Highlights

Multistage Stochastic Blending of Recycled Copper Alloys with Endogenous Stopping: A Hybrid DAH-DDPG and Chance-Constrained Programming Approach

WANG CHENG, HU JUNHAN

- A multistage stochastic blending model is developed.
- DAH-DDPG is designed for hybrid feeding and stopping decisions.
- A feasibility-handling mechanism is embedded in policy learning.
- Numerical experiments verify the effectiveness of model and algorithm.

Multistage Stochastic Blending of Recycled Copper Alloys with Endogenous Stopping: A Hybrid DAH-DDPG and Chance-Constrained Programming Approach

WANG CHENG, HU JUNHAN

Zhejiang University of Technology, No. 288 Liuhe Road, Liuxia Subdistrict, Xihu District, Hangzhou, 310023, Zhejiang, China

Abstract

The multi-stage stochastic blending problem of recycled copper alloys is characterized by compositional uncertainty, stage-wise information revealed through inspection, and decision features that integrate continuous feeding actions with discrete shutdown operations. Due to the interaction of these characteristics, this problem is more appropriately formulated as a stochastic sequential decision problem with chance constraints rather than as a single static optimization model. Accordingly, this study develops a hybrid solution framework that integrates deep reinforcement learning with chance-constrained programming to address the multi-stage stochastic blending decision problem. A dual-actor hybrid deep deterministic policy gradient (DAH-DDPG) algorithm is designed at the upper level to generate stage-specific shutdown decisions and feeding strategies, coupled with a lower-level chance-constrained programming component that adjusts the batching plan within each stage to ensure probabilistic feasibility. Numerical experiments demonstrate that the proposed method achieves stable training behavior and consistent cost reductions across various problem scales and uncertainty settings. Using single-stage chance-constrained programming and linear programming as baseline models, the proposed method reduces the average total cost by approximately 18.8%, with even greater benefits under higher material uncertainty. These findings suggest that the hybrid framework provides a robust solution approach for multi-stage stochastic blending problems with endogenous stage structures and risk management requirements.

Keywords: Multistage stochastic blending, Recycled copper alloys, Stochastic sequential decision problem, DAH-DDPG, Chance-constrained programming

1. Introduction

Recycled copper alloy blending is a critical engineering decision in resource recovery and clean production. Its primary objective is to maximize scrap utilization while controlling manufacturing costs and product quality variability. In sustainable manufacturing, efficient scrap usage influences economic performance, primary resource consumption, and environmental impact. Uncertainty in scrap composition is a key factor in decision-making and a central focus of most studies on smelting proportion optimization [1, 2, 3].

This study addresses a multi-stage stochastic blending process with continuous inspection feedback. In this process, raw material composition follows specific distributions, introducing uncertainty in state transitions. This uncertainty is revealed after component inspection at each stage. The system must determine continuation or stopping timing, as well as the feeding strategy for subsequent stages. In practice, many small and medium-sized enterprises still rely on manual experience for blending decisions, often resulting in higher costs, lower interpretability, and the need for additional stages to reduce blending deviations. Extra stages increase quality variability, energy consumption, and process waste while limiting flexibility in scrap utilization [4]. Therefore, research on multi-stage smelting blending is essential. Component uncertainty has been the primary focus of most studies, with researchers employing various risk management approaches, including stochastic optimization, robust optimization, and sequential decision-making.

Prékopa [5] systematically examined chance-constrained programming (CCP) within stochastic programming, emphasizing its characterization of probabilistic feasibility. Nemirovski et al. [6] further proposed convex approximation methods for CCP. Rong et al. [7] applied fuzzy chance-constrained linear programming to optimize scrap steel blending. Another common approach to managing uncertainty in smelting is robust optimization (RO). Ben-Tal et al. [8] discussed conservative protection for uncertain parameters in RO. Yang et al. [9] applied RO to scrap blending and production scheduling under uncertain metal concentrations. Lappas et al. [10] demonstrated that stage-wise uncertainty can be effectively managed in multi-stage process

scheduling. RO has been widely applied across fields and remains a standard method for handling uncertainty [11, 12, 13].

