Proceedings of the International Conference on Industrial Engineering and Operations Management

Publisher: IEOM Society International, USA DOI: 10.46254/AP07.20260297
Published: March 25, 2026

Scenario Reduction for Two-Stage Stochastic Programming using Variance-Weighted K-means and Particle Swarm Optimization

Kanisorn Sritapak and Udom Janjarassuk
Department of Industrial Engineering, School of Engineering
King Mongkut's Institute of Technology Ladkrabang
Bangkok, 10520, Thailand
64601209@kmitl.ac.th, udom.ja@kmitl.ac.th

Abstract

Real-world optimization problems often involve uncertainties due to random parameters, leading to stochastic formulations that grow significantly in size as the number of scenarios increases, particularly in two-stage stochastic programming problems. A key challenge is managing the large number of scenarios, which contributes directly to the overall computational complexity. This paper introduces a hybrid scenario reduction method called Variance-weighted K-means with Particle Swarm Optimization (VwK-PSO). This method integrates variance-based analysis, clustering, and metaheuristic optimization to efficiently reduce the problem size while preserving essential stochastic characteristics from a scenario reduction perspective. The proposed method was tested on two case studies: cargo scheduling and aircraft allocation. Experimental results show that VwK-PSO outperforms traditional approaches like K-means clustering and Monte Carlo sampling in terms of solution quality and the Value of Stochastic Solution (VSS). Furthermore, this approach effectively maintains accuracy while significantly reducing computation time compared to the deterministic equivalent method. These findings highlight the effectiveness of VwK-PSO as a practical and scalable tool for addressing stochastic optimization problems, demonstrating that incorporating variable importance into scenario reduction can lead to more efficient and reliable decision-making in uncertain environments.

Keywords
Optimization, Scenario reduction, Particle swarm optimization, Metaheuristic, Two-stage stochastic programming

1. Introduction
Decision-making under uncertainty represents a fundamental challenge in operations research and industrial engineering. While deterministic optimization is well-established, it often fails to capture the inherent volatility of real-world parameters (Ning and You 2019). Consequently, Stochastic Programming (SP) has emerged as a robust framework for addressing problems where input parameters follow probabilistic distributions (Torres et al. 2022). SP has been successfully implemented across diverse sectors, including renewable energy management (Li et al. 2020), and inventory control (Johannsmann et al. 2022). These applications demonstrate that incorporating randomness is essential for reliability, yet it introduces significant mathematical complexity compared to traditional deterministic models.

The Two-Stage Stochastic Program (TSSP) is a dominant approach within this framework. Fundamentally, it divides the optimization process into two distinct periods: 'here-and-now' actions taken prior to any stochastic realization, and 'wait-and-see' recourse measures implemented once the random outcomes are observed (Grass and Fischer 2016). Uncertainty in TSSP is typically modeled using a set of discrete scenarios derived from historical data (Castellanos and Pozo 2021). However, achieving high-accuracy solutions requires a large number of scenarios. As the realization set grows, the problem size explodes, often rendering the model computationally intractable for practical applications. Therefore, scenario reduction—the process of selecting a compact but representative set—becomes a critical step to balance solution quality and computational efficiency.

To address these challenges, this study proposes a hybrid scenario reduction method named Variance-weighted K-means with Particle Swarm Optimization (VwK-PSO). Our approach assigns importance weights based on statistical variance, effectively prioritizing high-volatility features during the reduction process. This technique integrates variance analysis, K-means clustering, and metaheuristic optimization to preserve the essential stochastic characteristics of the original data. The proposed method is validated using standard benchmark problems. Performance is evaluated against the Deterministic Equivalent (DE) model to demonstrate significant reductions in computation time and the ability to capture the Value of Stochastic Solution (VSS), alongside comparisons with conventional reduction approaches.

1.1 Objectives

The primary objectives of this study are threefold.

1) To develop a novel hybrid scenario reduction approach that integrates variance-weighted K-means clustering with Particle Swarm Optimization for solving two-stage stochastic optimization problems.
2) To reduce computational time while producing a solution quality that remains close to that of the deterministic equivalent model.
3) To establish a scenario reduction framework that outperforms traditional approaches, specifically K-means clustering and Monte Carlo sampling, in terms of solution quality.

2. Literature Review

Scenario reduction is essential for maintaining the tractability of TSSP formulations. Early contributions established Forward Selection (FS) and Backward Reduction (BR), which aim to minimize the Wasserstein distance (WD) between original and reduced sets (Dupačová et al. 2003). However, these approaches lack theoretical bounds for solution quality. Recent advancements have introduced polynomial-time approximation algorithms that provide constant-factor performance guarantees (Rujeerapaiboon et al. 2022).

