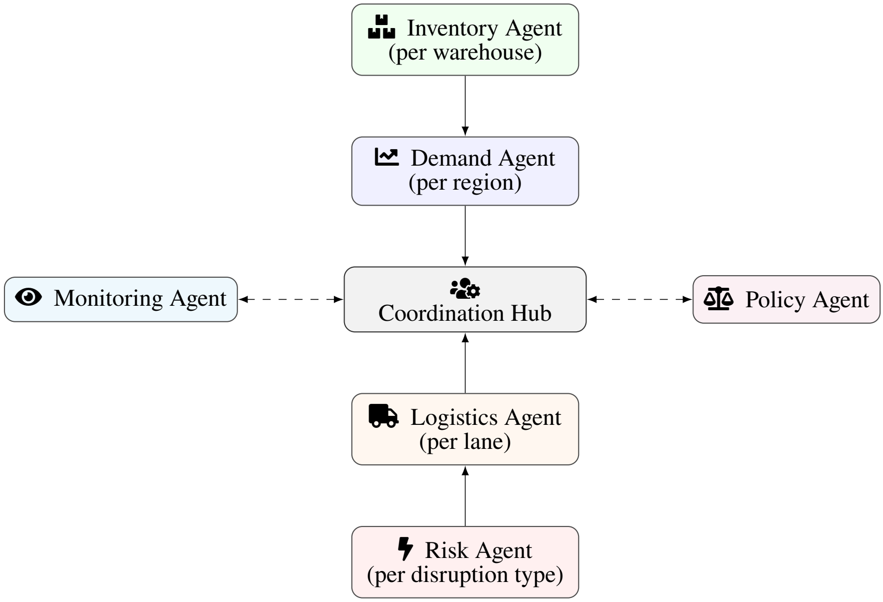


Figure 2: Multi-agent organization for global parts management. Demand, inventory, logistics, and risk agents specialize per region, warehouse, transport lane, and disruption type. A central coordination hub aggregates agent proposals and enforces joint decisions with assistance from policy and monitoring agents, enabling collective optimization while preserving specialization and local autonomy.

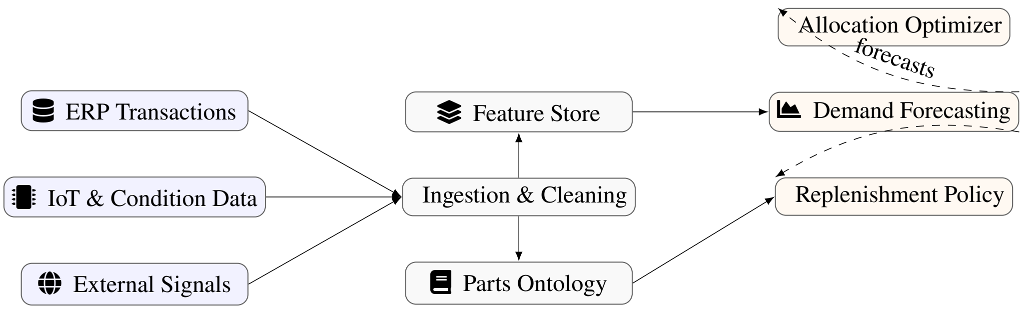
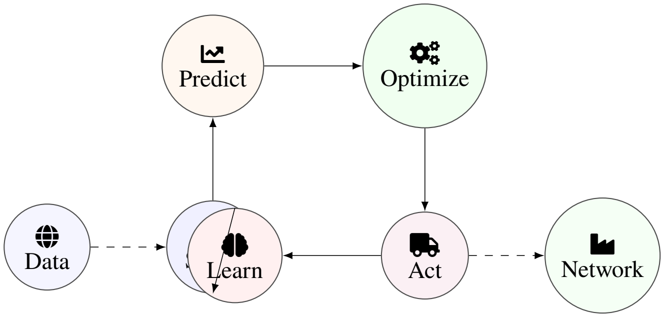


Figure 3: Data analytics pipeline underpinning multi-agent decision making. Heterogeneous ERP, IoT, and external data sources are ingested, cleaned, and fused into a feature store and shared parts ontology. Forecasting, replenishment, and allocation models consume the harmonized data and expose compact signals that are consumed by agents to drive coordinated global parts planning.

A traditional approach builds a centralized optimization model in which a single planner controls all decision variables [2]. In this view, all demand forecasts, lead-time estimates, and cost parameters are collected in a central database and used to parameterize a large linear or mixed-integer program. The model yields prescriptive decisions such as target inventory positions or planned shipments for each part and location over a finite horizon. While this approach can provide a clear global reference, it becomes increasingly difficult to maintain as network size grows and as data sources proliferate. Large models can be computationally demanding, and large-scale data integration can introduce latency and governance challenges [3].

A multi-agent perspective organizes the same problem differently. Instead of a single global decision maker, one associates an agent with each major organizational or physical entity, such as a plant, regional warehouse, or transportation group. Each agent maintains local state variables, receives local demand



Multi-agent collective optimization loop for global parts decisions

Figure 4: Closed-loop optimization cycle executed collectively by agents. Agents sense network-wide signals, use predictive models to anticipate demand and risk, jointly optimize replenishment and allocation, act on the global parts network, and learn from realized outcomes. The loop is repeated at multiple temporal and spatial scales, allowing decisions to adapt to local dynamics while remaining globally coordinated.

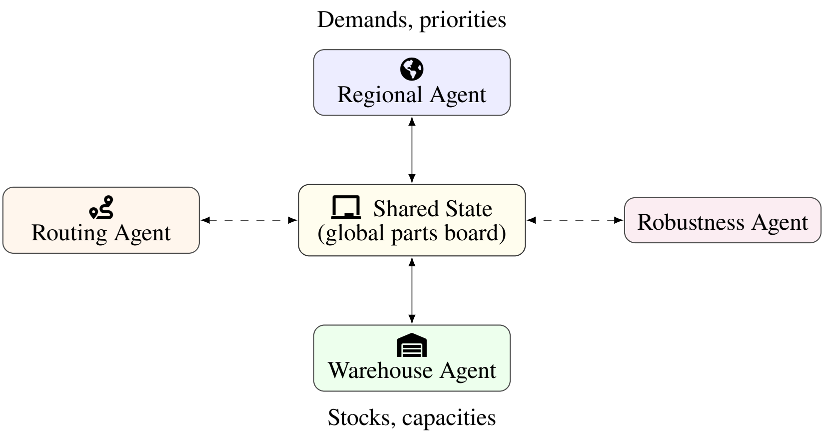


Figure 5: Collective intelligence through a shared multi-agent blackboard. Heterogeneous agents (regional, warehouse, routing, and robustness) read and write to a common global parts state. Each agent proposes partial updates based on its local view and expertise, while consistency mechanisms on the shared board ensure that the final allocation and routing plans remain globally feasible and conflict free.

signals and operational feedback, and has access to local historical data. Agents may differ in their objectives, cost structures, and information sets [4]. Collective behavior arises when agents exchange information or price-like signals and adjust their decisions in response. This view is compatible with organizational realities in global supply networks, where responsibilities are naturally distributed.