As a multi-stage decision process, sequential decision-making (SDM) is also a critical approach in smelting blending. SDM requires the decision-maker to observe the current stage's state and select actions leading to the next stage. The classical SDM method is Bertsekas' dynamic programming (DP) [14], which models state transitions and decision processes under uncertainty and has been widely applied across domains. SDM based on DP can be flexibly adapted to diverse problems. For the objective of identifying the optimal endogenous stopping time in multi-stage smelting blending, the SDM framework provides guidance. Research on the optimal stopping problem (OSP) and stochastic shortest path problem (SSP) offers relevant modeling and solution techniques. OSP systematically addresses the modeling of terminal stage and reward structures [15]. Li and Lee [16] proposed the ΔV-learning method and extended it to uncertain decision environments. Russo et al. [17] further examined OSP solutions under short decision horizons. SSP similarly focuses on optimal endogenous stopping time. By treating stage states as path nodes and completed blending as terminal states, part of this study's problem shares characteristics with SSP. Constrained SSP studies have addressed uncertainty and resource or risk constraints through heuristic search and policy optimization frameworks [18]. Recent methods, such as Hong and Williams' LAny algorithm [19] and Schmalz and Trevizan's CG-iLAO* [20], further develop comparable strategies for constrained SSPs.

In this problem, the feeding decision is a continuous variable, while the stopping decision is discrete. Therefore, policy search must accommodate hybrid action structures [21]. Fujimoto et al. [22] proposed an actor–critic framework as a foundation for continuous action control, where multi-network collaboration extends the agent's decision capabilities. Lillicrap et al. [23] introduced the Deep Deterministic Policy Gradient (DDPG) algorithm, applying actor-critic learning to high-dimensional continuous control. Jiang et al. [24] recently integrated such methods into deep reinforcement learning through the DRLH-JCSCT framework for mobile charging in wireless rechargeable sensor networks, providing practical engineering examples. For the present study, which requires simultaneous decisions on continuation or stopping and continuous feeding decisions, a hybrid action space is essential. These frameworks provide a reliable reference for designing the algorithm proposed herein [25, 26].

In summary, this study develops a stochastic sequential decision-making

method capable of autonomously determining the optimal continuous feeding decisions strategy. The method is demonstrated using the multi-stage recycled copper smelting blending problem as a case study.

To address this problem, the multi-stage stochastic blending of recycled copper alloys is formulated as a stochastic sequential decision problem with endogenous stopping time and risk constraints. Based on DDPG, a dual-layer hybrid algorithm framework, DAH-DDPG + CCP, is proposed. The upper layer performs discrete stopping decisions and continuous feeding decisions policy search, while the lower layer adopts the feasibility enforcement mechanism to ensure stage-wise probabilistic feasibility and adjusts the blending plan.

The main contributions of this study are summarized as follows.

This study establishes a unified stochastic sequential decision formulation for the multi-stage stochastic blending of recycled copper alloys. The proposed mathematical model comprehensively characterizes core practical attributes of the smelting process, including dynamic stage evolution, real-time inspection feedback, endogenous stopping time, and probabilistic risk constraints, thereby providing a rigorous modeling foundation for stochastic blending optimization under uncertainty.

This work develops a dual-layer hybrid learning-optimization framework that integrates the proposed DAH-DDPG algorithm and chance-constrained programming. The upper reinforcement learning layer conducts policy exploration with high dimensionality for sequential feeding and terminal indicator decisions, while the lower optimization layer guarantees stage-wise probabilistic feasibility. This hybrid structure effectively unifies data-driven policy learning and rigorous constraint satisfaction, achieving a favorable balance between decision flexibility and operational reliability.