Clustering-based approaches, particularly K-means, have been widely adopted due to their efficiency in grouping scenarios (Arthur and Vassilvitskii 2007). Other variants, such as K-medians and K-modes, are utilized for specific data types (Keutchayan et al. 2023). However, standard clustering treats all random variables with equal weight, potentially overlooking parameters with high volatility that significantly impact the objective function. Alternatively, sampling techniques like Monte Carlo (MC) within the Sample Average Approximation (SAA) framework are simple to implement but often require extensive sample sizes to guarantee stability (Linderoth et al. 2006). To mitigate this, variance reduction methods such as Importance Sampling have been introduced (Papavasiliou and Oren 2013). To further enhance precision, Mixed-Integer Linear Programming (MILP) models utilizing the Wasserstein distance (WD) as an objective have been developed; however, while offering high accuracy, they incur prohibitive computational costs (Li and Li 2016).

In recent years, machine learning approaches, such as the use of mixed autoencoders for dimensionality reduction combined with clustering, have been proposed to balance computational efficiency and solution quality (Liang and Tang 2021). Neural network-based frameworks, including the Scenario Reduction Network, have also been advanced (Dong 2024). However, these methods typically require substantial data and significant training time. Additionally, an optimization-based scenario reduction method tailored to data-driven two-stage stochastic optimization models was developed (Bertsimas and Mundru 2022). Although this approach yields high-quality solutions, it often incurs substantial computational overhead due to problem complexity. Furthermore, reviews of intelligent optimization highlight swarm-based metaheuristics as the most widely applied methods (Mohammadi and Sheikholeslam 2023), showing significant potential for complex scheduling problems. Among these, Particle Swarm Optimization (PSO) is specifically recognized for its ability to effectively balance global and local search capabilities (Toaza and Esztergar-Kiss 2023).

Despite recent advancements, a critical limitation persists in existing reduction techniques: most methods operate in a 'feature-blind' manner, ignoring that varying random variables exhibit different levels of volatility and impact. In terms of complexity, traditional approaches (e.g., FS, BR, MC, and K-means) possess low algorithmic complexity but often struggle with solution quality and computational stability for large-scale problems. Current literature lacks a streamlined framework that integrates statistical feature weighting with clustering and metaheuristics to address these issues efficiently.

To bridge this gap, this study introduces the VwK-PSO algorithm. By combining variance-based weighting, K-means clustering, and PSO-based selection, the proposed approach prioritizes high-variance parameters, offering a practical balance between solution accuracy and execution time. Furthermore, the algorithm avoids the prohibitive computational burdens of MILP, optimization-based formulations, and data-intensive machine learning models. Consequently, its applicability is highly versatile, allowing it to efficiently solve diverse large-scale stochastic models without being strictly problem-dependent.

3. Methods

3.1 Stochastic optimization and Solving Framework

Stochastic Programming (SP) serves as a mathematical framework for optimization problems under uncertainty with a known distribution. This study specifically adopts the two-stage stochastic linear program with fixed recourse. Fundamentally, this formulation structures the optimization process into two distinct phases: here-and-now actions (first-stage), determined before the realization of uncertain parameters, and wait-and-see recourse measures (second-stage), executed to compensate for the actualized outcomes.

To ensure computational tractability, the random vector ξ is assumed to be discrete and finite. Let k = 1, 2, …, K denote the index of scenarios, where K represents the total number of possible realizations. Each scenario corresponds to a specific realization of the random vector with an associated probability p_k, such that $\sum_{k=1}^{K} p_k = 1$. Consequently, the problem can be transformed into a Deterministic Equivalent (DE) formulation in an extensive form. This large-scale linear programming model explicitly optimizes the first-stage decision variables (x) and the scenario-dependent second-stage variables (y_k) simultaneously. The objective is to minimize the total cost, comprising the deterministic first-stage cost ($c^T x$) and the expected second-stage recourse cost ($\sum_{k=1}^{K} p_k q_k^T y_k$), subject to the constraints defined by the technology matrix T_k and recourse matrix W. The DE formulation is defined in Eq. (1) (Birge and Louveaux 2011).

$$
\begin{aligned}
\min \quad & c^T x + \sum_{k=1}^{K} p_k q_k^T y_k \\
\text{s.t.} \quad & Ax = b, \\
& T_k x + W y_k = h_k, \quad k = 1, \dots, K; \\
& y_k \geq 0, \quad k = 1, \dots, K; \\
& x \geq 0
\end{aligned}
\tag{1}
$$

In this formulation, A and b represent the coefficient matrix and right-hand side vector for the first-stage constraints, respectively, while h_k denotes the right-hand side vector for the second stage. To evaluate the benefit of incorporating uncertainty into the decision-making process, this study utilizes the Value of the Stochastic Solution (VSS). The VSS quantifies the cost advantage of solving the stochastic recourse problem compared to the expected result of using the mean value solution. This indicates that considering stochasticity yields a superior result than relying solely on average parameter values.

