A New Framework for Simulation-Based Goal Programming Using Retrospective Optimization

Shing Chih Tsai

Department of Industrial and Information Management, National Cheng Kung University

No.1, University Road, Tainan City, Taiwan

sctsai@mail.ncku.edu.tw

Hao-Chun Lu [1]

Department of Management Science, National Yang Ming Chiao Tung University

No.1001, Daxue Road, Hsinchu City, Taiwan

haoclu@nycu.edu.tw

Honggang Wang

Department of Technology and Operations, California State Polytechnic University

3801 West Temple Avenue, Pomona, CA, USA

hwang@cpp.edu

July 9, 2025

[1]Corresponding author.

Abstract

In this paper, we present a simulation-based goal programming framework to address stochastic optimization problems with multiple performance measures. We specifically consider the weighted goal programming and minmax goal programming formulations, which are widely used in the field of goal programming. Due to the analytical intractability of the objective functions in this stochastic optimization model, traditional mathematical programming techniques cannot be readily applicable. To overcome this challenge, we propose using retrospective optimization algorithms, a type of simulation-optimization approach. To establish the validity of our method, we provide a convergence analysis of the proposed algorithms, demonstrating their ability to converge to a globally optimal solution under specific conditions. In addition to the theoretical aspects, in the experimental section, we use stochastic inventory optimization problems as an illustration. We present empirical results that showcase the effectiveness of our proposed framework in comparison to other benchmark solution approaches.

Keywords: Simulation; Goal programming; Retrospective optimization; Simulation optimization.

1 Introduction

Goal programming (GP) is a well-known optimization approach frequently used to handle problems involving multiple, often competing objectives. A detailed survey of GP can be found in Colapinto et al. (2017). The fundamental concept of GP is to identify the most satisfactory solution by minimizing the deviations from pre-specified aspiration levels. Conventional GP approaches typically assume that objective functions are *deterministic* and precisely defined. There are variations to the standard GP approach, including (i) Weighted GP (WGP), where the positive and negative deviations from goals are weighted according to the importance of the objectives. The model's purpose is to minimize the weighted sum of all goal deviations (i.e., the shortfall from achieving the goals); (ii) Minmax GP (MGP), which focuses on minimizing the maximum deviation among all goals. The purpose is to compress the range of deviations, reducing the extent to which any one goal is significantly unmet; (iii) Lexicographic GP, where multiple objectives are optimized sequentially according to their priority levels. The idea is to achieve the most important goal first, then move on to the next goal in the hierarchy, and so on. In this paper, we particularly focus on the WGP and MGP formulations due to their ease of implementation and popularity in the field of GP. WGP is more appropriate when there is a clear hierarchy of importance among the goals, as it allows for the expression of preferences through weights (Zografidou et al., 2016). MGP is ideal when all goals are equally important and a balanced achievement of all goals is desired, as it focuses on minimizing the worst-case deviation.

Practical decision problems often involve uncertainties (i.e., the objective functions are not deterministic), which necessitate the development of appropriate GP approaches. For instance, fuzzy GP is an extension of traditional goal programming that incorporates fuzzy set theory and was developed specifically to handle situations where decision-makers can only provide vague or imprecise goal values. The flexibility of fuzzy GP allows decision-makers to express these goals as approximate targets, accommodating real-world uncertainties (e.g., Özcan & Toklu, 2009; Dutta & Kumar, 2015). Another probable scenario is when objective functions involve stochastic components – meaning their values depend on uncertain parameters that follow probability distributions. However, these functions are still partially known in the sense that they can be expressed analytically, albeit with random elements (e.g., demand levels, costs, or resource availability following

known distributions such as normal or uniform). In such cases, adopting a stochastic GP approach, such as those proposed by Bravo and Gonzalez (2009) and Masri (2017), is justifiable. This approach is appropriate when sufficient historical data or knowledge exists to accurately model these uncertainties using probability distributions. Robust GP, on the other hand, does not rely on specific probability distributions. Instead, it seeks solutions that perform well across a range of possible scenarios or uncertainties, explaining uncertainty through set theory or uncertainty sets (e.g., Hanks et al., 2017; Lu & Tsai, 2024). The robust GP model typically sacrifices some level of optimality to ensure that the solution remains feasible and resilient, even under worst-case or uncertain conditions. It is worth noting that all the existing GP approaches designed to handle uncertain conditions (i.e., the aforementioned fuzzy GP, stochastic GP, and robust GP approaches) require an explicit form of the objective function (embedded with some uncertain parameters), such that traditional optimization techniques relying on gradients or explicit function forms are applicable. To the best of our knowledge, there is a lack of research on developing appropriate algorithms for GP formulations where the objective function is a true black box – that is, it has no explicit mathematical representation and can only be evaluated through simulation-based methods.

Simulation-based optimization (SO) is widely used for black-box functions, where the relationship between inputs and outputs is unknown but can be evaluated through simulation. The simulation model serves as a black box that evaluates given inputs and typically produces stochastic performance measures, reflecting randomness in system behavior. Since the objective function is not available in closed form, its performance can only be estimated through multiple simulation replications. In this paper, we explore the application of GP formulation within the context of simulation-optimization. SO is an effective approach for optimization problems under complex uncertainty, where explicit objective functions and constraints are absent (Hong et al., 2015). It aims to identify the set of decision variables that optimize the expected or probabilistic performance of a stochastic system using simulation. When the decision variables are discrete, various SO approaches are available, including random search algorithms (Zabinsky, 2015), ranking and selection (R&S) procedures (Hong et al., 2021), and retrospective optimization (Wang et al., 2013). Numerous studies have explored these methods for solving SO problems with multiple stochastic objectives or constraints (e.g., Park & Kim, 2015; Wang, 2017; Tsai et al., 2021b).

In this paper, we embrace the method of retrospective optimization, which is specifically de-

signed for stochastic or black-box problems whose objective functions are not given in closed form but must be estimated via simulation. Unlike standard (deterministic) optimization techniques – which typically assume an explicit model and allow for direct gradient computation – retrospective optimization proceeds by iteratively refining candidate solutions using progressively larger simulation sample sizes, thereby enhancing the statistical accuracy of performance estimates as the algorithm advances. Building on these principles, we present algorithms adapted from the Retrospective Search with Piecewise-Linear Interpolation and Neighborhood Enumeration (R-SPLINE) framework introduced by Wang et al. (2013). Specifically, R-SPLINE addresses sample-path problems by enlarging the sample size at each iteration and decreasing the error tolerance, using the SPLINE algorithm to integrate gradient- and local-search methods while leveraging prior solutions as warm starts. In the context of simulation-optimization, a sample-path refers to a specific realization of a stochastic system generated by running a simulation with a given set of decision variables. Since simulation outputs depend on random inputs, different sample-paths correspond to different observed outcomes. Sample-path optimization problems estimate the objective function and constraints based on a finite number of simulation replications, rather than assuming exact analytical formulations. The primary objective of R-SPLINE is to ascertain a local optimum of SO problems with discrete decision variables. Subsequently, Nagaraj and Pasupathy (2024) extended the capabilities of R-SPLINE to a more sophisticated cgR-SPLINE algorithm, enabling it to identify a global optimum for SO problems in the presence of stochastic constraints. Tsai et al. (2021a) also adapted R-SPLINE for security screening problems at land border crossing stations. Their approach maximized the probability of identifying threat travelers while ensuring that the expected waiting time and operating cost for travelers remained below certain thresholds. Tsai et al. (2024) further explored the stochastic production planning problem with product substitution, which can be decomposed into several optimization subproblems involving sequential decisions. They adapted R-SPLINE algorithms to address unique problem scenarios characterized by stochastic constraints due to uncertain product quality and substitution. Additional research on retrospective optimization has been conducted to address SO problems that involve multiple objective functions. One example is Wang's (2013) implementation of a gradient-based zigzag search method for multi-objective optimization problems. This approach explores the Pareto front by utilizing an efficient local search procedure. Wang (2017) subsequently proposed a novel retrospective optimization method for

solving multi-objective SO problems, building upon the methodology developed in Wang (2013). Furthermore, Wang et al. (2019) extended these approaches by introducing a new directed search method along the Pareto front for solving mixed-integer multi-objective SO problems.

This paper aims to develop a GP-based approach to solve SO problems with multiple performance measures, specifically considering WGP and MGP formulations. Since we focus on cases with a large number of candidate solutions, not all SO approaches are suitable for solving this problem. Therefore, this paper proposes the use of retrospective optimization approaches to address this issue. We introduce adapted R-SPLINE algorithms to solve the SO problem under the WGP and MGP formulations, referred to as WGP-SPLINE and MGP-SPLINE algorithms. To establish the validity of our method, we provide a convergence analysis of these algorithms, demonstrating their ability to converge to a globally optimal solution under specific conditions. In the experimental section, we illustrate the approach using stochastic inventory optimization problems. For a comprehensive overview of research studies that employ SO techniques to address practical inventory problems, please refer to Jalali and Nieuwenhuyse (2015).

The rest of the paper is organized as follows. In Section 2, we present the notation used in this work and provide detailed formulations of the GP problem. Section 3 outlines the proposed WGP-SPLINE and MGP-SPLINE algorithms to solve the stochastic optimization problem and provides a convergence analysis for each. In Section 4, we conduct an empirical evaluation of the proposed approach in the context of inventory optimization problems. Finally, we conclude the paper in Section 5 with final remarks. For brevity, a detailed description of the modified SPLINE algorithm and the benchmark R&S approach is provided in the Appendix, which is available as an online supplement.

2 GP Problem Formulation

Suppose there are R objective (performance) measures Z_r for $r \in \{1, \ldots, R\}$. Furthermore, suppose that each objective Z_r is associated with a target level of achievement, denoted by t_r, for each $r \in \{1, \ldots, R\}$. The deviations from these targets are represented by the positive and negative deviations, d_r^+ and d_r^-, respectively, where $d_r = d_r^+ - d_r^-$ and at most one of the two is nonzero. Let $\boldsymbol{X}$ be a vector of decision variables with d dimensions. The positive and negative deviations

are defined as follows: $d_r^+ = \max\{Z_r(\boldsymbol{X}) - t_r, 0\}$ and $d_r^- = \max\{t_r - Z_r(\boldsymbol{X}), 0\}$. We denote the relative weights of d_r^+ and d_r^- as w_r^+ and w_r^-, respectively. This deviation-based approach enables us to quantify the extent to which each objective's performance either exceeds or falls short of its target level, while also accounting for the relative importance of each objective.

The WGP formulation is presented as follows:

$$\text{WGP(P1)} \quad \text{Min} \quad \sum_{r=1}^{R}[w_r^- \times d_r^- + w_r^+ \times d_r^+] \tag{1}$$

$$\text{s.t.} \quad Z_r(\boldsymbol{X}) + d_r^- - d_r^+ = t_r, \quad \forall r \in \{1, \ldots, R\}, \tag{2}$$

$$\boldsymbol{X}^L \leq \boldsymbol{X} \leq \boldsymbol{X}^U, \tag{3}$$

$$d_r^-, d_r^+ \geq 0, \forall r \in \{1, \ldots, R\}. \tag{4}$$

Equation (1) is the overall goal programming function (i.e., the aggregated achievement function) in the WGP formulation. It represents the weighted sum of the positive and negative deviations from the specified target levels for all objectives. Note that for each objective, only one of the two deviations (either positive or negative) can be non-zero. Constraint (2) defines these positive and negative deviations for each objective with respect to a given solution $\boldsymbol{X}$. Since the performance measure $Z_r(\boldsymbol{X})$ does not admit a closed analytical form, it must be estimated using simulation experiments. Constraint (3) specifies the lower and upper bounds on the decision vector $\boldsymbol{X}$, and Constraint (4) imposes non-negativity restrictions on the deviation variables.