Data analytics provides the substrate on which such agents build predictive views of their environment. Transactional demand and shipment histories can be used to estimate time-varying intensities or trends, while sensor and tracking data provide information about effective lead times, delays, and capacity utilization [5]. Rather than assuming fixed parameters, one treats cost coefficients and constraint right-hand sides as outputs of learning processes that evolve over time. The resulting optimization problem is therefore coupled with estimation dynamics: improved models lead to updated optimization parameters, while decisions shape future data through their influence on inventories and flows [6].

This paper develops a linear modeling and multi-agent framework for collective optimization in global parts management. The emphasis is on linear structures because they provide a tractable and

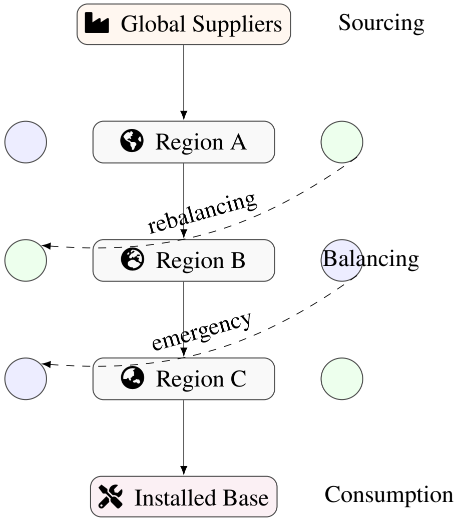


Figure 6: Global parts network under collective optimization. Multi-agent policies coordinate supplier sourcing, regional stock levels, and cross-region rebalancing. Vertical flows represent nominal movement of parts down the network, while selected dashed arcs capture agent-triggered reallocation moves that mitigate localized shortages and align global parts availability with installed base risk.

Table 4: Optimization scenarios for global parts management evaluated in the study

Scenario	Description	Main constraints	Coordination level
S1: Local myopic	Site-specific reorder rules	Storage, budget	None
S2: Regional pooling	Shared safety stock in region	Transport, service level	Regional
S3: Global coordination	Cross-region rebalancing	Capacity, customs	Global
S4: Risk-aware sourcing	Dual sourcing with hedging	Supplier limits, risk caps	Global
S5: Disruption response	Dynamic rerouting and expediting	Expedite budget	Global + local

Table 5: Comparison of collective optimization strategies on key performance indicators

Strategy	Fill rate (%)	Inventory cost (% of baseline)	Expediting cost (% of demand)	CO_2
Baseline rules	93.1	100.0	3.4	
Centralized MIP	97.8	91.2	2.1	
Independent agents	95.4	89.7	2.9	
Collective agents (no risk layer)	97.1	86.5	1.8	
Collective agents (full model)	98.3	84.1	1.3	

interpretable backbone that can incorporate a wide range of operational constraints while supporting decomposition and sensitivity analysis [7]. The modeling constructs accommodate multi-period inventory dynamics, shared capacity constraints on transportation and storage, and service-level requirements expressed through fill-rate or backorder bounds. The multi-agent decomposition partitions decision variables and constraints across agents and introduces coordination mechanisms that operate through dual variables, consensus updates, or augmented Lagrangian formulations.

A central theme is the interaction between optimization and data-driven parameter estimation. Forecasts and uncertainty characterizations influence safety stocks and allocation rules, while dual prices and constraint activity patterns can feed back into learning processes by highlighting which parts of the

Table 6: Agent communication structures considered for collective decision-making

Topology	Description	Avg. path length (hops)	Robustness to node failure
Star	Single global coordinator	1.0	Low
Ring	Each agent connected to neighbors	2.5	Medium
Hierarchy	Regional clusters with hub	1.6	Medium–High
Fully connected	All agents share state	1.0	High (communication costly)
Federated clusters	Overlapping regional groups	1.8	High

Table 7: Ablation analysis of analytics components in the multi-agent pipeline

Configuration	Forecast module	Risk module	Change in total cost vs. full model (%)
Full model	Advanced	Yes	0.0
No risk layer	Advanced	No	+4.7
Simple forecasts	Simple	Yes	+6.3
No demand learning	Naive	Yes	+9.8
Rule-based baseline	None	No	+15.6

Table 8: Regional impact of collective optimization on service and cost metrics

Region	Δ fill rate (pp)	Δ inventory cost (%)	Δ backorders (%)	Δ stockouts (%)
North America	+3.2	-11.4	-18.6	-21.9
Europe	+2.7	-9.8	-15.3	-17.1
Asia-Pacific	+5.1	-13.7	-22.4	-25.8
Latin America	+4.4	-10.1	-19.8	-23.3
Middle East & Africa	+4.9	-12.9	-21.1	-24.5

network require more accurate models. By describing these interactions within a single linear-analytic framework, the paper aims to clarify how multi-agent data analytics and collective optimization can be combined to support scalable and adaptive global parts management [8].

2. Network Representation and Linear State Models

A global parts network can be represented by a directed graph with a finite set of nodes and a finite set of arcs. Nodes represent suppliers, warehouses, plants, and demand points, while arcs represent feasible transportation or production links. Parts form a finite set, and time is indexed over discrete periods. For each part, location, and time period, one can define an inventory variable and a backlog variable [9]. For each part and arc, one can define shipment variables that capture material dispatched along that arc at a given time.

Let the set of agents be denoted by

$$K = \{1, 2, \ldots, K_{\max}\},$$

where each index identifies a plant, warehouse, supplier group, or transportation coordinator. For each agent k, consider a local state vector [10]

$$x_t^k \in \mathbb{R}^{n_k}$$

at period t, which stacks the inventories and backorders of all parts controlled by this agent. The control vector

$$u_t^k \in \mathbb{R}^{m_k}$$

represents decision variables such as order releases, shipments to downstream locations, and loads assigned to outbound transportation modes.

A linear state transition model can approximate the inventory-flow dynamics at the agent level [11]. For each k and time t, one writes

$$x_{t+1}^k = R_t^k x_t^k + S_t^k u_t^k \tag{2.1}$$
$$+ \eta_t^k, \tag{2.2}$$

where the matrix

$$R_t^k \in \mathbb{R}^{n_k \times n_k}$$

encodes deterministic carry-over of inventory and backlog, and the matrix [12]

$$S_t^k \in \mathbb{R}^{n_k \times m_k}$$

maps decisions to state changes. The disturbance vector

$$\eta_t^k \in \mathbb{R}^{n_k}$$

captures realized demand and any unmodeled effects. For many inventory models, the matrix R_t^k is close to an identity operator, while S_t^k introduces signed contributions from shipments and receipts.