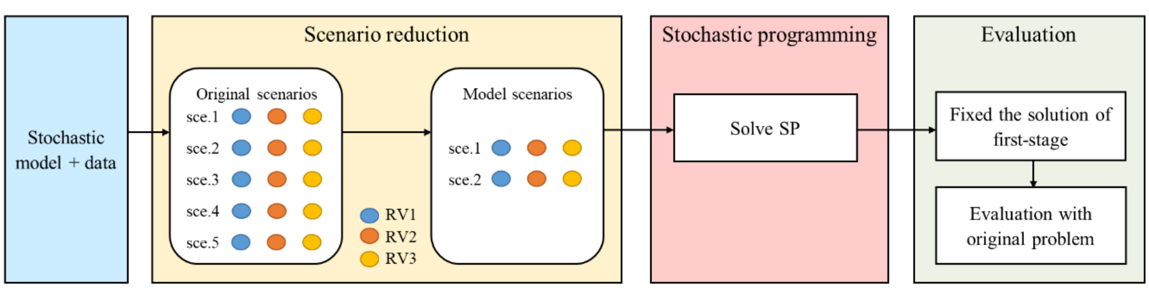

Figure 1. Framework for solving TSSP (sce. = Scenario, RV = Random Variable)

The overall solution methodology for solving TSSP is illustrated in Figure 1. The process begins with defining the stochastic data, which represents the uncertainty within the problem. Next, a data preprocessing step is performed using the scenario reduction method. This step extracts a compact representative subset while preserving key characteristics. The reduced scenario set is then utilized to solve the TSSP. Finally, the obtained first-stage solution is fixed and evaluated against the original problem to assess its true performance and computation time. This workflow ensures that each technique is validated rigorously against the complete uncertainty set.

3.2 Scenario reduction technique

This study employs a discrete scenario reduction strategy to select a representative subset directly from the original dataset. Initially, the complete set of scenarios is generated, where each scenario S_k is constructed by combining realizations from the random variables, and its associated probability p_k is computed as the product of the corresponding marginal probabilities. The proposed method is VwK-PSO. The process begins with variance filtering, which assigns weights to each random variable based on its variance. Next, K-means clustering is applied to group scenarios based on their centroids. Finally, PSO is used to select representative scenarios from each cluster. The complete procedure is outlined in Algorithm 1.

Step 1: Variance-based feature weighting.
The process begins by analyzing the volatility of each random variable. Let D_j represent the vector of random variable j. To ensure comparability across different scales, the data is normalized relative to its expected value. Subsequently, the variance is computed to quantify the dispersion of each parameter, as defined in Eq. (2).

$$Var(D_j) = \sum_{l=1}^{L_j} \left(\left(v_{j,l}^{norm} - \mathbb{E}(D_j) \right)^2 \cdot p_{j,l} \right), \qquad \forall j \tag{2}$$

where $v_{j,l}^{norm}$ denotes the normalized value of the l-th data point of random variable j, $p_{j,l}$ represents its corresponding probability, and L_j denotes the total number of data points for random variable j. Once the variance for each random variable is computed, a hierarchical ranking system based on percentiles is designed to assign importance weights (w_j). These variables are classified into four priority levels, as shown in Table 1. This quartile-based thresholding is designed to retain all parameters rather than eliminating those with minimal dispersion. Systematic weighting ensures that highly fluctuating factors appropriately govern the scenario selection in the PSO process.

Table 1. Weighting of random variables

Rank	Percentile (Pct)	Weight
1	Pct <= 25	0.25
2	25 < Pct <= 50	0.50
3	50 < Pct <= 75	0.75
4	Pct > 75	1.00

Step 2: Scenario clustering.
This phase begins by defining the target number of clusters (M), which dictates the size of the reduced dataset. The partitioning is executed using the K-means algorithm, implemented via the Scikit-learn library (Arthur and Vassilvitskii 2007). In this process, realizations are assigned to the nearest centroid based on the Euclidean distance, with centroids iteratively refined until convergence is reached. To preserve the total probability mass, the probability of each resulting cluster (p'_i) is computed by aggregating the probabilities of its constituent members, as formulated in Eq. (3).

$$p'_i = \sum_{S_k \in G_i} p_k \tag{3}$$

where S_k denotes the k-th scenario belonging to the set G_i of cluster i, and p_k represents the corresponding probability of scenario S_k.