To directly solve Problem WGP(P1), we need to determine the feasibility of each solution $\boldsymbol{X}$ by assessing whether $Z_r(\boldsymbol{X}) \geq t_r$ for each $r \in \{1, \ldots, R\}$. Specifically, if $Z_r(\boldsymbol{X}) \geq t_r$, then $d_r^- = 0$ and we only need to deal with d_r^+. However, in the context of simulation, $Z_r(\boldsymbol{X})$ represents an unknown expected or probabilistic performance measure. For example, in healthcare scheduling, $Z_r(\boldsymbol{X})$ might represent the expected patient wait time under uncertain arrival processes. In quality control for manufacturing, it could denote the probability that a product meets quality standards, given uncertain production conditions. Due to random noise and finite sample variability, it is generally difficult to determine with statistical confidence whether $Z_r(\boldsymbol{X}) \geq t_r$ based on a limited number of simulation runs. Thus, even though the constraints (3) and (4) ensure that d_r^+ and d_r^- are well-defined for any given solution, deciding whether $d_r^- = 0$ (i.e., verifying that $Z_r(\boldsymbol{X}) \geq t_r$)

is non-trivial without extensive sampling. This feasibility determination problem in simulation has long been recognized and studied. In both the subareas of R&S and retrospective optimization, algorithms often incorporate a parameter called the "tolerance level" to relax the given threshold value concerning the stochastic constraints. However, selecting an appropriate "tolerance level" is challenging, as its value often needs to be adjusted based on the iteration number to ensure convergence and is not easy to interpret (e.g., Park & Kim, 2015; Lee et al., 2018; Nagaraj & Pasupathy, 2024).

An alternative approach to solve Problem P1 that avoids confronting the feasibility determination problem is to reformulate it as Problem P2, which is as follows:

$$\begin{aligned} \text{WGP(P2)} \quad \text{Min} \quad & \sum_{r=1}^{R} [w_r^- \times \max\{t_r - Z_r(\boldsymbol{X}), 0\} + w_r^+ \times \max\{Z_r(\boldsymbol{X}) - t_r, 0\}] \\ \text{s.t.} \quad & \boldsymbol{X}^L \leq \boldsymbol{X} \leq \boldsymbol{X}^U. \end{aligned} \tag{5}$$

Equation (5) is the objective function of the reformulated GP problem, which is a nonlinear function. If $Z_r(\boldsymbol{X})$ is a deterministic and explicit objective measure, Problem P2 would be more challenging to solve than Problem P1 due to the non-linearity of the objective function in Equation (5). However, when dealing with stochastic simulation, solving Problem P2 is easier than solving Problem P1 because it provides consistent estimators of Equation (5) without requiring the identification of the feasibility of each solution $\boldsymbol{X}$. More specifically, since we use simulation to estimate the desired performance measures for each candidate solution, the objective measure Z_r is replaced by the sample-path estimator $\widehat{Z}_{r,m}$ of sample size m, $\forall r \in \{1, \ldots, R\}$. That is, $\widehat{Z}_{r,m} = (1/m) \sum_{j=1}^{m} Z_{rj}$, where Z_{rj} represents the jth simulation observation of the objective measure Z_r. We then compute the estimated deviations: $\widehat{d}_r^- = \max\{t_r - \widehat{Z}_{r,m}(\boldsymbol{X}), 0\}$ and $\widehat{d}_r^+ = \max\{\widehat{Z}_{r,m}(\boldsymbol{X}) - t_r, 0\}$. These estimators can be directly used in the objective function of Equation (5) without explicitly determining whether $Z_r(\boldsymbol{X})$ meets the target t_r.

By employing this reformulation, we circumvent the need for a "tolerance level" and avoid the complexities associated with feasibility determination in stochastic settings. The use of sample-path estimators allows us to seamlessly integrate the stochastic nature of the simulation outputs into the optimization process. Moreover, the reformulated Problem P2 aligns well with common

optimization algorithms that can handle non-differentiable objective functions, such as stochastic approximation methods or derivative-free optimization techniques. These methods are well-suited for problems where the objective function involves max-functions or other non-smooth components resulting from simulation estimations. Consequently, Problem P2 provides a more flexible and efficient framework for addressing goal programming problems in stochastic simulation environments, facilitating better decision-making under uncertainty.

We also consider the MGP problem, whose original formulation is presented as follows:

$$
\begin{aligned}
\text{MGP(P1)} \quad \text{Min} \quad & T && (6)\\
\text{s.t.} \quad & w_r^- \times d_r^- + w_r^+ \times d_r^+ \leq T, \quad \forall r \in \{1, \ldots, R\}, && (7)\\
& Z_r(\boldsymbol{X}) + d_r^- - d_r^+ = t_r, \quad \forall r \in \{1, \ldots, R\},\\
& \boldsymbol{X}^L \leq \boldsymbol{X} \leq \boldsymbol{X}^U,\\
& d_r^-, d_r^+ \geq 0, \forall r \in \{1, \ldots, R\}.
\end{aligned}
$$

Equation (6) defines the objective function of the MGP formulation, where the goal is to minimize the maximum deviation, denoted as T, across all objectives. This approach focuses on minimizing the worst-case deviation from the goals to ensure a balanced achievement of all objectives. Equation (7) represents the constraint that the weighted sum of positive and negative deviations for each objective must not exceed T, effectively compressing the deviations of all objectives to lie within a common threshold. The second set of constraints ensures that the deviation variables accurately capture the extent to which each objective meets or deviates from its target. The third set of constraints ensures that the decision variables $\boldsymbol{X}$ respect the upper and lower bounds, while the final constraints enforce non-negativity on the deviation variables.

Similar to the reformulation for WGP, we present the reformulated MGP problem, referred to as MGP(P2), as follows:

$$
\begin{aligned}
\text{MGP(P2)} \quad \text{Min} \quad & \max_{r \in \{1,\ldots,R\}} [w_r^- \times \max\{t_r - Z_r(\boldsymbol{X}), 0\} + w_r^+ \times \max\{Z_r(\boldsymbol{X}) - t_r, 0\}] && (8)\\
\text{s.t.} \quad & \boldsymbol{X}^L \leq \boldsymbol{X} \leq \boldsymbol{X}^U.
\end{aligned}
$$

Equation (8) presents the objective function of MGP(P2), which seeks to minimize the maximum weighted deviation across all goals, using max-functions to capture the positive and negative deviations for each goal r. Notice that we can express the weighted deviations in Equation (7) as: $w_r^- \times \max\{t_r - Z_r(\boldsymbol{X}), 0\} + w_r^+ \times \max\{Z_r(\boldsymbol{X}) - t_r, 0\} \leq T$. Since T is minimized in MGP(P1), finding the smallest T that satisfies all these inequalities is equivalent to minimizing the maximum of the weighted deviations across all objectives, as in MGP(P2).

The max-function approach simplifies the estimation of deviations in the context of stochastic simulations, where the goals are subject to uncertainty. By utilizing simulation-based estimators, we can directly estimate the deviations without needing to determine feasibility explicitly. The reformulated MGP(P2) problem provides a more flexible and practical framework for solving goal programming problems under uncertainty, leveraging simulation outputs while minimizing the worst-case deviation across all objectives.

3 The Solution Approaches

In this section, we propose the WGP-SPLINE and MGP-SPLINE algorithms to solve Problems WGP(P2) and MGP(P2). The GP-SPLINE approach is adapted from the R-SPLINE framework introduced by Wang et al. (2013), which stands for "Retrospective Search with Piecewise-Linear Interpolation and Neighborhood Enumeration" and was originally designed for single-objective, integer-ordered simulation-optimization problems. The SPLINE algorithm is an integer-ordered optimization algorithm, meaning that the decision variables must take integer values that are naturally ordered. This differs from general discrete optimization, where variables may be categorical or unordered. The integer-ordered structure allows for efficient local search strategies, such as piecewise-linear interpolation and neighborhood enumeration, which iteratively refine solutions by exploring nearby integer values. First, let us introduce a high-level description of the R-SPLINE framework. Specifically, R-SPLINE solves a series of sample-path problems using SPLINE with progressively increasing sample sizes and common random numbers to reduce the variance of the estimators. The SPLINE algorithm iterates between two routines: (i) SPLI (search with piecewise-linear interpolation), which produces the next candidate solution via continuous piecewise-linear interpolation of the objective function in the sample-path problem; (ii) NE (neighborhood enumer-

ation), which ensures that the chosen solution is not inferior in terms of the sample-path function to any of the $2d$ neighboring solutions that are exactly one unit away in each dimension. Here, d represents the dimensionality of the decision vector $\boldsymbol{X}$. To ensure global convergence of the local optimum found by R-SPLINE, a multi-start approach is implemented, where SPLINE is repeatedly applied with randomly selected initial solutions (see Pardalos and Romeijn (2002) and Nagaraj and Pasupathy (2024) for similar methods).

3.1 The WGP-SPLINE Algorithm

We begin by considering WGP(P2). In this case, the objective measure Z_r is replaced by its sample-path estimator $\widehat{Z}_{r,m}$, which is based on a sample size of m, $\forall r \in \{1, \ldots, R\}$. Specifically, $\widehat{Z}_{r,m} = (1/m)\sum_{j=1}^{m} Z_{rj}$, where Z_{rj} represents the jth simulation observation of the objective measure Z_r. The resulting sample-path Problem WGP(P2) can be formulated as follows:

$$\begin{aligned} \text{WGP(P2}(m)) \quad \text{Min} \quad & \sum_{r=1}^{R} [w_r^- \times \max\{t_r - \widehat{Z}_{r,m}(\boldsymbol{X}), 0\} + w_r^+ \times \max\{\widehat{Z}_{r,m}(\boldsymbol{X}) - t_r, 0\}] \\ \text{s.t.} \quad & \boldsymbol{X}^L \leq \boldsymbol{X} \leq \boldsymbol{X}^U. \end{aligned}$$

The presence of multiple objectives and max-functions introduces non-differentiability and piecewise-defined regions in the objective function. To adapt the R-SPLINE framework for solving the WGP(P2) Problem, several key modifications are necessary due to the differences in the objective function. Below are the specific modifications required:

1. Handling Multiple Objectives

R-SPLINE is designed for single-objective optimization problems, whereas WGP(P2) involves multiple objectives aggregated into a single objective function using weighted sums of positive and negative deviations. WGP-SPLINE modifies the objective function to compute the weighted sum of deviations for all objective measures: $\widehat{g}(\boldsymbol{X}) = \sum_r [w_r^- \times \max\{t_r - \widehat{Z}_{r,m}(\boldsymbol{X}), 0\} + w_r^+ \times \max\{\widehat{Z}_{r,m}(\boldsymbol{X}) - t_r, 0\}]$. For each objective Z_r, WGP-SPLINE maintains separate simulation estimators $\widehat{Z}_{r,m}(\boldsymbol{X})$.

2. Managing Non-Differentiability Due to Max-Functions

The presence of max-functions introduces kinks in the objective function, leading to points of non-differentiability that standard gradient-based methods cannot handle directly. This occurs when

$\widehat{Z}_{r,m}(\boldsymbol{X}) = t_r$. Although this may seem uncommon, it is important to address in WGP(P2) for several reasons: (a) As $\boldsymbol{X}$ is updated during optimization, $\widehat{Z}_{r,m}(\boldsymbol{X})$ may oscillate around t_r due to estimation errors, frequently encountering regions near non-differentiable points. (b) Since $\boldsymbol{X}$ takes discrete values in WGP(P2), small changes in $\boldsymbol{X}$ can cause significant jumps in $\widehat{Z}_{r,m}(\boldsymbol{X})$, increasing the likelihood of hitting non-differentiable points. (c) Each max-function adds a kink at $\widehat{Z}_{r,m}(\boldsymbol{X}) = t_r$, and with multiple goals r, the number of kinks rises. In higher-dimensional problems, this cumulative effect makes non-differentiability more likely. (d) When target values t_r are set close to expected performance metrics, the optimization naturally drives solutions toward these targets. Since the max-function $\max\{t_r - \widehat{Z}_{r,m}(\boldsymbol{X}), 0\}$ is non-differentiable at t_r, small fluctuations due to simulation noise increase the likelihood of encountering these points. This effect is more pronounced when feasible candidates exist near t_r, as observed in our experiments (Section 4.2). (e) In multi-objective optimization, trade-offs between competing objectives often push solutions near multiple targets simultaneously, increasing the likelihood of encountering non-differentiable kinks at t_r, particularly near feasibility boundaries. This effect is well-documented by literature on multi-objective problems (e.g., Deb, 2001) and is consistent with our experimental results.

