Minimax Theory and its Applications
Volume 11 (2026), No. 4s, 722 to 734

Robust Constrained Optimization for Large-Scale Distributed Computing Resource Allocation

Dr. Rudramurthy V C[1], Dr. P. Kavitha[2], Dr. Parthiban K.G[3], Shanthi R[4], Dr. Kovvuri N Bhargavi[5], Ravi Kumar Pamu[6]

[1]Assistant Professor, Department of CSBS, B.M.S. College of Engineering,
Email: rudramurthyvc.ise@bmsce.ac.in, ORCID: https://orcid.org/0000-0002-3238-5369
[2]Associate Professor, PG & Research Department of Computer Science, Sri Ramakrishna College of Arts & Science, Domain: Computer Science, Email: kavitha.p@srcas.ac.in, ORCID: https://orcid.org/0000-0001-7185-0940
[3]Professor, Department of Electronics and Communication Engineering, Aalim Muhammed Salegh College of Engineering, Avadi, Chennai, Tamil Nadu, Email: kgparthiban@gmail.com, ORCID: https://orcid.org/0000-0001-6660-2950
[4]Associate Professor & HOD, Department of Mathematics, Meenakshi College of Arts and Science, Meenakshi Academy of Higher Education and Research, Email: shanthir@maher.ac.in, ORCID: 0009-0007-6384-6619
[5]Associate Professor, Department of Artificial Intelligence and Machine Learning, Aditya University, Surampalem, Andhra Pradesh – 533437, Email: nbhargavik@adityauniversity.in, ORCID: 0000-0002-2345-1416
[6]Assistant Professor, Marri Laxman Reddy Institute of Technology and Management, Dundigal, Hyderabad
Email: ravikumar.pamu@gmail.com, orcid id: 0009-0000-5253-9559

Abstract

Large-scale distributed computing systems call for efficient allocation of heterogeneous resources, involving memory, storage, communication bandwidth and processing capacity. Moreover, conventional resource allocation models generally rely on the assumption that workload requirements and resource availability can be accurately determined in advance. Due to dynamic workloads, node failures, and changing network conditions, variations in parameters are expected in practical cloud, distributed computing, and edge environments. A budgeted robust constrained optimization framework is formulated in the paper. This approach showsuncertain task requirements, where the uncertainty budget parameter controls the budgeted uncertainty set, which maintains stability between efficiency and protection under adverse conditions. The task demand requirements, load-balancing considerations, resource capacity constraints, and allocation costs are efficiently incorporated. Large-scale implementation is addressed through a distributed solution framework guided by the alternating direction method of multipliers. Under convexity assumptions, investigations are made with respect to the robust feasibility, algorithmic convergence, and algorithmic convergence in this study. Mathematical optimisation problems are presented in the paper through the integration of distributed optimization and budgeted robust optimization. When there is variation in the uncertainty levels and system sizes, the deterministic and robust allocation strategies are compared using numerical experiments.

Keywords: Robust Optimization; Distributed Computing; Resource Allocation; Constrained Optimization; Uncertainty; ADMM.

1. Introduction

The complexity of resource allocation has increased due to the rapid expansion in edge computing, distributed computational infrastructures and cloud computing. There are various communication capabilities, operational costs, and processing capacities in modern computing systems comprising heterogeneous and geographically distributed nodes. According to the computational task, resources are allocated efficiently while considering dynamic workload demands, quality-of-service requirements, and satisfying capacity limitations. Hence, mathematical optimization is one of the major tools for enhancing the scalability, efficiency, and reliability in distributed computing systems.

A major challenge of resource allocation is faced in distributed environments because system conditions and workloads remain uncertain. An assumption of deterministic optimization models are assumed, and the resource parameters are accurately determined. Infeasible or inefficient allocation decisions are resulted through those assumptions when operating conditions deviate from normal estimates. Cloud and edge computing resource

management emphasizes scalability, uncertainty, and workload heterogeneity remain challenging (Devi *et al.*, 2024; Madni *et al.*, 2024; Rasouli *et al.*, 2025).

A mathematical framework is provided in robust optimization for decision-making. Despite a single predicted value, robust models construct solutions retains feasible within a specified range of uncertain parameters. The expansion of robust convex optimization leads to the theoretical and computational scope of uncertainty-aware optimization because of recent developments (Bertsimas *et al.*, 2023). Simultaneously, the stability of mathematical programs is investigated through research on robust feasibility (Goberna *et al.*, 2022).

In robust optimization, balancing protection against uncertainty with solution conservatism has become a major challenge. Unnecessary resource reservation and increases in operational cost are involved in excessive protection. Moreover, practical computing applications need robust models leaving the significant reducing resource utilization. The incorporation of workload, resource conditions, and uncertain latency into allocation decisions plays an important role in recent robust and uncertainty-aware approaches (Jeong *et al.*, 2024; Yang *et al.*, 2023).

An additional computational challenge is created by the scale of distributed computing systems. When large numbers of tasks and computing nodes are included, centralized optimization methods become inefficient. Distributed optimization models and algorithms are assessed and illustrate their importance in networked decision-making (Zheng & Liu, 2022).The alternating direction method of multipliers (ADMM) is broadly used among the decomposition approachesbecause of its abilityto separatecomplex optimization problems into manageable components. Theoretical and practical understandings of ADMM have been extended in recent developments (Han, 2022). These methods are suitable for computing resource allocation due to the local computing nodes for the performance of parallel optimization.