At the global level, it is convenient to collect all agent states into a single vector [13]

$$x_t = \begin{bmatrix} x_t^1 \\ x_t^2 \\ \vdots \\ x_t^{K_{\max}} \end{bmatrix} \in \mathbb{R}^n,$$

and all controls into [14]

$$u_t = \begin{bmatrix} u_t^1 \\ u_t^2 \\ \vdots \\ u_t^{K_{\max}} \end{bmatrix} \in \mathbb{R}^m.$$

The global state transition can then be written in block form as [15]

$$x_{t+1} = R_t x_t + S_t u_t \tag{2.3}$$
$$+ \eta_t, \tag{2.4}$$

where the matrix

$$R_t \in \mathbb{R}^{n \times n}$$

is block diagonal with blocks R_t^k, and the matrix

$$S_t \in \mathbb{R}^{n \times m}$$

contains diagonal blocks S_t^k and possible off-diagonal blocks representing flows that directly influence multiple agents [16].

Inventory and backlog variables are subject to nonnegativity constraints at each period. Let

$$x_t = \begin{bmatrix} x_{t,\mathrm{inv}} \\ x_{t,\mathrm{bo}} \end{bmatrix},$$

where $x_{t,\mathrm{inv}}$ collects inventory components and $x_{t,\mathrm{bo}}$ collects backlog components. The constraints are

$$x_{t,\mathrm{inv}} \geq 0 \quad \text{and} \quad x_{t,\mathrm{bo}} \geq 0$$

for all t. Linear inequalities can capture storage capacity limits at locations [17]. For a given location ℓ, one can write

$$H^{\ell} x_t \leq h^{\ell},$$

where the matrix H^{ℓ} selects the inventory components that reside at ℓ and the vector h^{ℓ} encodes the capacity level.

Transportation capacities on arcs can be represented by linear constraints on shipment variables [18]. Let $v_t \in \mathbb{R}^q$ denote the vector of all shipments in period t. A capacity constraint for a lane a can be written as

$$g_a^{\top} v_t \leq c_a,$$

where the vector $g_a \in \mathbb{R}^q$ has entries equal to one for shipments that use lane a and zero otherwise, and c_a denotes the available capacity. When multiple agents use the same lane, the corresponding shipments belong to different blocks of the global control vector and appear jointly in the constraint through a suitable selection matrix.

Lead times can be incorporated by introducing in-transit state components. For an arc with deterministic lead time L, shipments dispatched at time t become available at time $t + L$ [19]. This can be modeled by a shift register, where in-transit variables

$$z_{t,\ell} \in \mathbb{R}^{q_\ell}$$

move through positions as time advances. The associated linear dynamics can be written as

$$z_{t+1,1} = v_t, \tag{2.5}$$

$$z_{t+1,j} = z_{t,j-1} \tag{2.6}$$

for $j \geq 2$, and receipts at the destination are equal to [20]

$$r_t = z_{t,L}.$$

These relations can be embedded in the global state vector and transition matrix by adding appropriate rows and columns.

A global objective for a finite horizon T often takes the form of a linear stage cost summed over time. One may write

$$J = \sum_{t=0}^{T-1} c_t^{\top} x_t + d_t^{\top} u_t,$$

where the vector $c_t \in \mathbb{R}^n$ contains holding and backlog cost coefficients and the vector $d_t \in \mathbb{R}^m$ contains transportation and ordering cost coefficients. The finite-horizon deterministic optimization problem is then to choose u_t for all t such that the linear dynamics and constraints are satisfied and the cost J is minimized [21]. This formulation serves as a basis for multi-agent decompositions and for embedding data-driven parameter updates.

3. Multi-Agent Decomposition and Coordination

The multi-agent viewpoint partitions the global decision u_t and state x_t into local components controlled and observed by different agents. For each agent k, the local decision sequence can be collected into

$$u^k = \begin{bmatrix} u_0^k \\ [22] u_1^k \\ \vdots \\ u_{T-1}^k \end{bmatrix} \in \mathbb{R}^{m_k T},$$

and the local state sequence into

$$x^k = \begin{bmatrix} x_0^k \\ [23] x_1^k \\ \vdots \\ x_T^k \end{bmatrix} \in \mathbb{R}^{n_k (T+1)}.$$

The global vectors are concatenations of these local sequences, which implies a block structure in the constraints and cost function.

Local dynamics for an agent over the horizon can be written in stacked form as

$$E^k x^k = F^k u^k + w^k, [24]$$

where

$$E^k \in \mathbb{R}^{p_k \times n_k (T+1)}$$

collects the constraints arising from the state transition equations and initial conditions,

$$F^k \in \mathbb{R}^{p_k \times m_k T}$$

maps control sequences to state changes, and

$$w^k \in \mathbb{R}^{p_k}$$

encodes known disturbances and initial states. This representation is derived by writing the equations [25]

$$x_{t+1}^k - R_t^k x_t^k - S_t^k u_t^k = \eta_t^k$$

for $t = 0, \ldots, T-1$ in a stacked form and embedding the initial condition

$$x_0^k = \bar{x}_0^k$$

as a linear equality.

Coupling between agents arises through shared constraints. For example, a transportation capacity constraint for a lane used by agents i and j can be expressed as [26]

$$G^i u^i + G^j u^j \le c,$$

where G^i and G^j select the relevant shipments in each agent's control sequence and $c \in \mathbb{R}^r$ is the capacity vector. More generally, one can write a set of global coupling constraints as

$$\sum_{k=1}^{K_{\max}} H^k u^k \le h,$$

with matrices

$$H^k \in \mathbb{R}^{r \times m_k T}$$

and a right-hand side $h \in \mathbb{R}^r$. These constraints express aggregate limits on flows, inventories, or service metrics that involve multiple agents.

The global optimization problem can be written as a structured linear program [27]. Let

$$u = \begin{bmatrix} u^1 \\ u^2 \\ \vdots \\ u^{K_{\max}} \end{bmatrix}, \quad x = \begin{bmatrix} x^1 \\ [28] x^2 \\ \vdots \\ x^{K_{\max}} \end{bmatrix}.$$

Define a global constraint matrix

$$A = \begin{bmatrix} E^1 & 0 & \cdots & 0 \\ [29] 0 & E^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & E^{K_{\max}} \\ \tilde{H}^1 & \tilde{H}^2 & \cdots & \tilde{H}^{K_{\max}} \end{bmatrix},$$

where the matrices $\tilde{H}^k$ incorporate coupling constraints written in terms of both states and controls if needed. The global right-hand side vector is

$$b = \begin{bmatrix} w^1 \\ w^2 \\ [30] \vdots \\ w^{K_{\max}} \\ h \end{bmatrix}.$$