To handle the issue of non-differentiability in the WGP problem and enhance the WGP-SPLINE algorithm, we propose combining the use of subgradients with the cutting plane method. This combination allows us to effectively navigate the non-differentiable landscape of the objective function by: (1) computing subgradients at points of non-differentiability, providing valid search directions; (2) generating cutting planes using the subgradients, refining the feasible region or the approximation of the objective function. Subgradients are a generalization of gradients for convex functions that may not be differentiable everywhere. They provide a way to define a direction of descent even at points where the function is not differentiable.

See Appendix A.1 for details on how to modify each component of the WGP-SPLINE algorithm and its embedded algorithms (SPLINE, SPLI, and PLI) to incorporate these methods. A high-level description of the WGP-SPLINE approach is provided in Table 1. In each outer iteration h, an approximated local solution $\boldsymbol{X}_h$ is obtained by repeatedly solving a sequence of sample-path problems P2(m_k) for each inner iteration k. A randomly generated initial solution $\boldsymbol{X}_0^{(h)}$ is used to start the process (Step 1). In the k-th inner iteration, the sample-path problem P2(m_k) is solved using the SPLINE algorithm with the previous sample-path solution $\boldsymbol{X}_{k-1}^{(h)}$ as the initial solution.

The resulting solution $\boldsymbol{X}_k^{(h)}$ and the utilized simulation budget N_k are returned. The sample size sequence $\{m_k\}$ increases as the iteration proceeds to ensure the accuracy of the estimators. During each inner iteration k, SPLINE is terminated either when a local optimum is found or when the number of objective-function simulation estimates reaches the budget of v_k. At the end of each outer iteration h, the current best solution is recorded and possibly updated, referred to as the incumbent solution $\boldsymbol{X}_h^*$ (Step 10). The update is made when the estimated local optimal solution $\boldsymbol{X}_h$ is superior to the previous incumbent solution $\boldsymbol{X}_{h-1}^*$ in terms of the objective value of the sample-path problem P2(n_h), where n_h indicates the incumbent sample size (Step 9).

3.2 The MGP-SPLINE Algorithm

The GP-SPLINE algorithm can be readily adapted to solve Problem MGP(P2), resulting in the MGP-SPLINE algorithm. The primary difference lies in the objective function, which seeks to minimize the maximum weighted deviation across all goals. The sample-path problem for MGP(P2) can be formulated as:

$$\begin{aligned} \text{MGP(P2}(m)) \quad \text{Min} \quad & \max_{r\in\{1,\ldots,R\}} [w_r^- \times \max\{t_r - \widehat{Z}_{r,m}(\boldsymbol{X}), 0\} + w_r^+ \times \max\{\widehat{Z}_{r,m}(\boldsymbol{X}) - t_r, 0\}] \\ \text{s.t.} \quad & \boldsymbol{X}^L \le \boldsymbol{X} \le \boldsymbol{X}^U. \end{aligned}$$

In this formulation, the objective function involves a max-min structure, introducing additional non-differentiability and non-smoothness compared to the WGP(P2) problem. The presence of the outer maximum operator means that, at any given point $\boldsymbol{X}$, the objective function $g(\boldsymbol{X})$ is equal to the largest weighted deviation among all goals. The subgradient of $g(\boldsymbol{X})$ depends on which goal(s) are active, i.e., which goal(s) attain the maximum weighted deviation at $\boldsymbol{X}$. We consider the following cases:

Case 1. Single Active Goal

If only one goal r^* attains the maximum weighted deviation at $\boldsymbol{X}$, then the subgradient of $g(\boldsymbol{X})$ is simply the subgradient of the weighted deviation for that goal: $\gamma = \nabla_{\boldsymbol{X}} d_{r^*}(\boldsymbol{X})$, where $d_{r^*}(\boldsymbol{X}) = w_{r^*}^- \times \max\{t_{r^*} - \widehat{Z}_{r^*,m}(\boldsymbol{X}), 0\} + w_{r^*}^+ \times \max\{\widehat{Z}_{r^*,m}(\boldsymbol{X}) - t_{r^*}, 0\}$. We compute this subgradient using the subgradient rules for max-functions, as discussed in Section A.1.

Case 2. Multiple Active Goals

Table 1: The WGP-SPLINE algorithm for solving Problem WGP(P2).

Input: Problem P2 parameters: $\{w_r^-, w_r^+, t_r\}$, $\forall r \in \{1, \ldots, R\}$; and R-SPLINE algorithm parameters: $\{b_h, m_k, v_k\}$.

Output: Global optimal solution $\boldsymbol{X}^*$ for Problem P2.

Initialize: incumbent solution $\boldsymbol{X}_0^*$ and incumbent sample size $n_0 = 0$.

For $h = 1, 2, \ldots$ {outer iterations}

1. Randomly select an initial solution $\boldsymbol{X}_0^{(h)}$.
2. Initialize: budget utilized $u_0 = 0$ and inner iteration number $k = 0$.
3. **While** $u_k \leq b_h$ {budget utilized does not exceed outer simulation budget}
4. Update: inner iteration $k \leftarrow k + 1$.
5. Solve sample-path problem P2(m_k) using SPLINE algorithm:

$$(N_k, \boldsymbol{X}_k^{(h)}) = \text{SPLINE}(\boldsymbol{X}_{k-1}^{(h)}, m_k);$$

   SPLINE is terminated when either a local minimum is identified or the number of objective-function estimates exceeds v_k.

6. Update: utilized budget $u_k = u_{k-1} + N_k$.
7. **End While**
8. Update: returned local solution $\boldsymbol{X}_h = \boldsymbol{X}_k^{(h)}$.
9. Update: incumbent sample size $n_h = \max\{n_{h-1}, m_k\}$.
10. Update: incumbent solution $\boldsymbol{X}_h^* = \arg\min_{\boldsymbol{z}} \{\widehat{g}(\boldsymbol{z}) : \boldsymbol{z} \in \{\boldsymbol{X}_{h-1}^*, \boldsymbol{X}_h\}\}$, where $\widehat{g}(\boldsymbol{z})$ is the objective value of sample-path Problem P2.

End For

While it is unlikely in practice due to simulation noise and numerical precision, there may be instances where multiple goals are tied for the maximum weighted deviation at $\boldsymbol{X}$. In such cases, let $\mathcal{R}^*$ denote the set of active goals where: $g(\boldsymbol{X}) = d_r(\boldsymbol{X})$, $\forall r \in \mathcal{R}^*$. Here, the subgradient of $g(\boldsymbol{X})$ is any convex combination of the subgradients of the weighted deviations for the active goals: $\gamma = \sum_{r \in \mathcal{R}^*} \lambda_r \gamma_r$, where $\lambda_r \geq 0$, $\sum_{r \in \mathcal{R}^*} \lambda_r = 1$, and $\gamma_r = \nabla_{\boldsymbol{X}} d_r(\boldsymbol{X})$ is the subgradient of the weighted deviation for goal r. A common choice is to assign equal weights, i.e., $\lambda_r = 1/|\mathcal{R}^*|$.

The other parts and modifications of the MGP-SPLINE algorithm are similar to those of the WGP-SPLINE algorithm. This includes the overall algorithm structure, the use of common random numbers for variance reduction, and the multi-start approach to enhance the likelihood of finding the global optimum. The major difference is that we first need to identify the active goal r^* or the set of active goals $\mathcal{R}^*$ tied for the maximum weighted deviation. We then compute the related subgradients accordingly. If multiple goals share the maximum value within a specified tolerance (to account for numerical precision), we consider them as active.

The next subsection presents the convergence analysis and proof of the WGP-SPLINE algorithm, demonstrating its ability to converge to a global optimum under specific conditions. The convergence analysis for MGP-SPLINE differs from that of WGP-SPLINE due to appropriate adjustments made for the structure of the objective function.

3.3 The Convergence Analysis of WGP-SPLINE Algorithm

To facilitate the discussion on optimality and convergence analysis, we introduce additional notation. Let $\mathbb{X}$ denote the feasible solution set of Problem WGP(P2), where each $\boldsymbol{X} \in \mathbb{X}$ represents a feasible solution. As the feasible region of WGP(P2) is bounded and consists of integer values, the set $\mathbb{X}$ is finite, implying the existence of one or more global optimal solutions. The objective function of WGP(P2) is represented by $g(\boldsymbol{X})$, i.e., $g(\boldsymbol{X}) = \sum_{r=1}^{R} [w_r^- \times \max\{t_r - Z_r(\boldsymbol{X}), 0\} + w_r^+ \times \max\{Z_r(\boldsymbol{X}) - t_r, 0\}]$. We also define the set $\mathbb{X}^*$ of local optimal solutions for WGP(P2). A solution $\boldsymbol{X}^* \in \mathbb{X}^*$ is a local optimum if it satisfies: $g(\boldsymbol{X}^*) \leq g(\boldsymbol{X})$, $\forall \boldsymbol{X} \in N(\boldsymbol{X}^*)$, where $N(\boldsymbol{X}^*)$ is a neighborhood of $\boldsymbol{X}^*$ that contains all the neighboring solutions of $\boldsymbol{X}^*$.

There is flexibility in defining the neighborhood N. For instance, we can consider the unit ball neighborhood, defined as $N(\boldsymbol{X}) = \{\boldsymbol{Y} \in \mathbb{X} : \|\boldsymbol{X} - \boldsymbol{Y}\| = 1\}$, where $\|\cdot\|$ denotes the L_1 norm (i.e., the sum of absolute differences in each coordinate). The neighborhood $N(\boldsymbol{X})$ includes all feasible

solutions that are exactly one unit away from $\boldsymbol{X}$ in one coordinate. By this definition, the local optimal set $\mathbb{X}^*$ for Problem WGP(P2) is nonempty and finite. The global optimum for Problem WGP(P2) is the solution with the smallest objective value $g(\boldsymbol{X})$ among all $\boldsymbol{X} \in \mathbb{X}$ (which may be unique or multiple).

Starting with an initial solution $\boldsymbol{X}_0$, if it is not a local optimum, we can find a better solution $\boldsymbol{X}_1$ within its neighborhood $N(\boldsymbol{X}_0)$ such that $g(\boldsymbol{X}_1) < g(\boldsymbol{X}_0)$. If $\boldsymbol{X}_1$ is not a local optimum, we can further improve it to $\boldsymbol{X}_2$ in $N(\boldsymbol{X}_1)$, and so on, until we reach a locally optimal solution $\boldsymbol{X}_n$. This iterative process is known as a "neighborhood search", and it generates a solution path $\boldsymbol{X}_0, \boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$ that leads to a local optimum from the starting point $\boldsymbol{X}_0$. We note that a solution path refers to the sequence of iteratively generated solutions in a neighborhood search, whereas a sample path represents a realization of a stochastic process used to approximate the objective function. It is evident that any non-optimal solution $\boldsymbol{X}_0$ has at least one solution path leading to a local optimum in $\mathbb{X}^*$. Additionally, for any local optimum $\boldsymbol{X}^* \in \mathbb{X}^*$, there exists at least one initial solution $\boldsymbol{X}_0 \in \mathbb{X}$ such that a solution path beginning from $\boldsymbol{X}_0$ reaches $\boldsymbol{X}^*$ through neighborhood moves. These observations enable us to utilize neighborhood search in the optimization algorithm design. In our GP-SPLINE approach, we incorporate such iterative neighborhood searches to identify multiple locally optimal solutions in $\mathbb{X}^*$ and eventually return a global optimum to Problem WGP(P2) from among them.

Given that WGP(P2) contains non-closed-form terms, we approximate it using WGP(P2(m)). The stochastic nature of WGP(P2(m)) poses challenges in achieving solution optimality and algorithm convergence. However, the discrete nature of WGP(P2) simplifies ensuring the quality of solutions and facilitates the convergence analysis. In the subsequent sections, we will discuss solution optimality in a more rigorous manner.