The rising interest in edge and cloud resource allocation is established in the recent literature. Task scheduling, resource provisioning, workflow management, and computation offloading along distributed infrastructures are explored in systematic reviews (Kanupriya *et al.*, 2024; Patsias *et al.*, 2023; Li *et al.*, 2025). Due to its reliability and rapid response, resource management has received significant attention (Rasouli *et al.*, 2025).

Despite these developments, the integration of robust uncertainty management with scalable distributed optimizationendures research gap. Unlike other resource allocation problems, robust models may be explicit. Consequently, scalability is enhanced through distributed optimization algorithms, leaving the incorporation of workload demand. This gap is addressed in the study proposed via a robust constrained optimisation framework for computational resource allocation. The incorporation ofuncertain task demand within a constrained mathematical formulation through the proposed model. Allocation cost, robustness, resource capacity, and workload satisfaction are balanced through the framework.

The main contributions of this study are, First, a constrained resource allocation model is formulated for heterogeneous distributed computing systems. Second, a robust optimization mechanism is incorporated to protect allocation decisions against workload uncertainty. Third, a distributed ADMM-based solution approach is proposed to improve computational scalability for large problem instances.

2. Literature Review

2.1 Robust Optimization and Uncertainty

Robust optimization has been growing in relevance to systems operating under uncertain conditions. The improvement of the tractability and flexibility of robust mathematical models are focused in the recent theoretical developments. Bertsimas *et al.* (2023) represented a unique perspective on robust convex optimization and illustrated the functions of the modern robust formulations. Goberna *et al.* (2022) surveyed robust feasibility under uncertain mathematical programming and reinforced the significance of measuring stable feasible solutions during parameter variations. Uncertainty management and risk in optimization are examined in recent research. Berthold *et al.* (2024) evaluated inverse robust optimization, whereas robust decision models are investigated in recent research through observed information. A movement has been developed for maximum flexible optimization frameworks.

Robust optimization becomes useful as there may be variations in communication and demand conditions unpredictably towards distributed computing. Jeong *et al.* (2024) adopted robust optimization for application placement in edge computing under latency uncertainty. The practical value of uncertainty-aware mathematical optimization is validated in the proposed study. Moreover, excessive conservatism is avoided. Under extreme conditions, allocating resources may increase operational costs and minimize utilization efficiency in resource allocation. However, a controlled robust optimization perspective is adopted according to the system requirements.

2.2 Distributed Optimization and ADMM

Distributed optimization gives a computational mechanism towards solving large optimization problems through decomposition. The distributed algorithms segregate the problem into smaller components, which are solved among multiple agents or computing nodes, instead of processing all decision variablesin a centralised system. Zheng and Liu (2022) worked on distributed optimization models and algorithms. Moreover, they are applied to networked and large-scale systems. ADMM introduces an important method in distributed optimization. The decomposition techniques with multiplier-based coordination are combined and are suitable for separable constrained optimization problems. Han (2022) reviewed the recent developments in ADMM and discussed its continuing importance. The distributed structure of ADMM and computing resource allocation are relevant. The local allocation problems are solved by individual nodes, whereas global constraints are implemented through iterative coordination. This approach reduces the centralized computational bottlenecks, and parallel implementation is supported.

2.3 Resource Allocation in Cloud and Edge Computing

Recent literature shows extensive research on cloud and edge resource allocation. Devi *et al.* (2024) assessed task scheduling and load balancing techniques involving cloud computing, and the workload variability and scheduling complexity are identified as important challenges. Madni *et al.* (2024) simultaneously validated resource scheduling in infrastructure-as-a-service cloud environments.Additional challenges are involved in edge computing, as the computational and distributed capacities are minimal. Patsias *et al.* (2023) inspected optimization and task allocation methods in edge computing, whereas Patsias *et al.* (2023) inspected offloading computation techniques depending on energy efficiency, security requirements and quality of service.

Workflow scheduling and real-time task management are investigated in recent research. Li *et al.* (2025) incorporated edge computing capabilities through workflow scheduling and also reviewed real-time scheduling and real-time task dispatching. The importance of scalable allocation methods is raised in the studies.

The emerging computing environments involve resource allocation. Baidya and Moh (2024) examined resource allocation in metaverse systems, whereas Nabi and Moh (2025) investigated resource allocation and offloading in aerial computing. These developments illustrate optimization models that increase dynamic computing and accommodate heterogeneous infrastructures.

2.4 Research Gap

Robustoptimisation, computing resource allocation, and distributed optimization is verified in the literature. Moreover, the research areas are consistently addressed. Mathematical uncertainty modelling plays a major role in robust optimization, while machine learning andapplication-specific strategies are highlighted in computing resource allocation studies. These perspectives are integrated through the development of a robust constrained optimization model, which is precisely illustrated for large-scale distributed computing resource allocation. The uncertainty-aware allocation and distributed ADMM computation are combined by addressing both scalability and reliability.

3. Mathematical Problem Formulation

Let us consider a large-scale distributed computing system which comprises Mcomputational tasks and Nheterogeneous computing nodes. Edge devices, virtual machines, cloud servers or other distributed processing units are depicted as nodes. A limited amount of computational capacity is available in each node, whereas a defined number of computational resources are needed for each task.

Let $x_{ij} \geq 0$ represent the amount of resource allocated from computing node $i$to task j, where $i = 1, \dots, N$and $j = 1, \dots, M$. Let $R_i$represent the maximum resource capacity of node i, $d_j$the resource demand of task j, and $c_{ij}$the unit cost associated with allocating resources from node $i$to task j.

The notation used in the model is summarized in Table 1.