The decision vector can be defined as

$$y = \begin{bmatrix} x \\ u \end{bmatrix},$$

and the global cost vector c stacks the stage-wise holding, backlog, and transportation costs. The linear program reads [31]

$$\min_{y} \quad c^\top y \tag{3.1}$$

$$\text{s.t.} \quad Ay \leq b, \tag{3.2}$$

$$y \geq 0. \tag{3.3}$$

Dual decomposition relaxes the coupling constraints by introducing a dual variable

$$\lambda \in \mathbb{R}^r,$$

associated with the inequalities

$$\sum_{k=1}^{K_{\max}} H^k u^k \leq h.$$

The partial Lagrangian for given λ can be written as [32]

$$\mathcal{L}(\{x^k, u^k\}, \lambda) = \sum_{k=1}^{K_{\max}} \left((c_x^k)^\top x^k + (c_u^k)^\top u^k \right) \tag{3.4}$$

$$+ \lambda^\top \left(\sum_{k=1}^{K_{\max}} H^k u^k - h \right), \tag{3.5}$$

where c_x^k and c_u^k are local cost vectors. Rearranging yields

$$\mathcal{L}(\{x^k, u^k\}, \lambda) = -\lambda^\top h \tag{3.6}$$

$$[33] \quad + \sum_{k=1}^{K_{\max}} \left((c_x^k)^\top x^k + \left(c_u^k + (H^k)^\top \lambda \right)^\top u^k \right). \tag{3.7}$$

For fixed λ, the Lagrangian separates into independent local problems for each agent k, each of which involves minimizing a linear function of x^k and u^k subject to local constraints. This gives a dual function [34]

$$\theta(\lambda) = \sum_{k=1}^{K_{\max}} \theta^k(\lambda) - \lambda^\top h,$$

where $\theta^k(\lambda)$ is the optimal value of agent k's local problem. A dual ascent algorithm updates λ using subgradients given by the violations

$$s(\lambda) = \sum_{k=1}^{K_{\max}} H^k u^{k*}(\lambda) - h,$$

where $u^{k*}(\lambda)$ is an optimal decision at the current dual iterate.

Consensus-based coordination exploits a network communication graph among agents. Suppose agents can exchange dual variables with neighbors and aim to agree on a common dual vector λ [35]. Each agent k maintains a local copy λ^k and updates it via

$$\lambda_{\ell+1}^k = \sum_{j \in \mathcal{N}_k} \alpha_{kj} \lambda_\ell^j \tag{3.8}$$

$$+ \rho s^k(\lambda_\ell^k), \tag{3.9}$$

where $\mathcal{N}_k$ is the neighbor set of agent k, the weights α_{kj} form a convex combination, ρ is a step size, and $s^k(\lambda_\ell^k) = H^k u^{k*}(\lambda_\ell^k)$ is a local contribution to the coupling constraint residual. Such schemes aim to drive both consensus on dual variables and satisfaction of shared constraints.

Augmented Lagrangian methods introduce quadratic penalties, but the core linear structure remains. One creates local copies z^k of a global variable z, imposes linear constraints [36]

$$z^k = z$$

for all k, and constructs an augmented functional

$$\mathcal{L}_\gamma(\{x^k, u^k, z^k\}, z, \{\mu^k\}) = \sum_{k=1}^{K_{\max}} \left((c_x^k)^\top x^k + (c_u^k)^\top u^k\right) \tag{3.10}$$

$$+ \sum_{k=1}^{K_{\max}} (\mu^k)^\top (z^k - z) \tag{3.11}$$

$$+ \frac{\gamma}{2} \sum_{k=1}^{K_{\max}} \|z^k - z\|_2^2, \text{ [37]} \tag{3.12}$$

with multipliers μ^k and penalty parameter $\gamma > 0$. Alternating minimization and dual ascent steps then provide a decomposition in which agents update (x^k, u^k, z^k) locally while a coordinating entity updates z and μ^k.

4. Data-Driven Parameter Learning and Uncertainty Sets

The linear models described above depend on parameters such as demand forecasts, lead-time distributions, and cost coefficients. In a data-analytic setting, these parameters are estimated from historical and streaming data [38]. For each agent k and for each part-location pair, consider a demand process

$$\{d_t^k\}_{t \geq 0},$$

where $d_t^k \in \mathbb{R}^{s_k}$ collects demands for a subset of parts. A simple predictive structure uses a feature vector

$$\phi_t^k \in \mathbb{R}^{p_k}$$

and a coefficient matrix

$$\Theta^k \in \mathbb{R}^{s_k \times p_k},$$

with conditional mean [39]

$$\hat{d}_t^k = \Theta^k \phi_t^k.$$

The coefficient matrix can be estimated by solving a local least-squares problem

$$\min_{\Theta^k} \quad \sum_{\tau=1}^{N_k} \left\| d_\tau^k - \Theta^k \phi_\tau^k \right\|_2^2, \tag{4.1}$$

where N_k is the number of historical observations. The solution

$$\Theta^{k*} = D^k (\Phi^k)^\top \left(\Phi^k (\Phi^k)^\top\right)^{-1}$$

can be expressed in terms of data matrices [40]

$$D^k = \left[d_1^k \; d_2^k \; \cdots \; d_{N_k}^k\right],$$

and

$$\Phi^k = \left[\phi_1^k \; \phi_2^k \; \cdots \; \phi_{N_k}^k\right].$$

Under streaming observations, a recursive update based on a gain matrix K_t^k can be used to track Θ^k.

Uncertainty in predictions can be summarized through residuals

$$e_\tau^k = d_\tau^k - \Theta^{k*}\phi_\tau^k,$$

and empirical covariance matrices [41]

$$\Sigma^k = \frac{1}{N_k - 1}\sum_{\tau=1}^{N_k} e_\tau^k (e_\tau^k)^\top.$$

An uncertainty set for demand over a planning window can be constructed as a Cartesian product of ellipsoids or intervals. For example, an interval set for each component i of d_t^k can be defined as

$$\mathcal{D}_t^{k,i} = \left[\hat{d}_{t,i}^k - \delta_{t,i}^k,\ \hat{d}_{t,i}^k + \delta_{t,i}^k\right],$$

with deviations $\delta_{t,i}^k$ chosen based on empirical quantiles of $e_{\tau,i}^k$. The full uncertainty set is then

$$\mathcal{D}_t^k = \mathcal{D}_t^{k,1} \times \mathcal{D}_t^{k,2} \times \cdots \times [42]\mathcal{D}_t^{k,s_k}.$$

Robust optimization can incorporate such sets by requiring that inventory constraints be satisfied for all $d_t^k \in \mathcal{D}_t^k$. Consider a scalar inventory balance for a part-location pair

$$I_{t+1}^k = I_t^k + q_t^k - d_t^k,$$

with order quantity q_t^k. A safety stock constraint can be written as

$$I_t^k \geq s_t^k [43]$$

for all t. To guarantee this for all $d_t^k \in \mathcal{D}_t^k$, one can rewrite the balance inequality as

$$I_t^k \geq I_{t-1}^k + q_{t-1}^k - \max_{d_{t-1}^k \in \mathcal{D}_{t-1}^k} d_{t-1}^k.$$

If $\mathcal{D}_{t-1}^k$ is an interval, the inner maximization reduces to a bound, and the robust counterpart remains linear.