Assuming the feasible region $\mathbb{X}$ of Problem WGP(P2) has q solutions $\{\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_q\}$, let $g(\boldsymbol{X}; t)$ denote the objective function of WGP(P2) with respect to the chosen target values $(t_1, t_2, \ldots, t_R)$ for these R objectives. Thus, we have q true objective function values $\{g(\boldsymbol{X}_1; t), g(\boldsymbol{X}_2; t), \ldots, g(\boldsymbol{X}_q; t)\}$ corresponding to the solutions. For further discussion, we consider the largest subset $\mathbb{X}' \subseteq \mathbb{X}$ that contains solutions $\{\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_{q'}\}$ with distinct objective function values, i.e., $g(\boldsymbol{X}_1) \neq g(\boldsymbol{X}_2) \neq \ldots \neq g(\boldsymbol{X}_{q'})$. Let $\delta_{\min} = \min_{\boldsymbol{X}_i, \boldsymbol{X}_j \in \mathbb{X}'} |g(\boldsymbol{X}_i; t) - g(\boldsymbol{X}_j; t)| > 0$. We will now discuss how the approximation of WGP(P2) with WGP(P2(m)) improves as the sample size m increases. We say

that the sample-path problem WGP(P2(m)) is *efficient* with respect to a sample size m if the local optimal solution set $\mathbb{X}^*(m) \subseteq \mathbb{X}^*$. The following proposition shows that WGP(P2(m)) is efficient with probability one if the sample size m is greater than or equal to a sufficiently large threshold M.

Proposition 1. Consider the problems WGP(P2) and WGP(P2(m)) under study. If $\widehat{Z}_{r,m}(\boldsymbol{X}) \to Z_r(\boldsymbol{X})$ almost surely as the sample size $m \to \infty$ for all $r \in \{1, \ldots, R\}$ and every $\boldsymbol{X} \in \mathbb{X}$, then there exists a sufficiently large sample size M such that $\Pr\{\mathbb{X}^*(m) \subseteq \mathbb{X}^*\} = 1$. Here, $\mathbb{X}^*(m)$ represents the local optimal set of WGP(P2(m)) and $\mathbb{X}^*$ represents the local optimal set of WGP(P2).

Proof: Assume that $\boldsymbol{X} \in \mathbb{X}$ but $\boldsymbol{X} \notin \mathbb{X}^*$. Our goal is to show that $\Pr\{\boldsymbol{X} \notin \mathbb{X}^*(m)\} \to 1$ as $m \to \infty$. Since $\boldsymbol{X} \notin \mathbb{X}^*$, there exists at least one $\boldsymbol{X'} \in N(\boldsymbol{X})$ such that $g(\boldsymbol{X'}) < g(\boldsymbol{X})$, and moreover, $g(\boldsymbol{X}) - g(\boldsymbol{X'}) \geq \delta_{\min} > 0$, where $\delta_{\min}$ is the minimum positive difference between distinct objective function values. Since $\widehat{Z}_{r,m}(\boldsymbol{X}) \to Z_r(\boldsymbol{X})$ almost surely for all $\boldsymbol{X} \in \mathbb{X}$ and all r, it follows that $\widehat{g}_m(\boldsymbol{X}) \to g(\boldsymbol{X})$ almost surely as $m \to \infty$. Therefore, for any $\epsilon > 0$, there exists an $M_1 > 0$ such that for $m \geq M_1$,

$$|\widehat{g}_m(\boldsymbol{X}) - g(\boldsymbol{X})| < \epsilon,$$

almost surely. Similarly, there exists an $M_2 > 0$ such that for $m \geq M_2$,

$$|\widehat{g}_m(\boldsymbol{X'}) - g(\boldsymbol{X'})| < \epsilon,$$

almost surely. Let us choose $\epsilon = \frac{\delta_{\min}}{3}$ and $M = \max\{M_1, M_2\}$. Then, for $m \geq M$, we have almost surely $\widehat{g}_m(\boldsymbol{X}) > g(\boldsymbol{X}) - \epsilon > g(\boldsymbol{X'}) + \delta_{\min} - \epsilon = g(\boldsymbol{X'}) + \frac{2}{3}\delta_{\min} > \widehat{g}_m(\boldsymbol{X'}) + \epsilon > \widehat{g}_m(\boldsymbol{X'})$, since $\widehat{g}_m(\boldsymbol{X'}) < g(\boldsymbol{X'}) + \epsilon$. Therefore, $\widehat{g}_m(\boldsymbol{X}) > \widehat{g}_m(\boldsymbol{X'})$, almost surely, implying that $\boldsymbol{X}$ is not a local optimum in WGP(P2(m)). Thus, $\Pr\{\boldsymbol{X} \notin \mathbb{X}^*(m)\} = 1$ for $m \geq M$. Since this holds for any $\boldsymbol{X} \notin \mathbb{X}^*$, we conclude that $\mathbb{X}^*(m) \subseteq \mathbb{X}^*$ almost surely for $m \geq M$. □

Assuming that the GP-SPLINE algorithm runs an infinite number of outer iterations with independent initial solutions, the accuracy of solving the sample-path problems in each iteration will improve with increasing outer-iteration counter h and computing budget b_h, due to decreased sampling errors. As h approaches infinity, the sample sizes m_k used in the inner iterations also increase, leading to more accurate estimates of the objective function values. Consequently, GP-SPLINE will

evaluate all finite candidate solutions infinitely many times with increasing accuracy. Therefore, as h tends to infinity, the best candidate solution $\boldsymbol{X}_h^*$, or the incumbent solution, will converge to the global optimum with probability one.

Next we discuss the solution quality in terms of the approximation of objective function values. When WGP-SPLINE returns $\boldsymbol{X}_h^*$ with a large enough sample size m (as described in Proposition 1), the estimated objective value $\widehat{g}_m(\boldsymbol{X}_h^*)$ will be close to the true optimal value $g(\boldsymbol{X}_h^*)$ with high probability.

The sample average approximation $\widehat{g}_m(\boldsymbol{X}) = (1/m)\sum_{i=1}^m \widehat{g}(\boldsymbol{X};\xi_i)$ in WGP(P2(m)) is based on a sample of m independent and identically distributed (i.i.d.) realizations of the simulation output, where ξ_i, $i = 1, 2, \ldots, m$, represent the random inputs or seeds in the simulation. If the random variables $\widehat{g}(\boldsymbol{X};\xi_i)$ are sub-Gaussian with variance proxy σ^2 (which includes Gaussian or bounded random variables; see, e.g., Buldygin and Kozachenko (1980)), then the estimation error can be bounded using Hoeffding's inequality (Hoeffding, 1963). Specifically, the following proposition holds:

Proposition 2. Assuming the simulation oracle in WGP(P2(m)) generates i.i.d. sub-Gaussian realizations $\widehat{g}(\boldsymbol{X};\xi)$ with variance proxy σ^2 and provides a consistent estimate $\mathrm{E}[\widehat{g}_m(\boldsymbol{X})] = g(\boldsymbol{X})$ for every $\boldsymbol{X} \in \mathbb{X}$, then for any $\epsilon > 0$, $\Pr\{|\widehat{g}_m(\boldsymbol{X}) - g(\boldsymbol{X})| \geq \epsilon\} \leq 2e^{-\frac{m\epsilon^2}{2\sigma^2}}$.

Note that in Proposition 2, consistent estimation is assumed, and the finite domain of Problem WGP(P2) ensures uniform convergence. This proposition provides a guideline for choosing the sample size m to achieve a desired accuracy ϵ in estimating $g(\boldsymbol{X})$. Specifically, to ensure that $\Pr\{|\widehat{g}_m(\boldsymbol{X}) - g(\boldsymbol{X})| \geq \epsilon\} \leq \alpha$, where $\alpha > 0$ is the significance level (so the confidence level is $1 - \alpha$), we need $2\exp(-\frac{m\epsilon^2}{2\sigma^2}) \leq \alpha$, which simplifies to $m \geq \frac{2\sigma^2}{\epsilon^2}\ln(\frac{2}{\alpha})$. The assumption that the random variables $\widehat{g}(\boldsymbol{X};\xi)$ are sub-Gaussian with variance proxy σ^2 independent of $\boldsymbol{X}$ is strong. However, we can consider $\sigma^2 = \sup_{\boldsymbol{X}} \sigma^2(\boldsymbol{X})$ to account for the worst-case variance over all $\boldsymbol{X}$. This inequality ensures that the probability of the sampling error exceeding ϵ is no more than α, meaning the estimate $\widehat{g}_m(\boldsymbol{X})$ is within ϵ of the true value $g(\boldsymbol{X})$ with probability at least $1 - \alpha$.

The following is an example. Assume $\sigma^2 = 1$, $\epsilon = 0.01$, and $\alpha = 0.05$. Then, we have: $m \geq \frac{2\times 1}{(0.01)^2}\ln(\frac{2}{0.05}) \approx 73,778$. Therefore, a sample size of approximately $73,778$ is required to ensure that the estimation error $|\widehat{g}_m(\boldsymbol{X}) - g(\boldsymbol{X})|$ does not exceed 0.01 with at least 95% confidence.

If the problem has a relatively large $\delta_{\min}$ (the minimum difference between distinct objective function values), then choosing $\epsilon \leq \delta_{\min}$ ensures that the sampling error does not obscure the distinctions between different solutions. This allows for accurate identification of the optimal solution without necessitating excessively large sample sizes.

As $\delta_{\min}$ is usually unknown for the studied WGP(P2), we may select a convergent sequence δ_k that approaches zero as $k \to \infty$. This ensures that for some K, $\delta_k \leq \delta_{\min}$ for all $k \geq K$. In the inner iteration k of WGP-SPLINE (see Table 1), we let $\epsilon_k = \delta_k$, for $k = 1, 2, \ldots$. Then, under the conditions of Proposition 2, we have $\Pr\{|\widehat{g}_{m_k}(\boldsymbol{X}) - g(\boldsymbol{X})| \geq \epsilon_k\} \leq 2\exp(-\frac{m_k\epsilon_k^2}{2\sigma^2})$. If we choose the sample size sequence $\{m_k\}$ and the sampling error sequence $\{\epsilon_k\}$ appropriately, then asymptotically the probability of the simulation error exceeding ϵ_k goes to zero. The formal statement of this desired result is described in Proposition 3 as follows.

Proposition 3. Assuming the simulation oracle in WGP(P2(m)) generates i.i.d. sub-Gaussian realizations $\widehat{g}(\boldsymbol{X};\xi)$ with variance proxy $\sigma^2(\boldsymbol{X}) > 0$ and provides a consistent estimate $\mathrm{E}[\widehat{g}_{m_k}(\boldsymbol{X})] = g(\boldsymbol{X})$ for every $\boldsymbol{X} \in \mathbb{X}$. If in the WGP-SPLINE algorithm, the sample sizes m_k and sampling errors ϵ_k are chosen such that $\lim_{k\to\infty} \frac{m_k\epsilon_k^2}{\ln k} = \infty$, then for any $\boldsymbol{X} \in \mathbb{X}$: $\Pr\{|\widehat{g}_{m_k}(\boldsymbol{X}) - g(\boldsymbol{X})| \geq \epsilon_k \ i.o.\} = 0$. That is, $|\widehat{g}_{m_k}(\boldsymbol{X}) - g(\boldsymbol{X})| < \epsilon_k$ eventually almost surely, and when $\epsilon_k \to 0$ as $k \to \infty$, the sample-path problem WGP(P2(m)) converges to the original problem WGP(P2) with probability one.