A hybrid action representation and corresponding feasibility enforcement mechanism is designed to accommodate the coexistence of discrete stopping decisions and continuous feeding decisions. By embedding multi-type action outputs within a unified learning framework, the proposed method enables integrated end-to-end decision-making for hybrid action spaces and resolves the incompatibility issue between discrete-continuous control and traditional single-mode reinforcement learning policies.

This study constructs a stochastic scenario generator to produce diversified experimental instances with varying problem scales and heterogeneous uncertainty configurations. Systematic numerical experiments validate the effectiveness and generalization capability of the proposed method. The

experimental results further clarify the applicable scenarios of multi-stage closed-loop decision mechanisms and reveal the inherent advantages of sequential corrective optimization in uncertain blending environments.

The remainder of the paper is organized as follows. Section 2 defines the problem and presents the mathematical model. Section 3 describes the hybrid algorithm framework and its key components. Section 4 presents the design of numerical experiments and analyzes the results. Section 5 concludes the study and discusses directions for future research.

2. Problem Description and Model Formulation

2.1. Problem Description

This study addresses a typical multi-stage uncertain blending optimization problem in recycled copper alloy remanufacturing, which is prevalent in metallurgical recycling and process optimization. The raw materials for alloy smelting exhibit distinct differences in component uncertainty and procurement cost, forming a hierarchical material system. Specifically, low-purity recycled copper alloy scraps feature high compositional uncertainty and low acquisition costs, while high-purity recycled copper alloys deliver relatively stable component contents with lower uncertainty yet higher material costs. In addition, standard pure metal materials (e.g., pure copper and pure zinc) possess deterministic and ultra-stable chemical compositions but incur the highest unit cost among all available materials. To balance production economy and product quality, the practical smelting process adopts a phased blending strategy: low-cost and high-uncertainty recycled scraps are prioritized for initial batch melting to reduce material expenditure, which conforms to the cost-saving principle of recycled resource utilization. Nevertheless, due to the random fluctuation of elemental components in low-grade recycled raw materials, the preliminary molten alloy often fails to satisfy the stringent compositional specifications of target copper alloys. To remedy component deviations, a secondary blending and adjustment stage is required, where high-purity recycled materials or high-precision pure metal additives are dosed quantitatively to correct elemental proportions. This multi-round blending and melting procedure constitutes a sequential multi-stage uncertain decision-making process, in which the material type and feeding quantity at each stage profoundly determine the final alloy quality and total production cost.

Taking the typical H62 brass remanufacturing scenario as an example, the target product requires copper (Cu) mass fraction of 60.5%–63.5% and zinc (Zn) mass fraction of 36.5%–39.5%, with total trace impurity content controlled below 0.3%. The alternative materials include multiple low-purity recycled copper alloys with highly fluctuating Cu and Zn contents, high-purity recycled alloy materials with slight component variation, and fixed-composition pure Cu and pure Zn raw materials. The entire production process is also constrained by the fixed furnace batch volume, which limits the total feeding mass of each smelting stage. By generalizing the practical production characteristics, this study formalizes the core research problem as follows: under the coupling constraints of raw material compositional uncertainty, limited furnace batch capacity, and target alloy quality specifications, how to optimize the multi-stage blending decision scheme. The decision variables cover the type and dosage of materials invested in each smelting stage, aiming to fully utilize low-cost recycled raw materials as much as possible, eliminate component deviations through staged supplementation of high-precision materials, and ultimately achieve qualified alloy products with the minimum total production cost. Based on the uncertain blending optimization framework, this study attempts to provide an effective multi-stage decision-making strategy for recycled copper alloy remanufacturing.