Federated learning across agents aggregates model information without sharing raw data. Suppose agents jointly estimate a shared coefficient vector

$$\theta \in \mathbb{R}^p$$

for a global feature vector ϕ_t [44]. Each agent k maintains a local estimate θ^k and a local gradient

$$g^k(\theta^k) = \frac{1}{N_k}\sum_{\tau=1}^{N_k} \nabla_\theta \ell(d_\tau^k, \theta^k, \phi_\tau^k),$$

where ℓ is a convex loss. A parameter server or coordination agent constructs an aggregate gradient [45]

$$\bar{g} = \sum_{k=1}^{K_{\max}} \omega_k g^k(\theta^k),$$

with weights $\omega_k \geq 0$ summing to one, and performs an update

$$\theta^{\text{new}} = \theta^{\text{old}} - \eta \bar{g},$$

with step size $\eta > 0$. The updated parameter is then shared back to agents, which adjust local predictions. This mechanism preserves the linear optimization structure because it only modifies cost and constraint parameters [46].

Cost coefficients can also be learned. Let $c^k_{x,i}$ denote the holding cost per unit of a given part at agent k, and let empirical data provide pairs of inventory levels $I^k_{\tau,i}$ and cost contributions $C^k_{\tau,i}$. A linear relation

$$C^k_{\tau,i} \approx c^k_{x,i} I^k_{\tau,i}$$

can be fitted by solving a scalar least-squares problem

$$\min_{c^k_{x,i}} \sum_{\tau=1}^{N_k} \left(C^k_{\tau,i} - c^k_{x,i} I^k_{\tau,i} \right)^2 .$$

The optimal coefficient [47]

$$c^{k*}_{x,i} = \frac{\sum_{\tau=1}^{N_k} C^k_{\tau,i} I^k_{\tau,i}}{\sum_{\tau=1}^{N_k} (I^k_{\tau,i})^2}$$

then enters the cost vector c^k_x. Similar procedures can be used for shortage penalties and transportation charges when their effective values are inferred from realized performance or contractual data.

Uncertainty in lead times can be treated through random variables L_a for each lane a. Empirical distributions can be approximated by a finite set of scenarios

$$\{L^{(s)}_a\}_{s=1}^S,$$

with probabilities π_s [48]. These scenarios influence the dynamics matrices in the state-space model. A linear decision rule that is affine in the scenario index provides a tractable approximation:

$$u^k_t = u^k_{t,0} + \sum_{s=1}^{S} \alpha^k_{t,s} \xi_s,$$

where ξ_s is an indicator of scenario s and $\alpha^k_{t,s}$ are linear decision coefficients. When the ξ_s form a basis of a simplex, the resulting control remains linear in the underlying random variables [49].

5. Algorithmic Schemes and Convergence Properties

Collective optimization algorithms occur on two interacting time scales. On the fast scale, within each planning epoch, agents exchange messages and adjust decisions to approximate solutions of the linear program with fixed parameters. On the slow scale, parameters are updated by data-driven learning processes based on new observations. Algorithmic properties such as convergence and stability can be analyzed by first fixing parameters and studying the optimization dynamics, then examining the perturbations induced by gradual parameter changes [50].

Consider a dual subgradient method for the relaxed coupling constraints. The dual problem is

$$\max_{\lambda \geq 0} \quad \theta(\lambda), \tag{5.1}$$

with concave dual function θ. A generic update has the form [51]

$$\lambda_{\ell+1} = [\lambda_\ell + \alpha_\ell s(\lambda_\ell)]_+ ,$$

where $s(\lambda_\ell)$ is a subgradient of $-\theta(\lambda_\ell)$, α_ℓ is a step size, and $[\cdot]_+$ denotes projection onto the nonnegative orthant. In the present setting, the subgradient is given by

$$s(\lambda_\ell) = \sum_{k=1}^{K_{\max}} H^k u^{k*}(\lambda_\ell) - h.$$

Under standard assumptions on step sizes, such as [52]

$$\sum_{\ell=0}^{\infty} \alpha_\ell = \infty, \quad \sum_{\ell=0}^{\infty} \alpha_\ell^2 < \infty,$$

the sequence λ_ℓ converges to the set of dual optimizers, and the average of the primal iterates converges to a primal optimal solution when strong duality holds.

Consensus plus innovation schemes introduce local copies λ_ℓ^k and perform updates of the form

$$\lambda_{\ell+1}^k = \sum_{j\in\mathcal{N}_k} W_{kj}\lambda_\ell^j \tag{5.2}$$

$$+ \alpha_\ell \left(H^k u^{k*}(\lambda_\ell^k) - \frac{1}{K_{\max}} h\right), \tag{5.3}$$

with a weight matrix

$$W = [W_{kj}]$$

that is doubly stochastic and respects the communication graph. Convergence analysis typically assumes that the graph is connected and that the step sizes satisfy similar conditions as in centralized subgradient methods. One can show that the disagreement among λ_ℓ^k diminishes and that their average converges to a dual optimizer.

Augmented Lagrangian and alternating direction methods exhibit improved convergence properties under certain conditions, including linear convergence in some cases [53]. For a simple splitting between two agents A and B, the alternating direction method of multipliers (ADMM) updates can be written as

$$y_{\ell+1}^A = \arg\min_{y^A} \Big\{ c_A^\top y^A + (\mu_\ell)^\top (M^A y^A - z_\ell) \tag{5.4}$$

$$+ \frac{\gamma}{2} \|M^A y^A - z_\ell\|_2^2 \Big\}, \tag{5.5}$$

$$y_{\ell+1}^B = \arg\min_{y^B} \Big\{ [54] c_B^\top y^B + (\mu_\ell)^\top (M^B y^B - z_\ell) \tag{5.6}$$

$$+ \frac{\gamma}{2} \|M^B y^B - z_\ell\|_2^2 \Big\}, \tag{5.7}$$

$$z_{\ell+1} = \frac{1}{2}\left(M^A y_{\ell+1}^A + M^B y_{\ell+1}^B\right), \tag{5.8}$$

$$\mu_{\ell+1} = \mu_\ell + \gamma\left(M^A y_{\ell+1}^A - z_{\ell+1}\right), \tag{5.9}$$

where M^A and M^B are linear mappings that extract shared quantities. Local subproblems remain linear programs with modified cost coefficients and additional quadratic terms, and the quadratic terms can be linearized by standard techniques when only approximate solutions are required.

The interaction with learning dynamics can be modeled as a slowly varying parameter sequence [55]

$$\theta_{\tau+1} = \theta_\tau + \Delta_\tau,$$

where θ_τ collects all forecast and cost parameters at epoch τ and Δ_τ represents updates from data. If the updates are small relative to the time scale of convergence of the optimization algorithm, one can view the optimization process as tracking a moving optimum. Tracking error bounds can be derived in terms of the magnitude of Δ_τ and the contraction properties of the algorithm in a neighborhood of the optimum.