Proof: Let $\sigma^2 = \sup_{\boldsymbol{X}} \sigma^2(\boldsymbol{X})$. From Proposition 2, we have $\Pr\{|\widehat{g}_{m_k}(\boldsymbol{X}) - g(\boldsymbol{X})| \geq \epsilon_k\} \leq 2\exp(-\frac{m_k\epsilon_k^2}{2\sigma^2})$. Let $p_k = 2\exp(-\frac{m_k\epsilon_k^2}{2\sigma^2})$. To apply the Borel-Cantelli lemma (Billingsley, 1995), we consider the convergence of the series $\sum_{k=1}^{\infty} p_k$. The series converges if p_k decreases rapidly enough. If we choose m_k and ϵ_k such that $\lim_{k\to\infty} \frac{m_k\epsilon_k^2}{\ln k} = \infty$, then for sufficiently large k, the exponent $\frac{m_k\epsilon_k^2}{2\sigma^2}$ exceeds any multiple of $\ln k$. Specifically, for any $\delta > 0$, there exists K such that for all $k \geq K$, $\frac{m_k\epsilon_k^2}{2\sigma^2} \geq (1+\delta)\ln k$. Thus, $p_k = 2\exp(-\frac{m_k\epsilon_k^2}{2\sigma^2}) \leq 2\exp(-(1+\delta)\ln k) = \frac{2}{k^{1+\delta}}$. Since $\delta > 0$, the series $\sum_{k=1}^{\infty} \frac{1}{k^{1+\delta}}$ converges. Therefore, $\sum_{k=1}^{\infty} p_k$ converges. By the Borel-Cantelli lemma, the probability that the events $\{|\widehat{g}_{m_k}(\boldsymbol{X}) - g(\boldsymbol{X})| \geq \epsilon_k\}$ occur infinitely often is zero. Hence, $|\widehat{g}_{m_k}(\boldsymbol{X}) - g(\boldsymbol{X})| < \epsilon_k$ eventually almost surely. Since $\epsilon_k \to 0$ as $k \to \infty$, this implies that the sample-path problem WGP(P2(m)) converges to the original problem WGP(P2) with probability one. □

In this proposition, we establish conditions under which the estimation error $|\widehat{g}_{m_k}(\boldsymbol{X}) - g(\boldsymbol{X})|$

becomes arbitrarily small as k increases. Specifically, we choose the sequences $\{m_k\}$ and $\{\epsilon_k\}$ such that $m_k\epsilon_k^2$ grows faster than $\ln k$. Similar results can be obtained using other convergence tests, such as the comparison test. Suppose we ensure that $\exp(-\frac{m_k\epsilon_k^2}{2\sigma^2}) \leq \exp(-\beta \ln k)$, where $\beta > 1$. Solving for m_k, we obtain: $m_k \geq 2\sigma^2\beta \ln k/\epsilon_k^2$. Since $\sum_{k=1}^{\infty} \exp(-\beta \ln k) = \sum_{k=1}^{\infty} k^{-\beta}$ converges when $\beta > 1$, it follows that $\sum_{k=1}^{\infty} \exp(-\frac{m_k\epsilon_k^2}{2\sigma^2}) < \infty$. Therefore, the conditions of Proposition 3 are satisfied. We can set the decreasing sequence $\epsilon_k \to 0$ in various ways. Let's choose $\epsilon_k = \frac{1}{\sqrt{k}}$; then the condition above becomes: $m_k \geq 2\sigma^2\beta k \ln k$. This result provides a practical guideline for WGP-SPLINE: if the sample sizes m_k grow at least as fast as the order of $k \ln k$ and $\epsilon_k = \frac{1}{\sqrt{k}}$, then the algorithm solves the true problem WGP(P2) asymptotically with probability one. In implementations, one might choose $m_k = \lceil Ck \ln k \rceil$ for some constant C determined by computational resources. In practice, σ^2 can be estimated from preliminary simulations. Using an upper bound ensures the validity of the probability bounds. By carefully selecting and adjusting the sequences $\{m_k\}$ and $\{\epsilon_k\}$, we can ensure that the WGP-SPLINE algorithm converges effectively while managing computational demands.

3.4 The Convergence Analysis of MGP-SPLINE Algorithm

The convergence analysis and propositions developed for WGP-SPLINE can be applied to MGP-SPLINE with adjustments to account for the presence of the max operator in the objective function, which affects the estimation error bounds and convergence conditions. The following Proposition 4 is a modified version of Proposition 1, where WGP is replaced by MGP.

Proposition 4. Consider the problems MGP(P2) and MGP(P2(m)) under study. If $\widehat{Z}_{r,m}(\boldsymbol{X}) \to Z_r(\boldsymbol{X})$ almost surely as the sample size $m \to \infty$ for all $r \in \{1, \ldots, R\}$ and every $\boldsymbol{X} \in \mathbb{X}$, then there exists a sufficiently large sample size M such that $\Pr\{\mathbb{X}^*(m) \subseteq \mathbb{X}^*\} = 1$. Here, $\mathbb{X}^*(m)$ represents the local optimal set of MGP(P2(m)) and $\mathbb{X}^*$ represents the local optimal set of MGP(P2).

Proof: Assume that $\boldsymbol{X} \in \mathbb{X}$ but $\boldsymbol{X} \notin \mathbb{X}^*$. Since $\boldsymbol{X} \notin \mathbb{X}^*$, there exists $\boldsymbol{X}' \in N(\boldsymbol{X})$ such that $g(\boldsymbol{X}') < g(\boldsymbol{X})$, and $\Delta = g(\boldsymbol{X}) - g(\boldsymbol{X}') \geq \delta_{\min} > 0$, where $\delta_{\min}$ is the minimum positive difference between distinct objective function values over $\mathbb{X}$. Let $g(\boldsymbol{X}) = \max_r G_r(\boldsymbol{X})$, where

$$G_r(\boldsymbol{X}) = w_r^- \times \max\{t_r - Z_r(\boldsymbol{X}), 0\} + w_r^+ \times \max\{Z_r(\boldsymbol{X}) - t_r, 0\}.$$

Similarly, define the estimated objective function $\widehat{g}_m(\boldsymbol{X}) = \max_r \widehat{G}_r(\boldsymbol{X})$, where

$$\widehat{G}_r(\boldsymbol{X}) = w_r^- \times \max\{t_r - \widehat{Z}_{r,m}(\boldsymbol{X}), 0\} + w_r^+ \times \max\{\widehat{Z}_{r,m}(\boldsymbol{X}) - t_r, 0\}.$$

Since $\widehat{Z}_{r,m}(\boldsymbol{X}) \to Z_r(\boldsymbol{X})$ almost surely for all r and $\boldsymbol{X}$, for any $\epsilon > 0$, there exists M such that for all $m \geq M$, we have almost surely:

$$|\widehat{Z}_{r,m}(\boldsymbol{X}) - Z_r(\boldsymbol{X})| < \epsilon, \quad \forall r, \boldsymbol{X} \in \{\boldsymbol{X}, \boldsymbol{X}'\}.$$

The function $G_r(\boldsymbol{X})$ is Lipschitz continuous in $Z_r(\boldsymbol{X})$ with Lipschitz constant: $L_r = \max\{w_r^-, w_r^+\}$. Therefore, the estimation error in $G_r(\boldsymbol{X})$ is bounded by:

$$\Delta_r(\boldsymbol{X}) = |G_r(\boldsymbol{X}) - \widehat{G}_r(\boldsymbol{X})| \leq L_r |Z_r(\boldsymbol{X}) - \widehat{Z}_{r,m}(\boldsymbol{X})| \leq L_r \epsilon.$$

Similarly for $\boldsymbol{X}'$. Let $L = \max_r L_r$; then $\Delta_r(\boldsymbol{X}), \Delta_r(\boldsymbol{X}') \leq L\epsilon$, $\forall r$. We now address the max operator over r. Since $g(\boldsymbol{X}) = \max_r G_r(\boldsymbol{X})$, there exists an index r^* such that $g(\boldsymbol{X}) = G_{r^*}(\boldsymbol{X})$. Similarly, there exists r'^* for $\boldsymbol{X}'$. We consider two cases:

Case 1: $r^* = r'^*$. The true difference is $\Delta = G_{r^*}(\boldsymbol{X}) - G_{r^*}(\boldsymbol{X}') \geq \delta_{\min} > 0$. The estimated difference satisfies: $\widehat{G}_{r^*}(\boldsymbol{X}) - \widehat{G}_{r^*}(\boldsymbol{X}') \geq \Delta - 2L\epsilon$. Choose $\epsilon = \frac{\delta_{\min}}{4L}$, so that $2L\epsilon = \frac{\delta_{\min}}{2}$, leading to:

$$\widehat{G}_{r^*}(\boldsymbol{X}) - \widehat{G}_{r^*}(\boldsymbol{X}') \geq \delta_{\min} - \frac{\delta_{\min}}{2} = \frac{\delta_{\min}}{2} > 0.$$

Since $\widehat{g}_m(\boldsymbol{X}) \geq \widehat{G}_{r^*}(\boldsymbol{X})$ and $\widehat{g}_m(\boldsymbol{X}') \leq \widehat{G}_{r^*}(\boldsymbol{X}') + L\epsilon$, we have:

$$\widehat{g}_m(\boldsymbol{X}) - \widehat{g}_m(\boldsymbol{X}') \geq \frac{\delta_{\min}}{2} - L\epsilon = \frac{\delta_{\min}}{4} > 0.$$

Case 2: $r^* \neq r'^*$. The true difference is $\Delta = G_{r^*}(\boldsymbol{X}) - G_{r'^*}(\boldsymbol{X}') \geq \delta_{\min} > 0$. The estimated difference satisfies:

$$\widehat{G}_{r^*}(\boldsymbol{X}) - \widehat{G}_{r'^*}(\boldsymbol{X}') \geq \Delta - 2L\epsilon.$$

Again, choosing $\epsilon = \frac{\delta_{\min}}{4L}$, we obtain:

$$\widehat{G}_{r^*}(\boldsymbol{X}) - \widehat{G}_{r'^*}(\boldsymbol{X}') \geq \frac{\delta_{\min}}{2} > 0.$$

For $\widehat{g}_m(\boldsymbol{X})$, estimation errors in other $\widehat{G}_r(\boldsymbol{X})$ cannot decrease its value, as we are taking the maximum. Similarly, for $\widehat{g}_m(\boldsymbol{X}')$, other $\widehat{G}_r(\boldsymbol{X}')$ cannot increase its value beyond $\widehat{G}_{r'^*}(\boldsymbol{X}') + L\epsilon$. Therefore, the estimated ordering is preserved: $\widehat{g}_m(\boldsymbol{X}) > \widehat{g}_m(\boldsymbol{X}')$. Thus, $\boldsymbol{X} \notin \mathbb{X}^*(m)$ almost surely for $m \geq M$. Since this holds for any $\boldsymbol{X} \notin \mathbb{X}^*$, and $\mathbb{X}$ is finite, we conclude that $\Pr\{\mathbb{X}^*(m) \subseteq \mathbb{X}^*\} = 1$ for $m \geq M$. □

By carefully addressing the maximum over r and ensuring that the estimation errors are uniformly small across all objectives, we have adjusted the proof to fully account for the complexities introduced in MGP(P2), effectively managing the uniform convergence required for the proof.

The following Proposition 5 is a modified version of Proposition 2, where WGP is replaced by MGP. Also note there is a difference: in Proposition 2, the probability bound includes the term σ^2, which represents the variance proxy for $\widehat{g}(\boldsymbol{X};\xi)$ generated by WGP(P2(m)); by contrast, in Proposition 5, the probability bound includes the term σ_r^2, which represents the variance proxy for $\widehat{Z}_r(\boldsymbol{X};\xi)$ generated by MGP(P2(m)).

Proposition 5. Assuming the simulation oracle in MGP(P2(m)) generates i.i.d. sub-Gaussian realizations $\widehat{Z}_r(\boldsymbol{X};\xi)$ with variance proxy σ_r^2 for each $r \in \{1,\ldots,R\}$ and provides consistent estimate $\mathrm{E}[\widehat{Z}_{r,m}(\boldsymbol{X})] = Z_r(\boldsymbol{X})$ for every $\boldsymbol{X} \in \mathbb{X}$ and r, then for any $\epsilon > 0$, $\Pr\{|\widehat{g}_m(\boldsymbol{X}) - g(\boldsymbol{X})| \geq \epsilon\} \leq 2R\exp\left(-\frac{m\epsilon^2}{2L^2\sigma_{\max}^2}\right)$, where $\sigma_{\max}^2 = \max_r \sigma_r^2$ and $L = \max_r L_r$ with $L_r = \max\{w_r^-, w_r^+\}$.

Proof:

Step 1: Decompose the Estimation Error

We have: $|\widehat{g}_m(\boldsymbol{X}) - g(\boldsymbol{X})| = |\max_r \widehat{G}_r(\boldsymbol{X}) - \max_r G_r(\boldsymbol{X})|$.