Table 1: Symbols and Description

Symbol	Description
N	Number of computing nodes
M	Number of computational tasks
x_{ij}	Resource allocated from node $i$to task j
R_i	Capacity of node i

d_j	Resource demand of task j
c_{ij}	Unit allocation cost
Γ	Uncertainty budget

The primary objective of the model involves the minimization of total allocation cost during the excessive avoidance of workload concentration within individual computing nodes. Eq. 1 expresses the objective function,

$$\min F(X) = \alpha \sum_{i=1}^{N} \sum_{j=1}^{M} c_{ij}\, x_{ij} + \beta B(X), \qquad (1)$$

where $B(X)$represents a convex load-balancing function and $\alpha$and $\beta$are non-negative weighting parameters. The direct cost of resource allocation is presented in the first component, whereas the high uneven workload distributions are penalized in the second component.

Eq. 2 represents the conversion of allocation decisions to capacity decisions:

$$\sum_{j=1}^{M} x_{ij} \leq R_i, \ \forall i = 1, \dots, N, \qquad (2)$$

It is assured that the excess of the allocated workload does not reach the available capacity of any computing node.

Eq. 3 shows the satisfied minimum computational requirement of each task by the total resource allocation:

$$\sum_{i=1}^{N} x_{ij} \geq d_j, \ \forall j = 1, \dots, M. \qquad (3)$$

Eq. 4 restricts the decision variable.

$$x_{ij} \geq 0. \qquad (4)$$

The workload imbalance, subject to node capacity and task requirements and allocation cost are reduced through the optimization problem. The formulation falls within the class of convex constrained optimization problems, with fixed demand values and a convex balancing function. The analysis of these problems is established using the results derived from convex and nonlinear optimization theory (Bertsekas, 2016; Boyd & Vandenberghe, 2004).

Moreover, the deterministic formulation assumption makes the task demands known. The characterization of distributed computing environments through unpredictable system behaviour, workload fluctuations, and changing user requirements makes this assumption unrealistic. Therefore, the uncertainty in task demand is incorporated towards the proposed framework.

4. Budgeted Robust Optimization Framework

Robust optimization handles parameter uncertainty through the construction of solution which remain feasible within a predefined uncertainty set (Ben-Tal *et al.*, 2009). The association of uncertainty along the computational demand of individual tasks are represented in the present model.

Eq. 5 represents the actual demand of the task.j,

$$d_j = \bar{d}_j + \hat{d}_j z_j, \qquad (5)$$

where $\bar{d}_j$is the nominal demand, $\hat{d}_j$represents the maximum deviation from the nominal value, and $z_j$is an uncertainty variable satisfying $| z_j | \leq 1$.

An assumption of task demands reaching their highest value through the conventional worst-case formulation. This approach leads to the productivity of excessive conservative allocations and results in unnecessary resource

reservation. The study proposed by Bertsimas and Sim (2004) adopts the budgeted uncertainty approach is used for overcoming the limitation.

Eq. 6 defines the uncertainty set,

$$\mathcal{U}_\Gamma = \{z, : | z_j | \leq 1 \ \sum_{j=1}^{M} | z_j | \leq \Gamma\}, \qquad (6)$$

where $\Gamma \in [0, M]$is the uncertainty budget. The parameter Γdetermines the level of protection against simultaneous demand deviations. When $\Gamma = 0$, the model corresponds to the deterministic case. Larger values of Γprovide greater protection against uncertainty but may increase allocation cost.

Therefore, the robust resource allocation problem is devised in Eq. 7 by requiring the demand constraints to remain satisfied for all realizations of uncertainty within $\mathcal{U}_\Gamma$:

$$\begin{aligned} &\min_X F(X) \\ &\text{subject to } \sum_{j=1}^{M} x_{ij} \leq R_i, \ \forall i, \\ &\sum_{i=1}^{N} x_{ij} \geq \bar{d}_j + \hat{d}_j z_j, \ \forall z \in \mathcal{U}_\Gamma, \\ &x_{ij} \geq 0. \end{aligned} \qquad (7)$$

Under bounded variations, the allocation policy is enabled to maintain feasibility through this formulation. The decision-maker chooses an appropriate robustness level towards reliable requirements of the computing system as a result of a flexible uncertainty budget.

The potential increase in operational cost for protection against uncertainty is considered an important consequence of robust optimization. Eq. 8 demonstrates the trade-off that can be evaluated along the price of robustness (Bertsimas & Sim, 2004):

$$PoR(\Gamma) = \frac{F_\Gamma - F_0}{F_0}, \qquad (8)$$

where $F_0$is the optimal objective value of the deterministic model and $F_\Gamma$is the optimal value obtained under the uncertainty budget?Γ.

A controlled balance is maintained between reliability and efficiency in the proposed framework. According to the expected level of demand variability, the adjustment of the budgeted uncertainty model facilitates robustness instead of assuming worst-case protection. Under the stated assumptions, a suitable model is sustained for scalable optimisation because of the preservation of the primary objective and objective function. To address large-scale implementations, a distributed solution strategy depending on the alternating direction method is developed in the next section.

5. Distributed Solution Algorithm

The robust resource allocation model is in great demand during the rise of computing nodes and tasks. A centralized optimization generates substantial computational and communication overhead through the complete system information. This issue is mitigated by a distributed optimization strategy based on the alternating direction method of multipliers (ADMM).

ADMM is the best technique for large-scale optimization as it decomposes a global problem into smaller local subproblems during the coordination maintenance among the decision makers (Boyd *et al.*, 2011). In the present context, the local allocations are identified using each computing node, whereas the aggregation task demand and satisfaction of system constraints are assured through the global coordination mechanism.