Stability of the coupled system of optimization and learning can be examined by considering Lyapunov-like functions that combine optimality gaps and prediction errors. Let V_τ denote a nonnegative function of the current dual variables, primal decisions, and parameter estimates. If one can show that

$$\mathbb{E}[V_{\tau+1} - V_\tau] \leq -\kappa \mathbb{E}[W_\tau] + \epsilon_\tau,$$

for some nonnegative function W_τ, constant $\kappa > 0$, and perturbation ϵ_τ that decays sufficiently fast, then convergence of both optimization variables and parameter estimates can be characterized in expectation.

6. Numerical Design Considerations and Scalability Analysis

From a numerical standpoint, the linear programs arising in global parts management can be extremely large, with decision vectors that include millions of variables and constraints across parts, locations, and time periods [56]. Multi-agent decomposition aims to partition this structure into smaller subproblems that can be solved in parallel. The practical effectiveness of such decompositions depends on the sparsity pattern of the constraint matrix, the strength of coupling constraints, and the communication topology.

Consider a block-structured constraint matrix

$$A = \begin{bmatrix} A^{1,1} & A^{1,2} & \dots & A^{1,K_{\max}} \\ A^{2,1} & A^{2,2} & \dots & A^{2,K_{\max}} \\ \vdots & \vdots & \ddots & \vdots \\ A^{K_{\max},1} & A^{K_{\max},2} & \dots & A^{K_{\max},K_{\max}} \end{bmatrix},$$

with diagonal blocks $A^{k,k}$ associated with local constraints and off-diagonal blocks capturing couplings. If most off-diagonal blocks are zero or of low rank, the cost of solving each local subproblem is dominated by $A^{k,k}$, and inter-agent communication is confined to a few neighboring agents. The total computational effort scales roughly with [57]

$$\sum_{k=1}^{K_{\max}} \text{cost}(A^{k,k}),$$

while coupling introduces additional overhead proportional to the number and size of nonzero off-diagonal blocks.

Preconditioning can be used to improve the convergence of iterative methods for large linear systems that arise in interior-point or first-order methods applied to the linear programs. For example, diagonal scaling matrices

$$D_r \in \mathbb{R}^{m\times m} \quad \text{and} \quad D_c \in \mathbb{R}^{n\times n}$$

can be chosen to equilibrate the rows and columns of A such that [58]

$$\|(D_r A D_c)_{i,:}\|_2 \approx \gamma$$

for all rows i and some scalar γ. This improves numerical conditioning and robustness of factorization routines across agents.

In distributed settings, communication delays and bandwidth limits motivate compressed representations of messages [59]. Dual variables and marginal costs can be quantized or sparsified before transmission. A simple sparsification strategy sends only entries of a dual vector λ that exceed a threshold in magnitude. Formally, the transmitted vector

$$\tilde{\lambda}_i = \begin{cases} \lambda_i, & [60] |\lambda_i| \geq \tau, \\ 0, & \text{otherwise}, \end{cases}$$

for each component i, with threshold $\tau > 0$. The resulting perturbation can be modeled as an additive error

$$e = \tilde{\lambda} - \lambda,$$

and convergence analysis can be extended to account for bounded errors in the dual updates [61].

Scalability can also be examined by studying asymptotic regimes in which the number of agents or parts grows. Suppose each agent manages a fixed number of parts and that coupling constraints are limited to a fixed number of neighbors. Under these conditions, the per-agent complexity of local optimization remains bounded as the network grows, and the cost of coordination grows at most linearly with the number of agents if communication is organized along sparse graphs. This suggests that multi-agent linear optimization can scale favorably with network size under structured sparsity.

From a modeling perspective, aggregation strategies such as grouping parts into families with similar demand and lead-time characteristics can reduce model size [62]. Let a mapping

$$\pi : \{1, \dots, P\} \to \{1, \dots, F\}$$

assign each part index to a family index. Aggregated demand for family f is then

$$D_t^f = \sum_{p:\pi(p)=f} d_t^p,$$

and aggregated inventory decision variables can be introduced [63]. Linear constraints can be maintained at the family level, with disaggregation rules applied heuristically or through secondary optimization when specific part-level decisions are required.

Sensitivity analysis using dual variables can support incremental updates to solutions when parameters change. If a capacity bound h_j in a constraint changes by Δh_j, the corresponding change in the optimal cost is approximately

$$\Delta J \approx \lambda_j^* \Delta h_j, [64]$$

where λ_j^* is the optimal dual variable for that constraint. Agents can use such sensitivities to prioritize which constraints should be tightened or relaxed in response to scenario analyses, without fully resolving the optimization problem.

7. Conclusion

The study of collective optimization strategies for global parts management using multi-agent data analytics combines three strands of structure: network-based linear models of inventory and flow, distributed decision architectures in which agents control local variables subject to shared constraints, and data-driven estimation of the parameters that define costs and feasible sets. Linear formulations provide a tractable representation of multi-echelon dynamics, capacity constraints, and service-level requirements, while multi-agent decompositions exploit block structure and sparsity for scalability [65].

Agents equipped with local data analytics modules learn demand forecasts, lead-time characteristics, and cost coefficients that populate the linear models. Coordinated solution methods such as dual decomposition, consensus-based schemes, and augmented Lagrangian approaches allow agents to compute decisions that approximate solutions of the global linear program while exchanging only limited information. The integration of learning and optimization introduces temporal dynamics in which both decision variables and model parameters evolve as new observations arrive.

Algorithmic considerations highlight the role of sparsity, communication constraints, and preconditioning in achieving practical scalability [66]. Sensitivity analysis based on dual variables supports incremental adaptation to changes in capacity or demand profiles without repeated full model resolutions. Uncertainty is handled through data-driven construction of sets and scenarios that are compatible with linear robust counterparts or affine decision rules. Numerical design choices such as aggregation of parts into families and federated learning of shared model components further contribute to manageable computation in large networks.

The multi-agent linear-analytic framework provides a foundation for further investigations into richer objectives, alternative coordination mechanisms, and integration with discrete decision layers such as sourcing selection or mode choice. It also creates opportunities to explore stability and performance guarantees when optimization algorithms and learning processes are tightly coupled in real-time operational environments [67].

References

[1] C. Lei and J. Ruan, "A novel swarm intelligence algorithm for finding dna motifs," *International journal of computational biology and drug design*, vol. 2, pp. 323–339, 1 2009.

[2] S. Nakrani and C. A. Tovey, "From honeybees to internet servers: biomimicry for distributed management of internet hosting centers," *Bioinspiration & biomimetics*, vol. 2, pp. S182–97, 10 2007.

[3] G. Bin, Z. Jinghua, and L. Wenchang, "A novel hybrid evolution optimization algorithm and its application," *International Journal of Database Theory and Application*, vol. 8, pp. 195–206, 10 2015.

[4] C. Ramachandran, R. Malik, X. Jin, J. Gao, K. Nahrstedt, and J. Han, "Videomule: a consensus learning approach to multi-label classification from noisy user-generated videos," in *Proceedings of the 17th ACM international conference on Multimedia*, pp. 721–724, 2009.