Step 2: Use the Inequality for the Difference of Maxima

For any finite set of real numbers $\{a_r\}$ and $\{b_r\}$, the following inequality holds:

$$|\max_r a_r - \max_r b_r| \leq \max_r |a_r - b_r|.$$

Applying this to our case:

$$|\widehat{g}_m(\boldsymbol{X}) - g(\boldsymbol{X})| \leq \max_r |\widehat{G}_r(\boldsymbol{X}) - G_r(\boldsymbol{X})|.$$

Step 3: Bound the Estimation Error for Each $G_r(\boldsymbol{X})$

Since $G_r(\boldsymbol{X})$ is Lipschitz continuous in $Z_r(\boldsymbol{X})$ with Lipschitz constant L_r, we have:

$$|\widehat{G}_r(\boldsymbol{X}) - G_r(\boldsymbol{X})| \leq L_r |\widehat{Z}_{r,m}(\boldsymbol{X}) - Z_r(\boldsymbol{X})|.$$

Step 4: Apply the Sub-Gaussian Tail Bound to Each $\widehat{Z}_{r,m}(\boldsymbol{X})$

Since $\widehat{Z}_{r,m}(\boldsymbol{X})$ is the sample mean of m i.i.d. sub-Gaussian random variables with variance proxy σ_r^2, the estimation error $|\widehat{Z}_{r,m}(\boldsymbol{X}) - Z_r(\boldsymbol{X})|$ satisfies:

$$\Pr\{|\widehat{Z}_{r,m}(\boldsymbol{X}) - Z_r(\boldsymbol{X})| \geq \delta\} \leq 2\exp\left(-\frac{m\delta^2}{2\sigma_r^2}\right).$$

Step 5: Relate the Estimation Error of $\widehat{g}_m(\boldsymbol{X})$ to that of $\widehat{Z}_{r,m}(\boldsymbol{X})$

From Step 3, we have: $|\widehat{G}_r(\boldsymbol{X}) - G_r(\boldsymbol{X})| \leq L_r |\widehat{Z}_{r,m}(\boldsymbol{X}) - Z_r(\boldsymbol{X})|$. Thus, for any $\epsilon > 0$, we have:

$$\Pr\{|\widehat{G}_r(\boldsymbol{X}) - G_r(\boldsymbol{X})| \geq L_r\delta\} \leq 2\exp\left(-\frac{m\delta^2}{2\sigma_r^2}\right).$$

Let $\delta = \frac{\epsilon}{L_r}$, so that:

$$\Pr\{|\widehat{G}_r(\boldsymbol{X}) - G_r(\boldsymbol{X})| \geq \epsilon\} \leq 2\exp\left(-\frac{m\epsilon^2}{2L_r^2\sigma_r^2}\right).$$

Step 6: Use Union Bound Over All r

Since we have R different $G_r(\boldsymbol{X})$, we can bound the probability that any of the estimation errors exceeds ϵ:

$$\Pr\{\max_r |\widehat{G}_r(\boldsymbol{X}) - G_r(\boldsymbol{X})| \geq \epsilon\} \leq \sum_{r=1}^{R} \Pr\{|\widehat{G}_r(\boldsymbol{X}) - G_r(\boldsymbol{X})| \geq \epsilon\}.$$

Using the bound from Step 5:

$$\Pr\{|\widehat{g}_m(\boldsymbol{X}) - g(\boldsymbol{X})| \geq \epsilon\} \leq 2\sum_{r=1}^{R} \exp\left(-\frac{m\epsilon^2}{2L_r^2\sigma_r^2}\right).$$

Step 7: Simplify the Exponential Terms

Let $\sigma^2_{\max} = \max_r \sigma_r^2$ and $L = \max_r L_r$. Then, since $L_r^2\sigma_r^2 \le L^2\sigma^2_{\max}$ for all r, we have: $\exp\left(-\frac{m\epsilon^2}{2L_r^2\sigma_r^2}\right) \le \exp\left(-\frac{m\epsilon^2}{2L^2\sigma^2_{\max}}\right)$. Thus, the bound becomes:

$$\Pr\{|\widehat{g}_m(\boldsymbol{X}) - g(\boldsymbol{X})| \ge \epsilon\} \le 2R\exp\left(-\frac{m\epsilon^2}{2L^2\sigma^2_{\max}}\right).$$

□

The proof carefully handles the complexities introduced by the max operator in the objective function by leveraging Lipschitz continuity and using the union bound over all objectives. To ensure that $\Pr\{|\widehat{g}_m(\boldsymbol{X}) - g(\boldsymbol{X})| \ge \epsilon\} \le \alpha$, we can select m such that: $2R\exp\left(-\frac{m\epsilon^2}{2L^2\sigma^2_{\max}}\right) \le \alpha$, which simplifies to $m \ge \frac{2L^2\sigma^2_{\max}}{\epsilon^2}\ln(\frac{2R}{\alpha})$. As R increases, the required sample size also increases to maintain the same confidence level and estimation accuracy.

Here is an example. Assume $\sigma^2_{\max} = \sigma^2 = 1$ (assuming all objectives have the same variance proxy), $\epsilon = 0.01$, $\alpha = 0.05$, $L = 1$, and $R = 3$. Then, we have: $m \ge \frac{2L^2\sigma^2_{\max}}{\epsilon^2}\ln(\frac{2R}{\alpha}) \approx 95,750$. Therefore, a sample size of approximately $95,750$ is required for MGP(P2) with $R = 3$ to ensure that the estimation error $|\widehat{g}_m(\boldsymbol{X}) - g(\boldsymbol{X})|$ does not exceed 0.01 with at least 95% confidence.

Remark 1. *We can use Borell's Inequality (Adler & Taylor, 2007) to achieve a tighter bound that scales logarithmically with the number of objectives. Specifically:* $\Pr\{|\widehat{g}_m(\boldsymbol{X}) - g(\boldsymbol{X})| \ge \epsilon\} \le 2\exp\left(-\frac{m(\epsilon-\mu_m)^2}{2L^2\sigma^2_{\max}}\right)$, *where* $\sigma^2_{\max} = \max_r \sigma_r^2$, $L = \max_r L_r$, *and* $\mu_m = \frac{L\sigma_{\max}}{\sqrt{m}}\sqrt{2\ln R}$ *(the derivation is omitted here). This leads to a tighter bound compared to the union bound, particularly effective when R is large, as it avoids the linear dependence on R in the exponent. Borell's Inequality states that for a Gaussian process $\{X_t\}$ with mean zero and variance at most σ^2 for all t, the following holds:* $\Pr\left(\sup_t X_t \ge \mathrm{E}\left[\sup_t X_t\right] + u\right) \le \exp\left(-\frac{u^2}{2\sigma^2}\right)$. *However, since μ_m depends on m, this forms an implicit equation requiring iterative methods or approximations to solve for m. Although Borell's Inequality is advantageous for handling a large number of objectives, in Proposition 5, we opted for standard concentration inequalities to ensure the proof remains straightforward and accessible.*

The following Proposition 6 is a modified version of Proposition 3, where WGP is replaced by MGP.

Proposition 6. Assume that in the MGP(P2) problem, the simulation oracle generates i.i.d. sub-

Gaussian realizations $\widehat{Z}_{r,m_k}(\boldsymbol{X};\xi)$ with variance proxy $\sigma_r^2 > 0$ for each objective $r \in \{1,2,\ldots,R\}$, and provides a consistent estimate $\mathrm{E}[\widehat{Z}_{r,m_k}(\boldsymbol{X})] = Z_r(\boldsymbol{X})$ for every $\boldsymbol{X} \in \mathbb{X}$. If in the MGP-SPLINE algorithm, the sample sizes m_k and sampling errors ϵ_k are chosen such that $\lim_{k\to\infty} \frac{m_k\epsilon_k^2}{\ln k} = \infty$, then for any $\boldsymbol{X} \in \mathbb{X}$: $\Pr\{|\widehat{g}_{m_k}(\boldsymbol{X}) - g(\boldsymbol{X})| \geq \epsilon_k \ i.o.\} = 0$. That is, $|\widehat{g}_{m_k}(\boldsymbol{X}) - g(\boldsymbol{X})| < \epsilon_k$ eventually almost surely, and when $\epsilon_k \to 0$ as $k \to \infty$, the sample-path problem MGP(P2(m)) converges to the original problem MGP(P2) with probability one.

Proof: From Proposition 5 for MGP(P2), the probability that the estimation error at a single solution $\boldsymbol{X}$ exceeds $\epsilon_k > 0$ is: $\Pr\{|\widehat{g}_{m_k}(\boldsymbol{X}) - g(\boldsymbol{X})| \geq \epsilon_k\} \leq 2R\exp\left(-\frac{m_k\epsilon_k^2}{2L^2\sigma^2_{\max}}\right)$, where $\sigma^2_{\max} = \max_r \sigma_r^2$ and $L = \max_r L_r$ with $L_r = \max\{w_r^-, w_r^+\}$. Let $p_k = 2R\exp\left(-\frac{m_k\epsilon_k^2}{2L^2\sigma^2_{\max}}\right)$. We are given that $\lim_{k\to\infty} \frac{m_k\epsilon_k^2}{\ln k} = \infty$. This implies that for sufficiently large k, the exponent satisfies: $\frac{m_k\epsilon_k^2}{2L^2\sigma^2_{\max}} \geq (1+\delta)\ln k$, for any $\delta > 0$. Therefore, for large k, we have $p_k \leq 2R\exp(-(1+\delta)\ln k) = \frac{2R}{k^{1+\delta}}$. Since $\delta > 0$, the series $\sum_{k=1}^{\infty} \frac{1}{k^{1+\delta}}$ converges. Therefore, $\sum_{k=1}^{\infty} p_k \leq 2R\sum_{k=1}^{\infty} \frac{1}{k^{1+\delta}} < \infty$. Since $\sum_{k=1}^{\infty} p_k < \infty$, by the Borel-Cantelli lemma, we have: $\Pr\{|\widehat{g}_{m_k}(\boldsymbol{X}) - g(\boldsymbol{X})| \geq \epsilon_k \ i.o.\} = 0$. Thus, the sample-path problem MGP(P2(m)) converges to the original problem MGP(P2) with probability one. □

The term R appears in the probability bound due to the union bound over the R objectives. However, since R is fixed and finite, it does not affect the convergence of the series. Thus, the convergence condition for MGP(P2) remains the same as that for WGP(P2).

4 ˇEmpirical Results

This section presents a numerical study to compare the performance of our WGP-SPLINE and MGP-SPLINE algorithms with another benchmark R&S approach for solving the stochastic inventory optimization problems. This benchmark approach, referred to as the ranking and selection framework (RSF), utilizes frequentist-type R&S procedures. These procedures adopt the indifference-zone (IZ) formulation, which assumes that the difference between the expected performances of the best and the second-best solutions is at least the value of the IZ parameter. This type of procedures are designed to achieve a pre-specified lower bound on the probability of correct selection. Please see Appendix A.2 for the details of RSF.

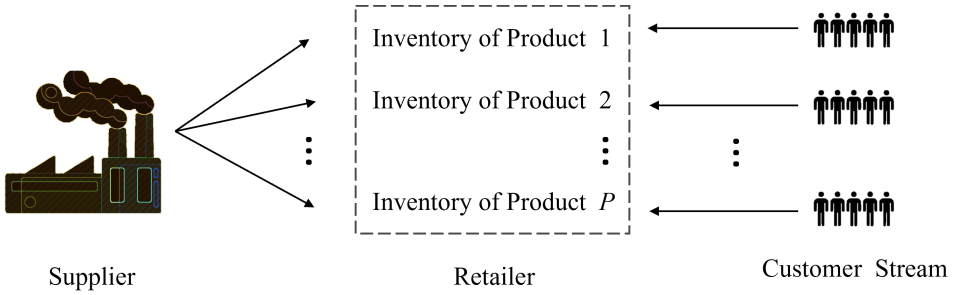


Figure 1: Inventory flow of a single-retailer inventory system with multiple products.

4.1 Inventory System Description and Objective Functions

We consider a centralized two-echelon supply chain comprising a single supplier and a single retailer that provides multiple product types to distinct customer streams, as illustrated in Figure 1. Customers arrive at the retailer with demands for various products. If the on-hand inventory cannot fully satisfy a demand, the unmet portion is backordered for future fulfillment. The inventory level is allowed to become negative to account for backorders. The demand values for different product types are assumed to be independent, a common assumption in previous studies on production inventory systems (e.g., Freeman et al., 2021). Each product type i (where $i \in \mathcal{P} = \{1, 2, \ldots, P\}$) is managed using a separate (S_i, Q_i) inventory control policy with continuous monitoring. Specifically, a replenishment order of size Q_i is placed immediately when the retailer's inventory level for product i drops to or below the reorder point S_i. We assume that the supplier always has ample stock of all items. Upon receiving a replenishment order – which may take a variable amount of time known as lead time – the backorders are filled, and the inventory level may increase. Shortages incur costs and can negatively impact the system's credibility, potentially resulting in customer loss.