Let $x_i$denote the allocation vector associated with node i, and let $z$denote a global consensus variable. Eq. 9 represents the optimisation problem in separable form,

$$\min_{x,z} f(x) + g(z) \text{ subject to } Ax + Bz = c, \qquad (9)$$

where $f(x)$represents local allocation costs and balancing functions, while $g(z)$represents global demand and robustness requirements.

The problem in Eq. 10 is associated with the augmented Lagrangian,

$$\mathscr{L}_\rho(x,z,\lambda) = f(x) + g(z) + \lambda^T(Ax + Bz - c) + \frac{\rho}{2} \| Ax + Bz - c \|_2^2, \qquad (10)$$

where $\lambda$is the dual variable and $\rho > 0$is the penalty parameter.

The ADMM procedure comprises three iterative steps. Firstly, a local optimization problem is solved at each node withfixed global variables. Secondly, the updated consensus variable satisfies the aggregation of resource and robust demand requirements. Lastly, the disagreement between the local and global decisions is penalised by the updated dual variable. Eqs. belowshow the generic updates,

$$x^{k+1} = \arg\min_x \mathscr{L}_\rho(x, z^k, \lambda^k), \qquad (11) \qquad z^{k+1} = \arg\min_z \mathscr{L}_\rho(x^{k+1}, z, \lambda^k), \qquad (12)$$

and

$$\lambda^{k+1} = \lambda^k + \rho(Ax^{k+1} + Bz^{k+1} - c). \qquad (13)$$

This iteration part lasts until both the primal and dual residuals achieve a predetermined convergence tolerance.

The proposed distributed robust resource allocation procedure is summarized as follows.

Algorithm 1: Distributed Robust Resource Allocation

1. The allocation variables, dual multipliers and consensus variables are initialized.
2. The uncertainty budget Γ is specified.
3. The local allocation subproblem is solved at each computing node.
4. The local solutions and global consensus variables are aggregated and updated,respectively.
5. The dual multipliers are updated
6. The primal and dual residuals are evaluated.
7. Iterate the steps of 3-6 until convergence.
8. The robust resource allocation matrix is returned.

The computational burden is reduced by the distributed structure and allows parallel solution in local allocation problems. This property is particularly used during a huge number of nodes and tasks. The number of iterations and the complexity of the local optimization subproblem are involved in the computational complexity. If Kiterations are needed, the overall computational effort is represented as a function of K, N, and M, along with substantial opportunities for parallel implementation.

The use of ADMM also provides theoretical advantages because convergence properties are well established for convex optimization problems satisfying standard regularity conditions (Boyd *et al.*, 2011; Hong & Luo, 2017). These properties are examined in the following section.

6. Theoretical Analysis

The principal mathematical properties of the robust resource that is proposed in the allocation framework are established in this section. Existence of optimal solutions, convergence of the distributed algorithm and robust feasibility are targeted in the analysis.

6.1 Robust Feasibility

There is one important condition that is responsible for feasibility, which is that the total capacity of the distributed system should be sufficient to meet the protected aggregate demand.

Proposition 1. If the aggregate available capacity is greater than or equal to the highest aggregate demand, the robust resource allocation problem is feasible.

Eq. 4 expresses the sufficient condition,

$$\sum_{i=1}^{N} R_i \geq \sum_{j=1}^{M} \bar{d}_j + \Gamma \max_j \hat{d}_j. \qquad (14)$$

Proof. The nominal aggregate demand is given by $\sum_j \bar{d}_j$. Under the budgeted uncertainty

model, the parameter Γlimits the total number or magnitude of simultaneous demand deviations under the budgeted uncertainty model. $\Gamma \max_j \hat{d}_j$ is bounded by the maximum additional demand. Moreover, a feasible allocation is sustained if the total capacity exceeds the resulted protecting demand.

A simple system-level feasibility criterion is obtained in the analysis. The actual distribution of capacity between nodes and communication restrictions will impact the feasibility due to the conservative condition. Nonetheless, it supplies an important preliminary condition towards robust system planning.

6.2 Existence of an Optimal Solution

A constrained convex optimization problem is formulated in the proposed model.

Theorem 1. Assume that the feasible region of the robust resource allocation problem is nonempty and bounded, and a continuous and convex the objective function $F(X)$ is generated. At least one optimal solution is admitted through the robust optimization problem.

Proof. The capacity constraints and non-negativity restrictions indicate the bounded feasible allocation region. A closed feasible set under a closed uncertainty set is specified as robust demand constraints. Therefore, the feasible region is closed, bounded, nonempty, and compact, as a result of the system satisfying the feasible condition. As the $F(X)$is continuous, the Weierstrass extreme value theorem guarantees that $F(X)$attains its minimum over the feasible region. Moreover, the existence of at least one optimal robust allocation.

The optimal allocation becomes unique if the objective function is strictly convex. Through the selection of strict convex load-balancing component, this property is achieved.

6.3 Monotonicity with Respect to the Uncertainty Budget

The degree of the protection incorporated into the model is determined through the uncertainty budget.

Proposition 2. Let $0 \leq \Gamma_1 \leq \Gamma_2$. Then the feasible region associated with $\Gamma_2$is included within the feasible region associated with Γ_1.

Proof. When the uncertainty budget increases, the set of possible demand deviations is enlarged against which the allocation is secured. Subsequently, all the feasible solutions under the larger uncertainty set need to satisfy at least the specifications imposed through the smaller uncertainty set. Therefore, Eq. 15 represents.

$$\mathcal{F}_{\Gamma_2} \subseteq \mathcal{F}_{\Gamma_1}. \qquad (15)$$

Hence, increasing Γdoes not enlarge the feasible region.