Algorithm 1: VwK-PSO

Input:
  Random variable vectors D_j and probabilities $P(D_j)$ for $j \in \{1, 2, \dots, N\}$
  Original scenario set $\mathcal{S} = \{S_1, S_2, \dots, S_K\}$ with probabilities $\{p_1, p_2, \dots, p_K\}$
  Number of clusters (also reduced scenarios) = M
  PSO parameters: Number of particles, number of iterations, inertia weight, cognitive coefficients, social coefficients
Output:
  Reduced scenario set $\mathcal{S}' = \{S'_1, S'_2, \dots, S'_M\}$ and adjusted probabilities $\{p'_1, p'_2, \dots, p'_M\}$

// *Step 1: Variance-based feature weighting*
for each random variable j = 1 to N **do**
  Normalize data relative to its expected value
    for each distinct value l = 1 to L_j **do**
      Compute variance $Var(D_j)$ using Eq. (2)
    end for
end for
Assign importance weight w_j based on ranking (Table 1)

// *Step 2: Scenario clustering*
Partition all scenarios $\mathcal{S}$ into M clusters $\{G_1, G_2, \dots, G_M\}$ using K-means clustering
for each cluster i = 1 to M **do**
  Calculate cluster probability $p'_i \leftarrow \sum_{S_k \epsilon G_i} p_k$ using Eq. (3)
end for

// *Step 3: Representative selection via PSO*
for each cluster i = 1 to M **do**
  Construct empirical marginal distributions F_j ← scenarios in G_i
  Initialize swarm particles within the domain of cluster G_i
  while termination criteria not met **do**
    for each particle k **do**
      Calculate Fitness Z using Eq. (4) and (5)
      if Z < Fitness ($pbest$) **then** update personal best ($pbest$)
      if Z < Fitness ($gbest$) **then** update global best ($gbest$)
    Update velocity and position
    end for
  end while
  Set representative S'_i ← global best position
end for

return $\{S'_1, S'_2, \dots, S'_M\}$ and $\{p'_1, p'_2, \dots, p'_M\}$

Step 3: Representative selection via PSO.
To identify the optimal representative for each cluster, a Particle Swarm Optimization (PSO) metaheuristic is utilized, implemented via the pyswarm library version 0.6 (Lee 2014). The primary objective is to minimize the aggregate weighted 1-Wasserstein distance (WD_1) between the value of random variable j from candidate scenario (S_k) and the empirical distribution (F_j) of the corresponding group. The WD_1 is computed for every random variable ($j = 1, \dots, N$) and synthesized using the importance weights (w_j) derived in Step 1. The objective is to minimize the total weighted distance, denoted as the fitness value Z, as formulated in Eq. (4). The distance is calculated in Eq. (5), where $v_{j,l}$ denotes the l-th distinct value observed within the cluster (up to L'_j) and $p_{j,l}$ represents its associated probability.

$$\min \quad Z = \sum_{j=1}^{N} w_j \cdot WD_1(S_{k,j}, F_j) \tag{4}$$

$$WD_1(S_{k,j}, F_j) = \sum_{l=1}^{L'_j} |S_{k,j} - v_{j,l}| \cdot p_{j,l} \tag{5}$$

In this process, each particle encodes a potential representative scenario. The swarm evolves through standard velocity and position update rules. The particle yielding the minimum weighted distance is designated as the reduced scenario for the respective cluster. Regarding the algorithmic configuration, an inertia weight of 0.5 and both global and personal acceleration coefficients of 1.49 are adopted from the literature (Hong et al. 2019). Experimentally, the population size is set to 20 particles, with a termination of 100 iterations, which was empirically found to strike a balance between identifying high-quality representatives and maintaining computational efficiency.

Upon completion of the reduction process, the resulting subset serves as the input for solving the SP formulations. In addition, our proposed approach is compared with two conventional scenario reduction methods: a clustering-based approach and a sampling-based approach. The first method applies the K-means clustering algorithm to partition the original scenario set into clusters (Arthur and Vassilvitskii 2007), where the scenario closest to the centroid of each cluster is selected as the representative. The probability of each representative scenario is then assigned as the sum of the probabilities of all scenarios within its cluster, and this method is referred to as K-means in our results. The second method is a Monte Carlo sampling approach, which randomly selects scenarios from the original set with replacement. This follows the baseline Monte Carlo sampling method (Bertsimas and Mundru 2022) and is referred to as MC in our results.