[5] R. Asokan, A. M. Natarajan, and A. Nivetha, "A swarm-based distance vector routing to support multiple quality of service (qos) metrics in mobile adhoc networks," *Journal of Computer Science*, vol. 3, pp. 700–707, 9 2007.

[6] S. KUKKUHALLI, "Global parts management through data and ai leveraging structured and unstructured data," *ESP INTERNATIONAL JOURNAL OF ADVANCEMENTS IN COMPUTATIONAL TECHNOLOGY : Eternal Scientific Publications*, 2023.

[7] F. Ducatelle, G. A. Di, and L. M. Gambardella, "Principles and applications of swarm intelligence for adaptive routing in telecommunications networks," *Swarm Intelligence*, vol. 4, pp. 173–198, 3 2010.

[8] T. Singh and M. K. Sandhu, "An approach in the software testing environment using artificial bee colony (abc) optimization," *International Journal of Computer Applications*, vol. 58, pp. 5–7, 11 2012.

[9] M. A. Dávila-Guzmán, W. Alfonso-Morales, and E. Caicedo-Bravo, "Arquitectura heterogénea para el procesamiento de los algoritmos de enjambres," *TecnoLógicas*, vol. 17, pp. 11–20, 1 2014.

[10] Y. Huang, A. Brocco, M. Courant, B. Hirsbrunne, and P. Kuonen, "Magate: An interoperable, decentralized and modular high-level grid scheduler," *International Journal of Distributed Systems and Technologies*, vol. 1, pp. 24–39, 7 2010.

[11] "Products," *ATZelektronik worldwide*, vol. 12, pp. 32–35, 2 2017.

[12] N. K. Sreelaja and G. A. V. Pai, "Swarm intelligence based key generation for stream cipher," *Security and Communication Networks*, vol. 4, pp. 181–194, 1 2011.

[13] N. Fortier, J. W. Sheppard, and S. Strasser, "Abductive inference in bayesian networks using distributed overlapping swarm intelligence," *Soft Computing*, vol. 19, pp. 981–1001, 5 2014.

[14] R. Chandrasekar and T. Srinivasan, "An improved probabilistic ant based clustering for distributed databases," in *Proceedings of the 20th International Joint Conference on Artificial Intelligence, IJCAI*, pp. 2701–2706, 2007.

[15] I. M. M. E. Emary and A. E. Hassanien, "Intelligent agent in telecommunication systems," *Telecommunication Systems*, vol. 46, pp. 191–193, 2 2010.

[16] T. Renk, C. Kloeck, D. Burgkhardt, F. K. Jondral, D. Grandblaise, S. Gault, and J.-C. Dunat, "Bio-inspired algorithms for dynamic resource allocation in cognitive wireless networks," *Mobile Networks and Applications*, vol. 13, pp. 431–441, 6 2008.

[17] M. Neshat, G. Sepidnam, M. Sargolzaei, and A. N. Toosi, "Artificial fish swarm algorithm: a survey of the state-of-the-art, hybridization, combinatorial and indicative applications," *Artificial Intelligence Review*, vol. 42, pp. 965–997, 5 2012.

[18] H.-S. Kim, J.-K. Park, T.-Y. Kuc, N. Y. Ko, and Y.-S. Moon, "A formation and traction control design for multiple mobile robots," *International Journal of Control, Automation and Systems*, vol. 15, pp. 1287–1301, 5 2017.

[19] P. Hu and M. Ibnkahla, "A consensus-based protocol for spectrum sharing fairness in cognitive radio ad hoc and sensor networks," *International Journal of Distributed Sensor Networks*, vol. 8, pp. 370251–, 10 2012.

[20] E. Jaswal and H. Kundra, "Earth observation and satellite imagery using spider monkey optimization (smo)," *International Journal of Advanced Research in Computer Science*, vol. 8, pp. 686–691, 8 2017.

[21] N. Valizadeh, M. Mirzaei, M. F. Allawi, H. A. Afan, N. S. Mohd, A. Hussain, and A. El-Shafie, "Artificial intelligence and geo-statistical models for stream-flow forecasting in ungauged stations: state of the art," *Natural Hazards*, vol. 86, pp. 1377–1392, 1 2017.

[22] W. Deng, H. Zhao, L. Zou, G. Li, X. Yang, and D. Wu, "A novel collaborative optimization algorithm in solving complex optimization problems," *Soft Computing*, vol. 21, pp. 4387–4398, 2 2016.

[23] Y. Hirokawa, N. Nishikawa, T. Asano, M. Terai, and T. Matsuzawa, "A study of real-time and 100 billion agents simulation using the boids model," *Artificial Life and Robotics*, vol. 21, pp. 525–530, 8 2016.

[24] R. Chandrasekar and S. Misra, "Using zonal agent distribution effectively for routing in mobile ad hoc networks," *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 3, no. 2, pp. 82–89, 2008.

[25] A. A. Rajguru, "Qos improved load balancing framework for wireless networks," *International Journal for Research in Applied Science and Engineering Technology*, vol. V, pp. 706–710, 4 2017.

[26] W. U. Kabir and M. S. Alam, "Bat algorithm with self-adaptive mutation: A comparative study on numerical optimization problems," *International Journal of Computer Applications*, vol. 100, pp. 7–13, 8 2014.

[27] J. Kennedy, "Review of engelbrecht's fundamentals of computational swarm intelligence," *Genetic Programming and Evolvable Machines*, vol. 8, pp. 107–109, 1 2007.

[28] D. R. Cañas, L. J. G. Villalba, A. L. S. Orozco, and T. hoon Kim, "Adaptive routing protocol for mobile ad hoc networks," *Computing*, vol. 96, pp. 817–827, 3 2013.

[29] K. Kumar and G. S. M. Thakur, "Advanced applications of neural networks and artificial intelligence: A review," *International Journal of Information Technology and Computer Science*, vol. 4, pp. 57–68, 6 2012.

[30] R. Gao and J. Wu, "Dynamic load balancing strategy for cloud computing with ant colony optimization," *Future Internet*, vol. 7, pp. 465–483, 11 2015.

[31] Z. He, K. Niu, T. Qiu, T. Song, W. Xu, L. Guo, and J. Lin, "A bio-inspired approach for cognitive radio networks," *Chinese Science Bulletin*, vol. 57, pp. 3723–3730, 7 2012.

[32] N. Bobde, V. Dixit, and M. T. Student, "Power allocation algorithm of ofdm-based cognitive radio network using particle swarm optimization," *International Journal of Science and Research (IJSR)*, vol. 5, pp. 1061–1064, 3 2016.

[33] K. Bidhan and A. Awasthi, "A review on parameter estimation techniques of software reliability growth models," *International Journal of Computer Applications Technology and Research*, vol. 3, pp. 267–272, 4 2014.

[34] 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.