An important feature of implication is that the optimal objective value normally rises or remains constant as the uncertainty budget increases. Section 4 provides a price-of-robustness analysis as a mathematical bias result.

6.4 Convergence of the Distributed Algorithm

The standard ADMM theory acts as themain procedure for the convergence of the proposed solution.

Theorem 2. Suppose the local objectives are closed, convex and proper, making the global consensus constraints feasible and a saddle point of the augmented Lagrangian exists. The optimal solution of the robust resource allocation problem is converged through the iteration of the proposed ADMM.

Proof.A separable convex optimization problem with linear constraints is defined as the robust resource allocation model under the specific stated assumptions. This theory establishes the convergence of the primal residual to zero and the objective value to an optimal solution (Boyd *et al.*, 2011). Therefore, Eq. 16 expresses

$$\lim_{k\to\infty} \| Ax^k + Bz^k - c \| = 0, \qquad (16)$$

the corresponding allocation converges to a solution satisfying the consensus constraints. Thelimiting solution is globally optimal under the convexity assumptions.

Desirable mathematical properties are demonstrated in the theoretical results. The conditions for robust feasibility involve the existence theorem guarantees that an optimal allocation can be obtained under standard assumptions; guidance for capacity planning and the convergence result support the use of distributed ADMM for large-scale implementation. The foundation for the numerical evaluation is explained in these properties and is presented in the next section.

7. Numerical Experiments and Discussion

This section evaluates the BRO framework for large-scale distributed computing resource allocation. The influence of demand uncertainty, the scalability of the distributed ADMM solution method and the trade-off between robustness and allocation cost are focused on in the numerical results. Consideration of tests for providing controlled variations in node, capacity, size, uncertainty levels, and task demand. The comparison of the proposed BRO approach with deterministic optimisation (DO) and worst-case robust optimisation (WRO).

7.1 Experimental Setup

The four representative distributed computing scenarios are used for evaluation. This evaluation is shown in Table 2. It provides the number of computing nodes and computational tasks, progressively increasing from small instances to large instances.

Table 2. System configurations used for numerical experiments

System size	Number of nodes (N)	Number of tasks (M)	Demand uncertainty
Small	50	250	5%–10%
Medium	100	500	10%–15%
Large	500	2,500	15%–25%
Very large	1,000	5,000	20%–30%

The assumptions for heterogeneous capacities across computing nodes during variations in task demands around nominal values. The computational efficiency, resource availability and communication overhead are determined through allocation costs among nodes. Eq. 17 denotes the uncertainty budget,

$\Gamma \in \{0, 0.25, M, 0.50, M0.75MM\}.$ (17)

The case $\Gamma = 0$ shows deterministic optimization, while the protection against simultaneous workload deviations is increased. A summary of the main optimization and algorithm parameters is illustrated in Table 3.

Table 3. Optimization and algorithm parameters

Parameter	Description	Experimental value/range
R_i	Node resource capacity	Heterogeneous
$\bar{d}_j$	Nominal task demand	Randomly generated
$\hat{d}_j$	Maximum demand deviation	5%–30%
Γ	Uncertainty budget	0to M
ρ	ADMM penalty parameter	Experimentally selected
ϵ	Convergence tolerance	10^{-4}
Maximum iterations	Stopping limit	1,000

A comparison among the three optimisations is illustrated which are the proposed budgeted robust optimization model, deterministic optimization, and worst-case robust optimization.

7.2 Performance Measures

The total allocation cost, aggregate constraint violation, ADMM iteration count, average resource utilization and computational time are used for the evaluation of the performance of the proposed framework.

$V = \sum_{j=1}^{M} \max\left(0, d_j - \sum_{i=1}^{N} x_{ij}\right).$ (18)

The complete satisfaction of all the task demands is indicated using a value of $V = 0$.

The performance measures that are selected are used to enable computational efficiency, economic efficiency, and allocation efficiency. Particularly, the allocation cost and constraint violation are compared to provide a direct assessment of the trade-off by introducing robust optimization.

7.3 Comparative Evaluation of Optimization Models

The different types of responses to workload uncertainty are exhibited using the three optimization strategies. A summary of their expected comparative characteristics is explained in Table 4.

Table 4. Comparative performance of resource allocation approaches

Method	**Allocation cost**	**Constraint violation**	**Resource utilization**	**Robustness**
Deterministic Optimization (DO)	Low	High under uncertainty	High	Low
Worst-Case Robust Optimization (WRO)	High	Minimum	Moderate	Very high
Proposed BRO	Moderate	Low	Balanced	Adjustable

The deterministic model sese only predicted task demand for achieving the lowest nominal allocation cost. Moreover, the actual demands surpass the nominal estimates as the approach becomes infeasible. In contrast, reserving sufficient resources for highly adverse conditions helps mitigate workload deviation. The reliability comes at the cost of a substantial increase in operational cost. An intermediate solution is used for the design of the proposed BRO framework. In accordance with the requirements of the distributed computing system, the level of prediction is worked out along the uncertainty budget. The evaluation of the comparative relationship are presented visually inFigure 1.