3.3 Case studies

In this study, the performance of the proposed method is evaluated using two standard benchmark problems for two-stage stochastic optimization.

The first benchmark is the Cargo Scheduling Problem (Cargo) (Ariyawansa and Felt 2002), which is adapted from a real-world case (Mulvey and Ruszczynski 1995), modeled as a two-stage stochastic mixed-integer linear program. This problem involves four stations, where the flight schedule is determined in the first stage. However, the amount of cargo to be transported is uncertain, represented by 12 independent random variables for the maximum problem size. The uncertainty requires recourse actions to allocate cargo to available flights in the second stage, with a penalty cost for undelivered cargo. The objective is to minimize the total transportation cost. The problem formulation consists of 238 decision variables (52 first-stage and 186 second-stage), and 88 constraints define the feasibility conditions for the scheduling process.

The second instance is the Aircraft Allocation Problem (GBD) (Dantzig 1963), which involves assigning different types of aircraft to specific routes to maximize profit under uncertain passenger demand. The model considers operating costs and penalty costs for exceeding available capacity. It consists of four aircraft types operating on five routes. The first-stage variables, totalling 17, determine the number of aircraft of each type assigned to each route. This stage has 4 constraints to ensure the number of assigned aircraft does not exceed the available fleet for each type. In addition, the second-stage variables comprise 10 variables, including the number of passengers transported and the number of passengers exceeding capacity on each route. The second stage also contains 5 constraints representing demand balance equations for the five routes. The uncertain demand is assumed to follow a discrete probability distribution, which consists of 5 independent random variables, as detailed in the literature (Linderoth et al. 2006).

4. Results and Discussion

This section presents the results from two problems, demonstrating the proposed hybrid scenario reduction method for TSSP. The implementation was done in Python using Pyomo for modeling (Hart et al. 2017). The solver is Gurobi, tested in a Jupyter Notebook environment. The experiments were conducted on a computer with the following specifications: [Intel ® Core™ i5-1035G1 CPU @ 1.00GHz and RAM 8.0 GB]. In addition, the two-stage stochastic optimization problems were solved by DE method to obtain the optimal solution. The results were summarized in terms of the objective value and computation time. Then, the proposed method was further tested with reduced scenario sets of 25, 50, and 100 scenarios. The performance was compared with MC and K-means. Each method was tested five times to compute the average results, and the first-stage solutions were evaluated against the full scenario set. Moreover, all approaches were compared with the mean-value approach, where the decision-maker did not have a stochastic programming tool for assessing the value of the stochastic solution (VSS).

4.1 Testing of cargo scheduling problem

The experiment tested the problem with a maximum of 32,768 scenarios. The objective of the problem was to minimize total transportation costs. The test setup followed the data specifications (Ariyawansa and Felt 2002). The optimal objective value of this case with 32,768 scenarios obtained by DE was $446.86, and the required computation time was 8,389.3 seconds. In addition, the proposed method (VwK-PSO) and other scenario reduction approaches (K-means, MC) were solved with 25, 50, and 100 scenarios. Then, the experiment of each was evaluated with the original size, which required an evaluation time of approximately 420 to 540 seconds. The results, including the average objective values and run times for each method, are presented in Table 2.

Table 2. Performance comparison of scenario reduction methods for the Cargo problem (Results are averages of 5 runs. Solve time includes scenario reduction overhead. Evaluation is the performance on the full scenario set.)

#Scenario	Methods	Solve		Evaluation	
		Avg. obj. ($)	Solve time (s)	Avg. obj. ($)	Std.
25	K-means	429.68	2.41	1,602.04	19.83
	MC	444.18	2.24	534.53	51.75
	VwK-PSO	446.07	40.78	472.32	31.91
50	K-means	436.41	3.72	981.51	18.94
	MC	444.76	3.24	493.08	46.15
	VwK-PSO	446.72	78.42	446.86	0.00
100	K-means	444.08	6.75	576.87	86.49
	MC	446.29	5.71	456.94	20.16
	VwK-PSO	446.87	152.65	446.86	0.00

As shown in Table 2, the objective value of hybrid scenario reduction method outperformed other approaches for the same number of scenarios, followed by MC, and K-means respectively. However, it required more computation time in the case of VwK-PSO. Furthermore, VwK-PSO demonstrated high stability, achieving the lowest standard deviations at 50 and 100 scenarios compared to other methods. In addition, the quality of the solution was enhanced as the number of scenarios was increased in the cargo problem. Based on the results in Table 2, the proposed method was further compared with the mean-value approach. This evaluation aimed to assess the value of the stochastic solution (VSS) and to compare the performance of the proposed method, along with traditional scenario reduction techniques, against the DE approach. The results of this comparison are presented in Table 3.