[35] S. Dixit and R. Singhai, "A survey paper on particle swarm optimization based routing protocols in mobile ad-hoc networks," *International Journal of Computer Applications*, vol. 119, pp. 1–5, 6 2015.

[36] J. E. Hershey, S. F. Bush, and R. T. Hoctor, "Communications and control–a natural linkage for swarm," *Journal of Network and Systems Management*, vol. 14, pp. 7–13, 3 2006.

[37] H. Li, P. Gao, Q. Xiong, W. Shi, and C. Qiang, “A vascular-network-based nonuniform hierarchical fault-tolerant routing algorithm for wireless sensor networks,” *International Journal of Distributed Sensor Networks*, vol. 8, pp. 981380–, 11 2012.

[38] H. Ozcan, “Comparison of particle swarm and differential evolution optimization algorithms considering various benchmark functions,” *Journal of Polytechnic*, vol. 20, pp. 899–905, 12 2017.

[39] P. Deepalakshmi and S. Radhakrishnan, “An ant colony-based multi objective quality of service routing for mobile ad hoc networks,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2011, pp. 153–, 11 2011.

[40] P. Ghosal, A. Chakraborty, S. Banerjee, and S. Barman, “Speed optimization in unplanned traffic using bio-inspired computing and population knowledge base,” *Computer Science & Engineering: An International Journal*, vol. 2, pp. 79–97, 6 2012.

[41] A. Madureira and J. P. Lopes, “Coordinated voltage support in distribution networks with distributed generation and microgrids,” *IET Renewable Power Generation*, vol. 3, pp. 439–454, 12 2009.

[42] Y. Wang, G. Li, Y. Xu, and J. Hu, “An algorithm for mining of association rules for the information communication network alarms based on swarm intelligence,” *Mathematical Problems in Engineering*, vol. 2014, pp. 1–14, 1 2014.

[43] H. Liu, P. Zhang, B. Hu, and P. Moore, “A novel approach to task assignment in a cooperative multi-agent design system,” *Applied Intelligence*, vol. 43, pp. 162–175, 1 2015.

[44] 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.

[45] K. S. Kumar and T. S. Mahalakshmi, “Computational intelligence techniques for classification of cancer data,” *International Journal of Computer Applications*, vol. 132, pp. 14–21, 12 2015.

[46] S. C. Tan and C. P. Lim, “Integration of supervised art-based neural networks with a hybrid genetic algorithm,” *Soft Computing*, vol. 15, pp. 205–219, 12 2010.

[47] P. Agrawal, H. Kaur, and D. Bhardwaj, “Analysis and synthesis of enhanced bee colony optimization with the traditional bee colony optimization to solve travelling sales person problem,” *INTERNATIONAL JOURNAL OF COMPUTERS & TECHNOLOGY*, vol. 2, pp. 93–97, 4 2012.

[48] E. Oztemel, H. Taşkın, and C. Kubat, “Editorial,” *Journal of Intelligent Manufacturing*, vol. 23, pp. 989–990, 2 2012.

[49] P. Ćurković and B. Jerbić, “Honey-bees optimization algorithm applied to path planning problem,” *International Journal of Simulation Modelling*, vol. 6, pp. 154–164, 9 2007.

[50] Y. Kao, M.-H. Chen, and Y.-T. Huang, “A hybrid algorithm based on aco and pso for capacitated vehicle routing problems,” *Mathematical Problems in Engineering*, vol. 2012, pp. 1–17, 8 2012.

[51] L. Wang, J. Yuan, C. Wu, and X. Wang, “Practical algorithm for stochastic optimal control problem about microbial fermentation in batch culture,” *Optimization Letters*, vol. 13, pp. 527–541, 11 2017.

[52] M. Mishra, S. Agarwal, P. Mishra, and S. Singh, “Comparative analysis of various evolutionary techniques of load balancing: A review,” *International Journal of Computer Applications*, vol. 63, pp. 8–13, 2 2013.

[53] C. W. Callaghan, “The probabilistic innovation theoretical framework,” *South African Journal of Economic and Management Sciences*, vol. 20, pp. 10–, 7 2017.

[54] 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.

[55] B. Meyer, “Optimal information transfer and stochastic resonance in collective decision making,” *Swarm Intelligence*, vol. 11, pp. 131–154, 4 2017.

[56] Y. Zhao, A. Pan, Z.-R. Peng, and H. Yin, “Artificial fish swarm algorithm based optimal sensor placement,” *International Journal of Control and Automation*, vol. 8, pp. 287–300, 4 2015.

[57] W. Al-Saedi, S. Lachowicz, D. Habibi, and O. Bass, “Pso algorithm for an optimal power controller in a microgrid,” *IOP Conference Series: Earth and Environmental Science*, vol. 73, pp. 012028–, 7 2017.

[58] Z. Yuan, M. A. M. de, M. Birattari, and T. Stützle, "Continuous optimization algorithms for tuning real and integer parameters of swarm intelligence algorithms," *Swarm Intelligence*, vol. 6, pp. 49–75, 12 2011.

[59] J. Pal, "A load balancing model for job scheduling using graph theory and allocation of time," *International Journal for Research in Applied Science and Engineering Technology*, vol. V, pp. 1550–1554, 4 2017.

[60] S. Hengliang, Z. Changwei, H. Tao, and D. Yongsheng, "Research on distributed software testing platform based on cloud resource," *International Journal of Computer Science & Engineering Survey*, vol. 4, pp. 17–25, 4 2013.

[61] S. Singh and K. Kaur, "Base station localization using artificial bee colony algorithm," *International Journal of Computer Applications*, vol. 64, pp. 1–5, 2 2013.

[62] T. qi Wu, M. Yao, and J. hua Yang, "Dolphin swarm algorithm," *Frontiers of Information Technology & Electronic Engineering*, vol. 17, pp. 717–729, 10 2016.

[63] M. P. Panigrahy and P. Kaur, "A hybrid ant colony optimization algorithm for network routing and planning," *International Journal of Computer Applications*, vol. 72, pp. 15–19, 6 2013.

[64] V. Vijaykumar, R. Chandrasekar, and T. Srinivasan, "An ant odor analysis approach to the ant colony optimization algorithm for data-aggregation in wireless sensor networks," in *2006 International Conference on Wireless Communications, Networking and Mobile Computing*, pp. 1–4, IEEE, 2006.

[65] H. He, S. Kamburugamuve, G. C. Fox, and W. Zhao, "Cloud based real-time multi-robot collision avoidance for swarm robotics," *International Journal of Grid and Distributed Computing*, vol. 9, pp. 339–358, 6 2016.

[66] P. Codognet and O. Pasquet, "Ambient sound spaces," *Multimedia Tools and Applications*, vol. 58, pp. 355–370, 12 2010.

[67] G. Chen, J. Wang, and R. Li, "Parameter identification for a water quality model using two hybrid swarm intelligence algorithms," *Soft Computing*, vol. 20, pp. 2829–2839, 4 2015.