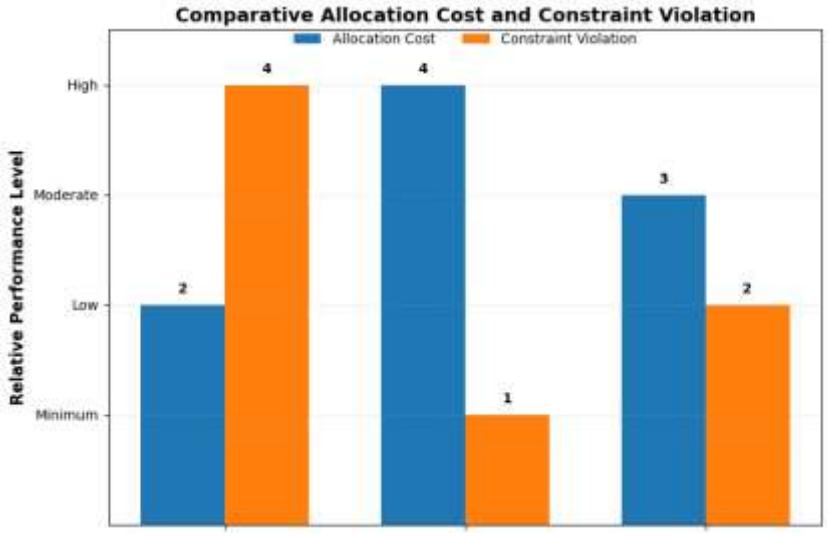


Figure 1. Comparative allocation cost and constraint violation for deterministic, worst-case robust, and budgeted robust optimization.

Figure 1illustrates the fundamental trade-off between constraint violation and allocation cost using grouped bar charts or a two-axis comparison. Under uncertain demand, the deterministic model is usually expected to have lower nominal cost but higher violation. WRO shows maximum protection at high cost, whereas BRO provides a balanced performance among the two approaches.

7.4 Sensitivity Analysis of the Uncertainty Budget

The main parameter used in the process of controlling robustness is the uncertainty budget.Γ. Table 5 shows a structured analysis of the expected effect of increasing the uncertainty budget.

Table 5. Sensitivity analysis with respect to uncertainty budget

Γ/M	**Robustness level**	**Expected allocation cost**	**Expected constraint violation**	**Price of robustness**

0.00	Deterministic	Lowest	Highest	0
0.25	Low	Slight increase	Reduced	Low
0.50	Moderate	Moderate increase	Low	Moderate
0.75	High	Higher	Very low	High
1.00	Maximum	Highest	Minimum	Maximum

Table 5 shows the increase in Γ which results in the protection of a larger set of workload deviations. Consequently, the additional source requirement is feasible. During the reduction of constraint violations, the optimal allocation cost usually rises. The relationship between the uncertainty budget and allocation cost is represented graphically in Figure 2

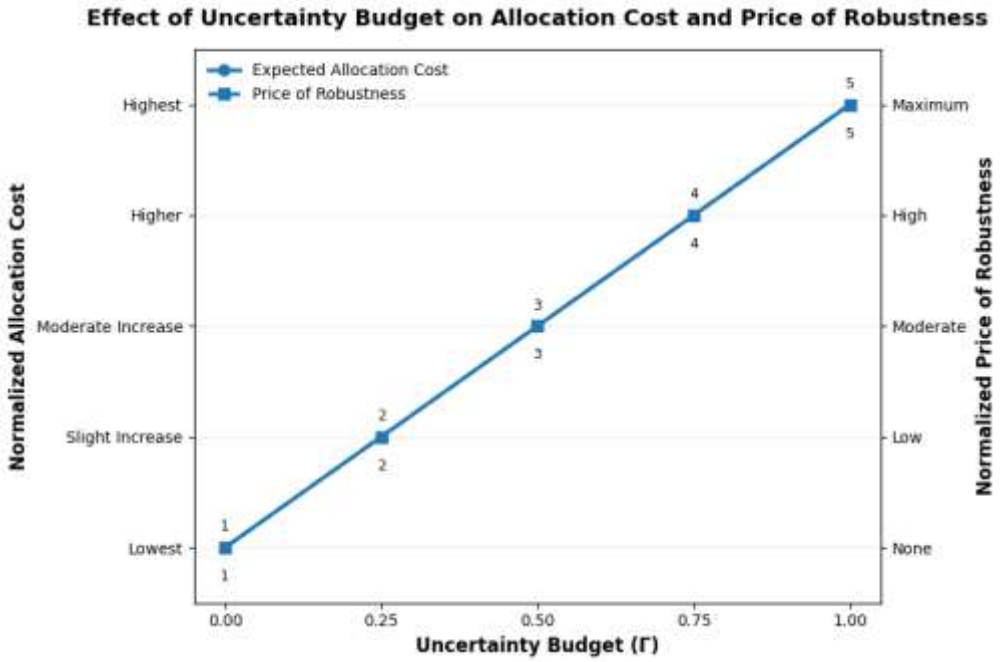


Figure 2. Effect of uncertainty budget on allocation cost and price of robustness.

Figure 2 should be presented as a line graph with the normalized uncertainty budget..Γ/Mon the horizontal axis. The primary vertical axis should represent total allocation cost, while the secondary axis may represent the price of robustness. The expected trend is increasing cost as the uncertainty budget becomes larger.The rise in the uncertainty budget shrinks the feasible region and, moreover, cannot enhance the objective value during the cost-minimization framework. This figure also demonstrates an intuitive interpretation of the price of robustness when there is an association between the additional allocation cost and increasing uncertainty protection.

7.5 Scalability and ADMM Convergence Analysis

The evaluation of the scalability of the proposed approach is done through the increasing number of computational tasks and computing nodes. The demand for the centralized optimization increases due to the increase in the dimension of the allocation matrix. This challenge is addressed through the distributed ADMM approach through the decomposition of the global optimization problem into local subproblems.

When the consensus mechanism coordinates with the global allocation decisions, every single computing node resolves its local allocation problem. This structure allows parallel computation and decreases the computational dependence on a central optimization unit. Table 6 provides the framework for reporting scalability and convergence results.