In our model, the planning period length is denoted by T. For each product i, the mean inter-arrival time between customer orders is λ_i, the mean demand size per customer order is D_i, and the mean transportation (lead) time between the supplier and the retailer is L_i. The holding cost per item per unit time is H_i, the backlog cost per item per unit time is B_i, and the ordering cost per replenishment order is O_i. The decision variables are the reorder points S_i and the order quantities Q_i for each product.

Let $I_i(t)$ denote the inventory level of product i at time t. The on-hand inventory level is represented by $I_i^+(t) = \max\{0, I_i(t)\}$, and the backlog amount is $I_i^-(t) = \max\{0, -I_i(t)\}$. The

number of replenishment orders placed for product i during the period $(0, T]$ is denoted by $N_i(T)$. We consider three objective measures in our model. The first is the average inventory cost $TC(T)$, which includes the holding costs, backlog costs, and ordering costs across all product types. It can be estimated based on a single simulation replication as follows:

$$TC(T) = \frac{1}{T} \sum_{i=1}^{P} \left[H_i \cdot \int_0^T I_i^+(t)\ dt + B_i \cdot \int_0^T I_i^-(t)\ dt + O_i \cdot N_i(T) \right].$$

The second objective measure is the average inventory level on hand $AQ(T)$ during the specified period, which corresponds to the first component of $TC(T)$. It is estimated as:

$$AQ(T) = \frac{1}{T} \sum_{i=1}^{P} \left[\int_0^T I_i^+(t)\ dt \right].$$

The third performance measure is the fraction of customer demands that cannot be immediately satisfied, denoted as $FD(T)$. For each product i, we estimate $FD_i(T)$ as follows:

$$FD_i(T) = \frac{\text{Number of unmet demands for product } i \text{ in } (0, T]}{\text{Total number of demands for product } i \text{ in } (0, T]}.$$

To ensure a fair performance measure, following the suggestion by Hopp et al. (1997), we weight the $FD_i(T)$ values by the expected demand proportion for each product. The expected total demand for product i during $(0, T]$ is $\Lambda_i = D_i \cdot (T/\lambda_i)$, and the total expected demand across all products is $\Lambda = \sum_{i=1}^{P} \Lambda_i$. Using these values, we estimate the overall fraction of unsatisfied demand as:

$$FD(T) = \sum_{i=1}^{P} \left[\frac{\Lambda_i}{\Lambda} \cdot FD_i(T) \right].$$

We denote the source of system randomness by ξ, representing the sequence of independent random numbers used in simulation replications. We perform N simulation replications, with independent realizations $\xi^1, \xi^2, \ldots, \xi^N$. Let $TC(T, \xi^n)$, $AQ(T, \xi^n)$, and $FD(T, \xi^n)$ denote the observations of the total inventory cost, average inventory level, and fraction of unsatisfied demand during $(0, T]$ for the n-th replication. The performance measures of interest are defined as follows: $Z_1 = \mathrm{E}[TC(T)] = \lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} TC(T, \xi^n)$, $Z_2 = \mathrm{E}[AQ(T)] = \lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} AQ(T, \xi^n)$, and $Z_3 = \mathrm{E}[FD(T)] = \lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} FD(T, \xi^n)$. These objective measures capture the key

Table 2: Product demand, inter-arrival time, and parameter settings for Scenarios 1 and 2.

Scenario	Parameter	Product 1	Product 2	Product 3	Product 4	Product 5
1	Demand	$U(5, 35)$	$U(10, 45)$	–	–	–
	Inter-arrival time	$N(1.3, 0.5)$	$N(1.6, 0.5)$	–	–	–
	H_i	1.0	1.05	–	–	–
	B_i	1.0	1.05	–	–	–
	O_i	0.1	0.105	–	–	–
2	Demand	$U(5, 35)$	$U(10, 45)$	$U(7, 37)$	$U(9, 41)$	$U(4, 32)$
	Inter-arrival time	$N(1.3, 0.5)$	$N(1.6, 0.5)$	$N(1.4, 0.5)$	$N(1.5, 0.5)$	$N(1.2, 0.5)$
	H_i	1.0	1.05	1.02	1.04	0.9
	B_i	1.0	1.05	1.02	1.04	0.9
	O_i	0.1	0.105	0.102	0.104	0.09

performance metrics of the inventory system, enabling us to evaluate and optimize the inventory control policies. Since smaller values are preferable for all three objectives, our goal is to minimize the weighted sum of positive deviations from the target values for each objective.

4.2 Configurations and Experiment Design

To simplify our presentation, we denote the decision variables as $\boldsymbol{X} = (\boldsymbol{S}, \boldsymbol{Q})$, where $\boldsymbol{S} = \{S_i, \forall i \in \mathcal{P}\}$ and $\boldsymbol{Q} = \{Q_i, \forall i \in \mathcal{P}\}$. In our experimental study, we examine two scenarios that differ in problem scale and system variability. The replication length is set to $T = 90$ days. Scenario 1 involves two product types ($P = 2$) with relatively low output variance. For both products, we consider reorder points $S_i \in \{500, 750, 1000, 1250\}$ and order quantities $Q_i \in \{1250, 1500, \ldots, 3000, 3250\}$, resulting in a total of $(4 \times 9)^2 = 1,296$ candidate solutions. In other words, we set the bounds for the reorder points as $\boldsymbol{S}^L = 500$ and $\boldsymbol{S}^U = 1250$, and for the order quantities as $\boldsymbol{Q}^L = 1250$ and $\boldsymbol{Q}^U = 3250$ for each product type. Although discretizing (S, Q) is not required for the implementation of the proposed GP-SPLINE algorithm, we do so in our experiments to fix the candidate solutions. This allows us to evaluate the algorithm's performance and verify whether the final selected solution is indeed the most promising among the candidates. In practice, the proposed algorithm can handle continuous decision variables, allowing for an infinite solution space. To ensure a logical inventory policy, it is essential that $Q \geq S$. Otherwise, there is a risk that a shipment arrival may not restore the inventory to the reorder point, compromising the effectiveness of the replenishment strategy.

In Scenario 1, we set the target levels $\{t_1, t_2, t_3\} = (7000, 100, 0.06)$ and positive weights

$\{w_1^+, w_2^+, w_3^+\} = (1, 5, 13000)$. Since our formulation focuses on minimizing positive deviations from the target values, we set the weights for negative deviations to zero, i.e., $w_r^- = 0$, for all objectives. Under these settings, no solutions are feasible with respect to the three specified performance metrics. The best solution, with the lowest objective function value for Problem WGP(P1), is given by $(S_1, S_2) = (1250, 1250)$ and $(Q_1, Q_2) = (1750, 1500)$. The best solution, where the largest weighted deviation is minimized, for Problem MGP(P1), is given by $(S_1, S_2) = (1000, 1000)$ and $(Q_1, Q_2) = (2250, 2250)$. In Scenario 2, which involves five product classes ($P = 5$) and higher output variance, we use the same values for the reorder points and order quantities as in Scenario 1. This results in a total of $(4 \times 9)^5 = 60,466,176$ candidate solutions. The target levels are set to $\{t_1, t_2, t_3\} = (10000, 500, 0.05)$ with positive weights $\{w_1^+, w_2^+, w_3^+\} = (1, 2, 20600)$.

Due to the analytical intractability of the inventory problem, in Scenario 1 we conduct an additional 5,000 simulation replications for each candidate solution during the configuration design step to estimate their expected performance. These results are used to evaluate the algorithms' performance efficiency but are excluded from the reported sampling cost. In Scenario 2, performing such extensive simulations is infeasible due to the enormous number of candidate solutions.

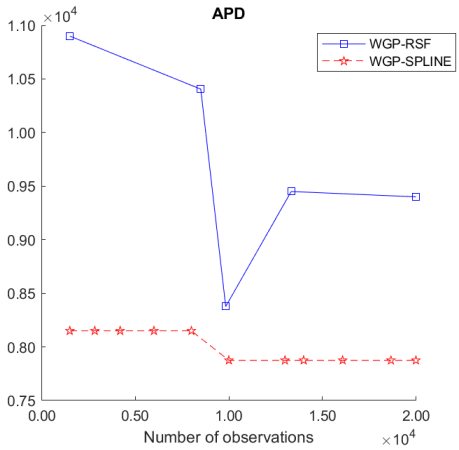
For Scenario 1, the shipping time between the supplier and the retailer follows a uniform distribution $U(3, 5)$ days, while in Scenario 2, it follows $U(1, 7)$ days. In both scenarios, the demand size for each customer order is uniformly distributed, and the inter-arrival time between customer orders is normally distributed. The specific parameters for these distributions across different products are provided in Table 2. Additionally, the travel time between the retailer and customer sites is considered negligible in both scenarios. The unit holding cost H_i, unit backlog cost B_i, and ordering cost O_i for each product type are also listed in Table 2.

For each configuration of the inventory problem with multiple performance measures, we perform 500 independent trials of the algorithms. To determine the sampling cost, we calculate the Average Number of Samples (ANS) per run, where each sample corresponds to a 90-day simulation. We also record the Average weighted sum of Positive Deviations (APD) from the three goals for the final selected solution in the WGP formulation. For the MGP formulation, we record the Average maximum weighted Positive Deviation. Additionally, for Scenario 1, which has a smaller problem scale, we estimate the Probability of Correctly identifying the Most promising solutions (PCM) – solutions with the smallest weighted sum or maximum weighted positive deviations for WGP and

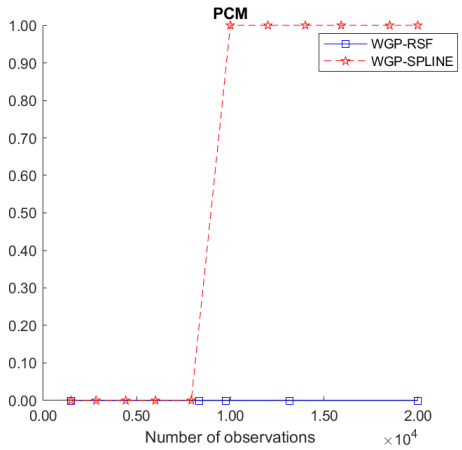
MGP, respectively. Note that there may be multiple most promising solutions. To present the experimental results clearly, we round the ANS and APD values to the nearest whole number and the PCM values to the nearest hundredth.

For both WGP-SPLINE and MGP-SPLINE algorithms, we set the parameters in Scenario 1 $\{b_h\} = \{1200 \times (1.05^h)\}$, $\{m_k\} = \{10 \times (3^{k-1})\}$, and $\{v_k\} = \{100 \times (k^{3.5})\}$. For Scenario 2, we use $\{b_h\} = \{2000 \times (1.05^h)\}$, $\{m_k\} = \{50 \times (3^{k-1})\}$, and $\{v_k\} = \{550 \times (k^{3.5})\}$. These settings ensure theoretical convergence while maintaining computational efficiency. The budget sequence $\{b_h\}$ grows at a rate of 1.05^h, allowing for a gradual increase in computational resources without excessive costs. The exponential growth of m_k ensures that $\frac{m_k \epsilon_k^2}{\ln k} \to \infty$ as $k \to \infty$, satisfying the convergence conditions outlined in Section 3. The initial values (10 and 50) were selected based on empirical testing to provide sufficient early exploration, while the growth rate (3^{k-1}) allows for precise estimation in later iterations. The simulation budget per inner iteration v_k follows a polynomial increase to balance exploration and solution refinement. These parameter settings indicate that Scenario 2 is allocated a larger sampling budget, which is reasonable since Scenario 2 presents a more difficult problem compared to Scenario 1, primarily due to its considerably larger solution space and substantially greater output variance.