Table 3. Comparison of the proposed method and benchmarks for the Cargo problem

Methods	Obj. value ($)	Solve time (s)	Optimality gap (%)	VSS
DE	446.86	8389.30	-	2,611.40
Mean-value	3,058.26	0.42	584.39	-
K-means (#100)	576.87	6.75	29.09	2,481.39
MC (#100)	456.94	5.71	2.26	2,601.32
VwK-PSO (#100)	446.86	152.65	0.00	2,611.40

As shown in Table 3, the value of the stochastic solution (VSS) for VwK-PSO was $2,611.40, accounting for approximately 85.39%. Therefore, the objective value obtained by the proposed method with 100 scenarios (evaluated against the full problem size) showed no significant deviation from the optimal solution obtained by DE. Furthermore, in terms of computation time, solving the TSSP using VwK-PSO with 100 scenarios significantly reduced the required time—from 8,389.30 seconds (with DE) to 152.65 seconds—representing a reduction of 8,236.65 seconds, or approximately 98.18%. In addition, the VwK-PSO solution with 100 scenarios was evaluated on the full problem set, and the computation time for this evaluation was also considered.

4.2 Testing of aircraft allocation problem

The test setup followed the data specifications (Linderoth et al. 2006). The optimal objective value of this case with 646,425 scenarios obtained by DE was $1,655.63, and the required computation time was 5,226.0 seconds. In addition, the proposed method and other approaches were tested with 25, 50, and 100 scenarios. Then, each resulting solution was evaluated with the original size, which required an evaluation time of approximately 480 to 600 seconds. The results, including the average objective values and run times for each method, are presented in Table 4.

Table 4. Performance comparison of scenario reduction methods for the GBD problem (Results are averages of 5 runs. Solve time includes scenario reduction overhead. Evaluation is the performance on the full scenario set.)

#Scenario	Methods	Solve		Evaluation	
		Avg. obj. ($)	Solve time (s)	Avg. obj. ($)	Std.
25	K-means	1,720.03	2.67	1,664.23	19.93
	MC	1,565.50	1.07	1,666.14	12.24
	VwK-PSO	1,546.17	30.35	1,658.72	0.54
50	K-means	1,714.93	4.47	1,673.78	4.78
	MC	1,692.77	2.51	1,670.66	13.24
	VwK-PSO	1,554.54	59.54	1,658.51	0.56
100	K-means	1,689.94	6.63	1,672.06	15.73
	MC	1,669.67	1.24	1,664.09	5.95
	VwK-PSO	1,569.22	91.24	1,659.11	0.00

From Table 4, the results for the GBD problem showed that the proposed method provided better average objective values than other approaches for the same number of scenarios. However, it required more computation time in the case of VwK-PSO. Moreover, the standard deviation of VwK-PSO was the lowest across all levels, demonstrating its high stability and precision, while K-means and MC exhibited higher variability. In addition, the proposed method was further compared with the mean-value approach, and against the DE approach to describe the performance. The results of this comparison are shown in Table 5.

Table 5. Comparison of the proposed method and benchmarks for the GBD problem

Methods	Obj. value ($)	Solve time (s)	Optimality gap (%)	VSS
DE	1,655.63	5,226.02	-	361.91
Mean-value	2,017.54	0.44	21.86	-
K-means (#100)	1,672.06	6.63	0.99	345.48
MC (#100)	1,664.09	1.24	0.51	353.45
VwK-PSO (#100)	1,659.11	91.24	0.21	358.43

As shown in Table 5, the value of the stochastic solution (VSS) for VwK-PSO was $358.43, accounting for approximately 17.77%. The objective value obtained by the proposed method with 100 scenarios (evaluated against the full problem size) differed from the optimal solution obtained by the DE approach by only 0.21%. Furthermore, in terms of computation time, solving the TSSP using VwK-PSO with 100 scenarios significantly reduced the required time—from 5,226.02 seconds (with DE) to 91.24 seconds—representing a reduction of 5,134.78 seconds, or approximately 98.25%. Moreover, the VwK-PSO solution with 100 scenarios was evaluated on 646,425 scenarios, and the computation time for this evaluation was also considered.