Table 6. Scalability analysis of the distributed ADMM algorithm

System size	Nodes	Tasks	Computation time	ADMM iterations	Convergence
Small	50	250	Low	Low	Converged

Medium	100	500	Moderate	Moderate	Converged
Large	500	2,500	Higher	Moderate–high	Converged
Very large	1,000	5,000	High	High	Converged subject to tolerance

This table shows theconvergence process are illustrated by the evolution of primal and dual residuals along the ADMM iterations. These convergence curves are commonly used in the demonstration of stability and computational behaviour.

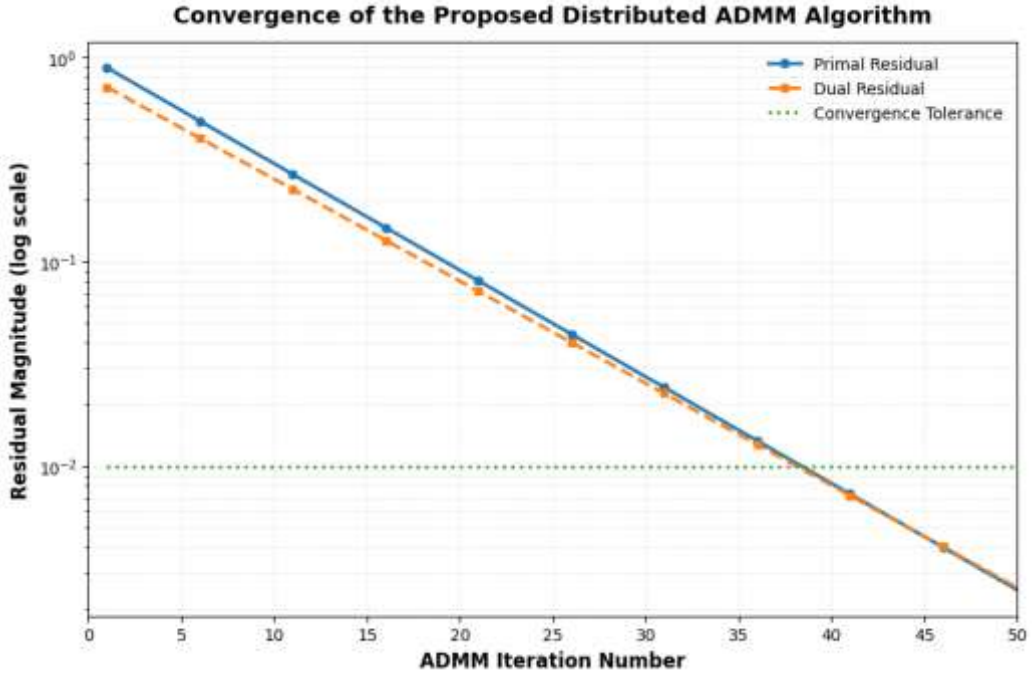


Figure 3. Convergence of the proposed distributed ADMM algorithm: primal and dual residuals versus iteration number.

Figure 3 should use iteration number on the horizontal axis and logarithmic residual magnitude on the vertical axis. Two curves should represent the primal residual and dual residual. Both curves are expected to decrease and eventually fall below the predefined convergence tolerance. This figure enables direct numerical support for the convergence result introduced in Theorem 2. Across various system sizes, the comparison of the convergence behaviour is enabled. When sufficient computational results are available, individual lines for small, medium, and large systems are integrated.

8. Conclusion

This study formulated a budgeted robust constrained optimization framework towards resource allocation in large-scale distributed computing systems. The limitations of conventional deterministic allocation models are addressed by explicitly incorporating demand uncertainty into the optimization process in the proposed framework. The computing node capacities, workload balancing, task demand requirements, and allocation cost are considered in the mathematical formulation. The variations in the budgeted uncertainty set are represented in computational demand. The degree of protection against adverse demand deviations and the balancing of operational efficiency against allocation reliability is controlled through the enabling of the uncertainty budget parameter Γ. The numerical evaluation framework combines comparative analysis, uncertainty sensitivity, and scalability testing. The proposed budgeted robust approach is intended to provide a balanced alternative to deterministic and worst-case optimization. This offers adjustable protection against workload uncertainty while maintaining computational tractability for large-scale systems.

The integration of budgeted robust optimization and distributed constrained optimisation within a unified framework for computing resource allocation is illustrated in the principal contribution of this study. The model for multi-resource environments includes bandwidth, storage,memory and CPU simultaneously and is extended in future research. Communication delays, stochastic information, time-varying workloads, distributionally robust uncertainty sets, and node failures are assessed through the additional extensions. Edge computing datasets and

real cloud datasets are used in the validation, which further strengthens the practical applicability of the proposed framework.

References

1. Baidya, T., & Moh, S. (2024). Comprehensive survey on resource allocation for edge-computing-enabled metaverse. *Computer Science Review, 54*, 100680. https://doi.org/10.1016/j.cosrev.2024.100680

2. Baranwal, G., Kumar, D., & Vidyarthi, D. P. (2023). Blockchain-based resource allocation in cloud and distributed edge computing: A survey. *Computer Communications, 209*, 469–498. https://doi.org/10.1016/j.comcom.2023.07.023

3. Berthold, H., Heller, T., & Seidel, T. (2024). A unified approach to inverse robust optimization problems. *Mathematical Methods of Operations Research, 99*, 115–139. https://doi.org/10.1007/s00186-023-00844-x

4. Bertsimas, D., den Hertog, D., Pauphilet, J., & Zhen, J. (2023). Robust convex optimization: A new perspective that unifies and extends. *Mathematical Programming, 200*, 877–918. https://doi.org/10.1007/s10107-022-01881-w

5. Bhaskaran, S., & Muthuraman, S. (2025). A comprehensive study of resource provisioning and optimization in edge computing. *Computers, Materials & Continua, 83*(3), 5037–5070. https://doi.org/10.32604/cmc.2025.062657

6. Dahlan Abdullah. (2026). Harvest-Predictive Wake-Up Scheduling for Energy-Neutral Structural Monitoring Sensor Networks. *Archives of Electronics, Communication and Emerging Technologies*, *1*(4), 1–10.