2.2. Model Formulation

Without loss of generality, this study considers that the specification of the recycled copper alloy product requires the mass fractions of N key elements to be strictly within their respective predefined control ranges, which is the core quality constraint that the final product must satisfy. Specifically, for each key element j ($j = 1, 2, \ldots, N$), its mass fraction in the final product must be between a predefined lower bound and upper bound. Let $\boldsymbol{L} = (L_1, L_2, \ldots, L_N)^T$ and $\boldsymbol{U} = (U_1, U_2, \ldots, U_N)^T$. Meanwhile, considering the physical constraint of furnace volume, the total mass of raw materials for a single smelting batch shall not exceed W, which restricts the feeding quantity of raw materials in each smelting stage.

There are M types of raw materials available for the smelting process, including recycled copper alloys with different purity levels and pure metal materials. The content of the N key elements in the M types of raw materials is described by a random matrix $\Xi = (\xi_{i,j})_{M\times N}$, where $\xi_{i,j}$ denotes the mass fraction of element j in raw material i. In this study, it is assumed that $\xi_{i,j}$ follows a normal distribution with mean $\mu_{i,j}$ and variance $\sigma_{i,j}^2$, i.e., $\xi_{i,j} \sim$

$N(\mu_{i,j}, \sigma_{i,j}^2)$. The mean values of all component contents are summarized into a mean matrix $\Theta = (\mu_{i,j})_{M\times N}$, and the variances of all component contents are summarized into a variance matrix $\Sigma = (\sigma_{i,j}^2)_{M\times N}$. Additionally, the unit price of each of the M raw materials is represented by a column vector $\boldsymbol{C} = (C_1, C_2, \ldots, C_M)^T$, where C_i denotes the unit price of raw material i. The available inventory of each raw material is represented by a column vector $\boldsymbol{I} = (I_1, I_2, \ldots, I_M)^T$, where I_i denotes the available inventory of raw material i (unit: kg), which serves as an additional constraint for the multi-stage blending decision.

In practical material characterization, elemental components include controlled major elements, harmful impurities, and filler elements with relatively flexible composition ranges. All these element types can be constrained by upper and lower bounds and are not distinguished individually in this model for simplification. To reduce problem dimensionality, unmonitored and unrestricted components are not considered in the formulation.

2.2.1. Decision process

This study focuses on a multi-stage blending decision problem, and the specific decision-making process is described as follows: In the first stage, based on the target product specifications, the compositional uncertainty of raw materials, and the price information, the weight of raw materials to be fed is determined. Considering the furnace volume constraint and the irreversibility of the feeding process—i.e., the weight of materials in the furnace cannot be reduced after feeding, and only additional feeding is allowed—it is necessary to decide whether to feed all the required weight or only a part of it. In other words, a decision is made on whether the current stage is the final feeding stage or whether subsequent stages are allowed to continue feeding. If it is the final feeding stage, the appropriate weight of raw materials is selected according to the specification requirements to produce the final product that meets the standards, and the decision-making process ends. If it is not the final stage, the weight of raw materials to be fed is determined, and the process proceeds to the next stage of decision-making. After the end of this stage, sampling testing is performed on the semi-finished product in the furnace to obtain its actual component contents.

If the previous stage does not end the decision-making process, the feeding selection is performed again based on the test results of the previous stage and the weight of the semi-finished product in the furnace, and the above analysis process is repeated. It is feasible to decide whether the current

stage is the final feeding stage or allow subsequent stages to continue adding materials.

This decision-making process can continue until the feeding is completed. The multi-stage state transition is illustrated in Figure 1. According to the decision process, a 0-1 variable $x_{n,1}$ is used to indicate whether the n-th stage is the final feeding stage during the decision-making of the n-th stage, where $x_{n,1} = 1$ means it is the final feeding stage and $x_{n,1} = 0$ means subsequent feeding stages are allowed. A vector $\boldsymbol{x}_{n,2}$ is used to represent the weight of raw materials fed during the decision-making of the n-th stage, where $n = 1, 2, \ldots$. Then, the total weight of raw materials fed in the n-th stage is $\|\boldsymbol{x}_{n,2}\|_1$, and the total weight of materials in the furnace after the n-th stage is $\sum_{i=1}^{n} \|\boldsymbol{x}_{i,2}\|_1$.Let τ denote the terminal stage of the blending process. Therefore, $\sum_{i=1}^{\tau} \|\boldsymbol{x}_{i,2}\|_1 = W$.