4.3 Discussion

The research findings indicate that the proposed scenario reduction method is effective in selecting representative scenarios by considering both the importance of each random variable and the scenario characteristics in solving TSSP problems. This approach results in high-quality solutions compared to traditional scenario reduction techniques, such as Monte Carlo sampling, which randomly selects scenarios without considering data structure, and K-means clustering, which selects representatives based on proximity to the cluster centroids. Moreover, the proposed method exhibits lower variability in the results, reflecting greater accuracy and stability. In addition, the parameters used in our approach were not tuned to find the optimal configuration, but were instead assigned fixed values for simplicity. These include parameters in both the PSO process—such as the number of particles in the swarm, velocity-related parameters, and the number of iterations—and the number of clusters. Additionally, the percentile weights are simple heuristic choices, and relying solely on variance remains a limitation concerning cost impacts.

In this study, the scenario reduction approach was designed to use less than 1% of the original number of scenarios. The method applies discrete scenario reduction, where representative scenarios are selected from actual or original data without generating new ones. In addition, all solution evaluations were conducted using the original full-scale problems to maintain the integrity and comparability of the results. The findings suggest that increasing the number of representative scenarios generally improves solution quality; however, this is not always guaranteed. A higher number of scenarios also leads to longer computation times. Therefore, determining an appropriate number of scenarios is critically important in practical applications.

The complexity of TSSP problems varies significantly depending on the specific characteristics of each case, including the objective function, constraints, random variables, and scenario set size. For instance, in the cargo problem, the proposed approach achieved satisfactory results using only 50 or 100 representative scenarios. Moreover, in the GBD problem, all reduction methods—K-means, MC, and VwK-PSO—yielded similarly good outcomes, with VwK-PSO delivering the best results, albeit with higher computational cost. These findings highlight the importance of both the number of scenarios and the choice of scenario reduction approach. Regarding scalability limits, larger problem sizes inevitably lead to longer execution times for the proposed method, particularly during the clustering and representative selection phases. Additionally, the evaluation time also escalates significantly as the dataset grows. For practical implementation, decision-makers can determine an appropriate number of scenarios and select a technique based on the specific characteristics and computational constraints of the problem at hand. Furthermore, users can assess the solutions using a satisfactory and manageable subset of scenarios.

5. Conclusion

This study proposed a scenario reduction method named VwK-PSO, which integrates variance-based weighting of random variables and K-means clustering for scenario grouping, followed by PSO to select representative scenarios for efficiently solving two-stage stochastic optimization problems. The experimental results demonstrated that the proposed approach outperformed traditional scenario reduction techniques such as K-means clustering and Monte Carlo sampling. It significantly improved the objective value compared to the mean-value approach, which represents a decision-making strategy without stochastic programming. The proposed method captured 85.39% and 17.77% of the VSS in the cargo scheduling and aircraft allocation problems, respectively. In addition, it effectively reduced computation time by 8,236.65 seconds for the cargo scheduling problem and by 5,134.78 seconds for the aircraft allocation problem when compared to the DE approach, which provided the optimal solution. These findings suggest that the proposed VwK-PSO method can serve as an effective alternative scenario reduction technique. It demonstrates the capability to reduce computation time while maintaining high solution quality in solving two-stage stochastic programming problems, thereby aligning with the research objectives. Future research could extend this approach to closed-loop stochastic programming, where feedback from solution outcomes is incorporated into the decision-making process. This extension could facilitate the development of an automated, data-driven scenario reduction framework, thereby enhancing adaptability and computational efficiency in solving stochastic optimization problems.

References

Ariyawansa, K. A. and Felt, A. J., On a new collection for Stochastic linear programming test problems, Available: https://www4.uwsp.edu/math/afelt/slptestset/download.html, Accessed on Mar. 15, 2024.

Arthur, D. and Vassilvitskii, S., k-means++: the advantages of careful seeding, *Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1027-1035, New Orleans, USA, January 7-9, 2007.

Bertsimas, D. and Mundru, N., Optimization-based scenario reduction for data-driven two-stage stochastic optimization, *Operations Research*, vol. 71, no. 4, pp. 1343-1361, 2022.

Birge, J. R. and Louveaux, F., *Introduction to Stochastic Programming*, 2nd ed., Springer, New York, NY, USA, 2011.

Castellanos, A. G. and Pozo, D., An Optimal Scenario Reduction Method for Stochastic Power System Problems, *Proceedings of the 2021 IEEE Madrid PowerTech*, Madrid, Spain, 2021.

Dantzig, G. B., The Allocation of Aircraft to Routes Under Uncertain Demand, *Linear programming and extensions*, Princeton University Press, 1963.