Given the excessively large solution space in both scenarios, simulating all candidate solutions using the benchmark R&S approaches, WGP-RSF and MGP-RSF, is impractical. Instead, we adopt a strategy where we first determine the number of solutions to evaluate, denoted by $\mathcal{K}$, and then randomly sample $\mathcal{K}$ solutions from the solution space. We report the sampling cost, measured in ANS, for different values of $\mathcal{K}$ based on this sampling strategy. For the parameter settings of WGP-RSF, we use the following values. In Scenario 1, we set $\alpha_1 = \alpha_2 = 0.025$, $n_0 = 350$, $\delta = 114$, $\psi_1 = 463$, $\psi_2 = 58$, and $\psi_3 = 0.019$. The confidence level for correct selection is set at 95%, achieved by choosing $\alpha_1 = \alpha_2 = 0.025$, which is standard in R&S literature. The initial sample size $n_0 = 350$ was selected based on variance estimates from pilot simulations, ensuring a balance between early-stage screening efficiency and robust performance evaluation. In Scenario 1, $\delta = 114$ was set based on 5,000 additional simulation replications per candidate solution to reflect the smallest meaningful difference between solutions. The difficulty in choosing appropriate indifference-zone parameters and tolerance levels also constitutes a disadvantage of the R&S approach, which is one of the reasons we resort to the proposed algorithm. For WGP-RSF in Scenario 2, the parameters are adjusted to



(a) APD of WGP-RSF and WGP-SPLINE



(b) PCM of WGP-RSF and WGP-SPLINE

Figure 2: ˇ[Only example] Experimental results for Scenario 1 using WGP-RSF and WGP-SPLINE.

$\alpha_1 = \alpha_2 = 0.025$, $n_0 = 350$, $\delta = 21,680$, $\psi_1 = 955$, $\psi_2 = 154$, and $\psi_3 = 0.018$. In Scenario 2, the problem scale is much larger, requiring $\delta = 21,680$ to maintain a similar relative selection precision across an exponentially larger solution space. Since the MGP objective function is typically smaller than the WGP objective function, the indifference-zone parameter δ should be smaller to reflect the smaller magnitude. We set $\delta = 80$ and $\delta = 12,000$ for MGP-RSF in Scenarios 1 and 2, respectively. Since the performance measures Z_r remain the same, we can keep ψ_r unchanged unless variance estimates suggest otherwise. All other parameter values remain the same as in WGP-RSF.

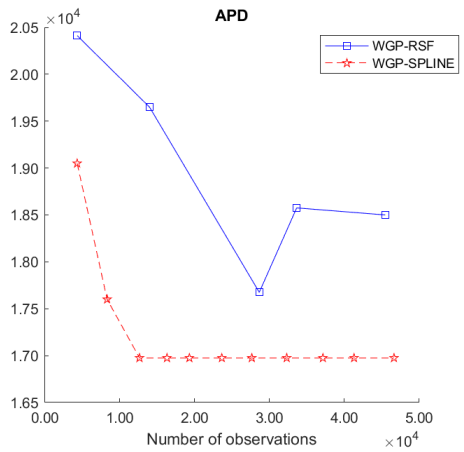
The algorithms are implemented using Matlab R2022a. The experiments are conducted on a computer equipped with an 11th Gen Intel(R) Core(TM) i7-11700@2.50GHz processor, running at 2.50 GHz with 32GB of RAM, and the Windows operating system.

4.3 Comparative Results

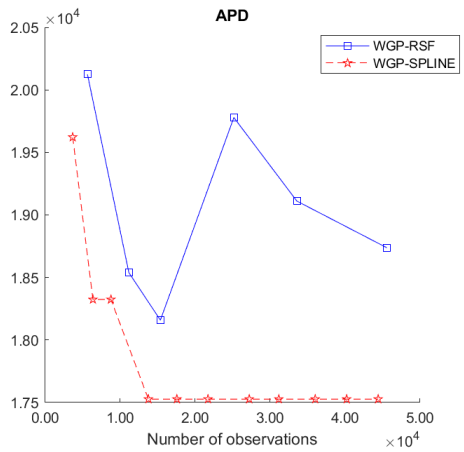
.....

5 Conclusions

In this paper, we address the challenge of solving GP problems in stochastic simulation environments. We propose two novel algorithms, WGP-SPLINE and MGP-SPLINE, designed to solve weighted goal programming (WGP) and minmax goal programming (MGP) formulations, respec-



(a) APD of WGP-RSF and WGP-SPLINE



(b) APD of WGP-RSF and WGP-SPLINE

Figure 3: ˇ[Only example] Experimental results for Scenario 2 using WGP-RSF and WGP-SPLINE (for two different trials).

Table 3: ˇExperimental results for Scenarios 1 and 2 using WGP-RSF, WGP-SPLINE, MGP-RSF, and MGP-SPLINE (across 500 trials).

	Algorithm	PCM	APD	ANS	Algorithm	PCM	APD	ANS
Scenario 1	WGP-RSF	...	...	...	MGP-RSF	...	...	...
	WGP-SPLINE	...	...	...	MGP-SPLINE	...	...	...
Scenario 2	WGP-RSF	–	...	...	MGP-RSF	–	...	...
	WGP-SPLINE	–	....	....	MGP-SPLINE	–	....	....

tively. We reformulate traditional GP problems using sample-path estimators for the objective functions, enabling direct integration of stochastic simulation outputs. This approach eliminates the need for explicit feasibility checks and tolerance levels, which are difficult to define in stochastic contexts. We extend the R-SPLINE framework to accommodate the multi-performance measure nature of GP problems and the non-differentiability introduced by max-functions. We provide rigorous convergence proofs for both algorithms, demonstrating that under appropriate conditions on the sample sizes and algorithm parameters, the algorithms converge to an optimal solution with probability one. We apply the proposed algorithms to stochastic inventory optimization problems in a multi-product, two-echelon supply chain system. The results of our empirical study indicate that the WGP-SPLINE and MGP-SPLINE algorithms are efficient tools for solving complex stochastic GP problems.

There are several valuable directions for extending or improving the proposed algorithms. For instance, variance reduction techniques could enhance the statistical efficiency of the WGP-SPLINE and MGP-SPLINE algorithm (e.g., Tsai et al., 2023). Another promising direction is to dynamically adjust the threshold values for constraints (i.e., target levels) used in the GP Problem, instead of fixing them a priori. In such cases, it would be beneficial to develop SO algorithms that reuse observations from previous feasibility checks, as demonstrated by Feng and Staum (2017). Additionally, incorporating stochastic constraints into the GP formulation could address practical applications (e.g., Tsai et al., 2024). Exploring alternative GP formulations, such as lexicographic or polynomial GP, is another potential extension.

References

Adler, R.J., & Taylor, J.E. (2007). *Random Fields and Geometry*. Springer Science & Business Media.

Billingsley, P. (1995). *Probability and Measure*, 3rd ed. New York: John Wiley & Sons.

Bravo, M., & Gonzalez, I. (2009). Applying stochastic goal programming: A case study on water use planning. *European Journal of Operational Research*, *196*(3), 1123–1129.

Buldygin, V.V., & Kozachenko, Y.V. (1980). Sub-Gaussian random variables. *Ukrainian Mathematical Journal*, *32*, 483–489.

Colapinto, C., Jayaraman, R., & Marsiglio, S. (2017). Multi-criteria decision analysis with goal programming in engineering, management and social sciences: a state-of-the art review. *Annals of Operations Research*, *251*(1-2), 7–40.

Deb, K. (2011). Multi-objective optimisation using evolutionary algorithms: An introduction. In *Multi-objective Evolutionary Optimisation for Product Design and Manufacturing* (pp. 3–34). London: Springer London.

Dutta, D., & Kumar, P. (2015). Application of fuzzy goal programming approach to multi-objective linear fractional inventory model. *International Journal of Systems Science*, *46*(12), 2269–2278.

Feng, M., & Staum, J. (2017). Green simulation: Reusing the output of repeated experiments. *ACM Transactions on Modeling and Computer Simulation*, *27*(4), 1–28.

Freeman, N.K., Narayanan, A., & Keskin, B.B. (2021). Optimal use of downward substitution in a manufacturing operation subject to uncertainty. *Omega*, *103*, Article 102372.

Hanks, R.W., Weir, J.D., & Lunday, B.J. (2017). Robust goal programming using different robustness echelons via norm-based and ellipsoidal uncertainty sets. *European Journal of Operational Research*, *262*(2), 636–646.

Hoeffding, W. (1963). Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, *301*(58), 13–30.

Hong, L.J., Fan, W., & Luo, J. (2021). Review on ranking and selection: A new perspective. *Frontiers of Engineering Management*, *8*(3), 321–343.

Hong, L.J., Nelson, B.L., & Xu, J. (2015). Discrete optimization via simulation. In: Fu, M.C. (Eds.), Chapter 2 of *Handbook of Simulation Optimization*, New York, NY: Springer.

Hopp, W.J., Spearman, M.L., & Zhang, R.Q. (1997). Easily implementable inventory control policies. *Operations Research*, *45*(3), 327–340.

Jalali, H., & Nieuwenhuyse, I.V. (2015). Simulation optimization in inventory replenishment: a classification. *IIE Transactions*, *47*(11), 1217–1235.

Lee, M.L., Park, C., & Park, D.U. (2018). Self-adjusting the tolerance level in a fully sequential feasibility check procedure. *European Journal of Operational Research*, *271*(2), 733–745.

Lu, H.C, & Tsai, S.C. (2024). Generalized robust goal programming model. *European Journal of Operational Research*, *319*(2), 638–657.

Masri, H. (2017). A multiple stochastic goal programming approach for the agent portfolio selection problem. *Annals of Operations Research*, *251*(1-2), 179–192.

Nagaraj, K., & Pasupathy, R. (2024). Stochastically constrained simulation optimization on integer-ordered spaces: The cgR-SPLINE algorithm. *Technical Report, Department of Statistics, Purude University, West Lafayette, Indiana, USA.*

Özcan, U., & Toklu, B. (2009). Multiple-criteria decision-making in two-sided assembly line balancing: A goal programming and a fuzzy goal programming models. *Computers and Operations Research*, *36*, 1955–1965.

Pardalos, P.M., & Romeijn, H.E. (2002). *Handbook of Global Optimization*, vol. 2. Kluwer Academic Publishers.

Park, C., & Kim, S.-H. (2015). Penalty function with memory for discrete optimization via simulation with stochastic constraints. *Operations Research*, *63*(5), 1195–1212.

Tsai, S.C., Chen, H., Wang, H., & Zhang, Z.G. (2021a). Simulation optimization in security screening systems subject to budget and waiting time constraints. *Naval Research Logistics*, *68*(7), 920–936.

Tsai, S.C., Luo, J., Jiang, G., & Yeh, W.C. (2023). Adaptive fully sequential selection procedures with linear and nonlinear control variates. *IISE Transactions*, *55*(6), 561–573.

Tsai, S.C., Yeh, Y., & Kuo, C.Y. (2021b). Efficient optimization algorithms for surgical scheduling under uncertainty. *European Journal of Operational Research*, *293*(2), 579–593.

Tsai, S.C., Yeh, Y., Wang, H., & Chou, T.C. (2024). Efficient optimization in stochastic production planning problems with product substitution. *Computers and Operations Research*, *164*, 106544.

Wang, H. (2013). Zigzag search for continuous multiobjective optimization. *INFORMS Journal on Computing*, *25*(4), 654–665.

Wang, H. (2017). Multi-objective retrospective optimization using stochastic zigzag search. *European Journal of Operational Research*, *263*(3), 946–960.

Wang, H., Laredo, D., Cuate, O., & Schütze, O. (2019). Enhanced directed search: A continuation method for mixed-integer multi-objective optimization problems. *Annals of Operations Research*, *279*(1-2), 343–365.

Wang, H., Pasupathy, R., & Schmeiser, B.W. (2013). Integer-ordered simulation optimization using R-SPLINE: Retrospective search with piecewise-linear interpolation and neighborhood enumeration. *ACM Transactions on Modeling and Computer Simulation*, *23*(3), 1–24.

Zabinsky, Z.B. (2015). Stochastic adaptive search methods: Theory and implementation. In: Fu, M.C. (Eds.), Chapter 11 of *Handbook of Simulation Optimization*, New York, NY: Springer.

Zografidou, E., Petridis, K., Arabatzis, G., & Kumar Dey, P. (2016). Optimal design of the renewable energy map of Greece using weighted goal-programming and data envelopment analysis. *Computers and Operations Research*, *66*, 313–326.