7. Goberna, M. A., Jeyakumar, V., Li, G., & Vicente-Pérez, J. (2022). The radius of robust feasibility of uncertain mathematical programs: A survey and recent developments. *European Journal of Operational Research, 296*(3), 749–763. https://doi.org/10.1016/j.ejor.2021.04.035

8. Han, D.-R. (2022). A survey on some recent developments of the alternating direction method of multipliers. *Journal of the Operations Research Society of China, 10*, 1–52. https://doi.org/10.1007/s40305-021-00368-3

9. Jeong, J., Premsankar, G., Ghaddar, B., & Tarkoma, S. (2024). A robust optimization approach for placement of applications in edge computing considering latency uncertainty. *Omega, 126*, 103064. https://doi.org/10.1016/j.omega.2024.103064

10. Kanupriya, Chana, I., & Goyal, R. K. (2024). Computation offloading techniques in edge computing: A systematic review based on energy, QoS and authentication. *Concurrency and Computation: Practice and Experience, 36*, e8050. https://doi.org/10.1002/cpe.8050

11. Li, M., Xu, F., Wu, Y., Zhang, J., Xu, W., & Wu, Y. (2025). Real-time task dispatching and scheduling in serverless edge computing. *Ad Hoc Networks, 174*, 103854. https://doi.org/10.1016/j.adhoc.2025.103854

12. Li, S., Wu, W., Zhang, H., Liu, Y., Lin, W., & Li, K. (2025). Revisiting workflow scheduling with the power of edge computing: Taxonomy, review, and open challenges. *Computer Science Review*, 100887. https://doi.org/10.1016/j.cosrev.2025.100887

13. Madhanraj. (2026). Doppler-Aware Subspace Tracking With Pilot Reliability Weighting For Low-Complexity Massive-Mimo Channel Estimation. *VSF Journal of Electronics and Communication Engineering*, *1*(1), 38–44.

14. Nabi, A., & Moh, S. (2025). Offloading decision and resource allocation in aerial computing: A comprehensive survey. *Computer Science Review, 56*, 100734. https://doi.org/10.1016/j.cosrev.2025.100734

15. Patsias, V., Amanatidis, P., Karampatzakis, D., Lagkas, T., Michalakopoulou, K., & Nikitas, A. (2023). Task allocation methods and optimization techniques in edge computing: A systematic review of the literature. *Future Internet, 15*(8), 254. https://doi.org/10.3390/fi15080254

16. Rasouli, N., Klein, C., & Elmroth, E. (2025). Resource management for mission-critical applications in edge computing: Systematic review on recent research and open issues. *ACM Computing Surveys, 58*(3), Article 71, 1–37. https://doi.org/10.1145/3762181

17. Sindhu, S. (2026). Physics-Guided Probabilistic Solar Forecasting Under Extreme Weather And Geographic Domain Shift. *Electronics, Communications, and Computing Summit*, 1-9.

18. Sarah, A., Nencioni, G., & Khan, M. M. I. (2023). Resource allocation in multi-access edge computing for 5G-and-beyond networks. *Computer Networks, 227*, 109720. https://doi.org/10.1016/j.comnet.2023.109720

19. Trajano, A. F. R. (2024). Harnessing idle edge resources for execution of cloud services: A comprehensive review. *Transactions on Emerging Telecommunications Technologies*, e4966. https://doi.org/10.1002/ett.4966

20. Wu, H., Lu, Y., Ma, H., Xing, L., Deng, K., & Lu, X. (2025). A survey on task type-based computation offloading in mobile edge networks. *Ad Hoc Networks, 169*, 103754. https://doi.org/10.1016/j.adhoc.2025.103754

21. Yang, S., Zhang, L., Cui, L., Dong, Q., Xiao, W., & Luo, C. (2023). RLCS: Towards a robust and efficient mobile edge computing resource scheduling and task offloading system based on a graph neural network. *Computer Communications, 206*, 38–50. https://doi.org/10.1016/j.comcom.2023.04.020

22. Yu, X., Xia, C., Jin, X., Xu, C., Li, D., & Zeng, P. (2025). Integrated network-computing resource allocation and optimized scheduling for cyber-physical production system. *Ad Hoc Networks, 173*, 103831. https://doi.org/10.1016/j.adhoc.2025.103831

23. Zheng, Y., & Liu, Q. (2022). A review of distributed optimization: Problems, models and algorithms. *Neurocomputing, 483*, 446–459. https://doi.org/10.1016/j.neucom.2021.06.097

24. Zhou, G., Tian, W., Buyya, R., Xue, R., & Song, L. (2024). Deep reinforcement learning-based methods for resource scheduling in cloud computing: A review and future directions. *Artificial Intelligence Review, 57*, 124. https://doi.org/10.1007/s10462-024-10756-9

25. Ziaullah, Nawaz, R., Ullah, R., *et al.* (2026). AI-driven resource allocation in cloud computing: A systematic review revealing critical sustainability and evaluation gaps. *Computing*. https://doi.org/10.1007/s00607-026-01655-8