Dong, X., Scenario Reduction Network Based on Wasserstein Distance With Regularization, *IEEE Transactions on Power Systems*, vol. 39, no. 1, 2024.

Dupačová, J., Gröwe-Kuska, N. and Römisch, W., Scenario reduction in stochastic programming: An approach using probability metric, *Mathematical Programming*, vol. 95, pp. 493-511, 2003.

Grass, E. and Fischer, K., Two-stage stochastic programming in disaster management: A literature survey, *Surveys in Operations Research and Management Science*, vol. 21, pp. 85-100, 2016.

Hart, W. E., Laird, C. D., Watson, J.-P., Woodruff, D. L., Hackebeil, G. A., Nicholson, B. L. and Siirola, J. D., *Pyomo — Optimization Modeling in Python*, 2nd ed., Springer, Cham, Switzerland, 2017.

Hong, Y., Choi, B. and Kim, Y., Two-stage Stochastic Programming Based on Particle Swarm Optimization for Aircraft Sequencing and Scheduling, *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 4, 2019.

Johannsmann, L. M., Craparo, E. M., Dieken, T. L., Fugenschuh, A. R. and Seitner, B. O., Stochastic mixed-integer programming for a spare parts inventory management problem, *Computers & Operations Research*, vol. 138, Art. no. 105568, 2022.

Keutchayan, J., Ortmann, J. and Rei, W., Problem-driven scenario clustering in stochastic optimization, *Computational Management Science*, vol. 20, Art. no. 13, 2023.

Lee, A., pyswarm: A gradient-free, evolutionary optimization package for Python that supports constraints, version 0.6, Available: https://pypi.org/project/pyswarm/, October 22, 2014.

Li, J., Zhou, J. and Chen, B., Review of wind power scenario generation methods for optimal operation of renewable energy systems, *Applied Energy*, vol. 280, Art. no. 115992, 2020.

Li, Z. and Li, Z., Linear programming-based scenario reduction using transportation distance, *Computers & Chemical Engineering*, vol. 88, pp. 50-58, 2016.

Liang, J. and Tang, W., Scenario reduction for stochastic day-ahead scheduling: A mixed autoencoder based time-series clustering approach, *IEEE Transactions on Smart Grid*, vol. 12, no. 3, pp. 2652–2662, 2021.

Linderoth, J., Shapiro, A. and Wright, S., The empirical behavior of sampling methods for stochastic programming, *Annals of Operations Research*, vol. 142, pp. 215-241, 2006.

Mohammadi, A. and Sheikholeslam, F., Intelligent optimization: Literature review and state-of-the-art algorithms (1965-2022), *Engineering Applications of Artificial Intelligence*, vol. 126, Art. no. 106959, 2023.

Mulvey, J. M. and Ruszczynski, A., A new scenario decomposition method for large-scale stochastic optimization, *Operations Research*, vol. 43, no. 3, pp. 477-490, 1995.

Ning, C. and You, F., Optimization under uncertainty in the era of big data and deep learning: When machine learning meets mathematical programming, *Computers & Chemical Engineering*, vol. 125, pp. 434-448, 2019.

Papavasiliou, A. and Oren, S. S., Multiarea stochastic unit commitment for high wind penetration in a trans mission constrained network, *Operations Research*, vol. 61, no. 3, pp. 578-592, 2013.

Rujeerapaiboon, N., Schindler, K., Kuhn, D. and Wiesemann, W., Scenario reduction revisited: Fundamental limits and guarantees, *Mathematical Programming*, vol. 191, pp. 207–242, 2022.

Toaza, B. and Esztergar-Kiss, D., A review of metaheuristic algorithm for solving TSP-based scheduling optimization problems, *Applied Soft Computing Journal*, vol. 148, Art. no. 110908, 2023.

Torres, J. J., Li, C., Apap, R. M. and Grossmann, I. E., A Review on the Performance of Linear and Mixed Integer Two-Stage Stochastic Programming Software, *Algorithms*, vol. 15, Art. no. 103, 2022.

Biographies

Kanisorn Sritapak is a Ph.D. student in the Department of Industrial Engineering at King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand. He received his Bachelor's degree in Industrial Engineering from King Mongkut's University of Technology North Bangkok in 2018 and his Master's degree in Industrial Engineering from Burapha University in 2020. His research interests include stochastic optimization, production scheduling, operations research, and metaheuristics.

Udom Janjarassuk is a lecturer in the Department of Industrial Engineering at King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand. He received his Ph.D. degree in Industrial Engineering from Lehigh University, USA in 2009. His research interests include stochastic optimization, supply chain management, discrete event simulation, metaheuristics, and operations research.