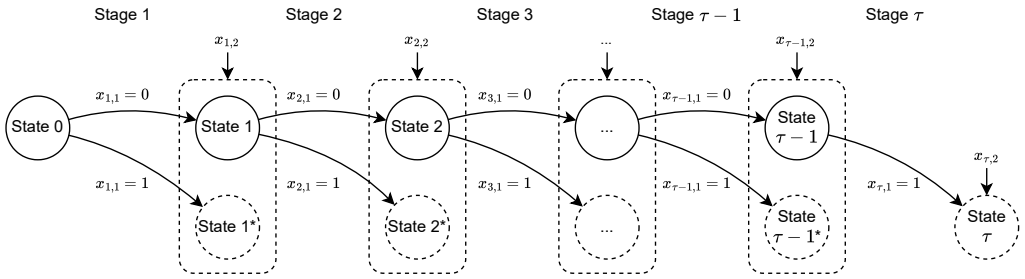


Figure 1: Multi-stage decision process: Schematic illustration of the multi-stage stochastic blending decision process for recycled copper alloys, depicting the sequential state transitions and decision evolution across consecutive operational stages.

2.2.2. State transformation

This section elaborates on the state transition and constraint analysis of the multi-stage blending decision process, which lays a foundation for the subsequent optimization model construction. At the beginning of the n-th decision stage, the available inventory level of raw materials is denoted as $\boldsymbol{I}_{n-1}$, where the initial inventory at the start of the first stage satisfies $\boldsymbol{I}_0 = \boldsymbol{I}$ (i.e., the initial available inventory vector defined previously). Let w_n represent the total alloy weight inside the furnace at the end of the n-th stage, and let the random vector $\tilde{\boldsymbol{P}}_n$ denote the elemental composition state of the molten alloy at the corresponding stage. Obviously, the initial physical state before production commencement satisfies the boundary conditions $w_0 = 0$ and $\tilde{\boldsymbol{P}}_0 = \boldsymbol{0}$, where $\boldsymbol{0}$ is an N-dimensional zero vector.

According to the practical operational rules of furnace feeding, the raw material dosage in each stage is strictly restricted by the real-time residual inventory. For the j-th type of raw material, the feeding quantity in the n-th stage, denoted as the j-th entry of decision vector $\boldsymbol{x}_{n,2}$, satisfies the non-negativity and inventory feasibility constraint:

$$0 \leq x_{n,2}^{j} \leq I_{n-1}^{j}, \quad j = 1, \ldots, M. \tag{1}$$

After the feeding operation is completed, the residual inventory is updated in a point-wise manner as $I_n^j = I_{n-1}^j - x_{n,2}^j$, and the overall inventory vector is synchronously updated as $\boldsymbol{I}_n = \boldsymbol{I}_{n-1} - \boldsymbol{x}_{n,2}$.

Correspondingly, the cumulative material weight inside the furnace is dynamically accumulated stage by stage. The total weight at the current stage equals the sum of the historical residual weight and the newly added materials, which yields

$$w_n = w_{n-1} + \|\boldsymbol{x}_{n,2}\|_1. \tag{2}$$

Owing to the inherent fluctuation of recycled raw material components, the alloy composition state is stochastic and cannot be directly determined in advance. The true composition can only be revealed through offline sampling and component detection after each melting cycle. In this multi-stage framework, the uncertainty of the composition state in the subsequent stage is only induced by the newly added raw materials in the current decision period. Without considering measurement errors in industrial inspection, the evolution law of the random composition state between adjacent periods follows the mass conservation principle, which is formulated as:

$$\tilde{\boldsymbol{P}}_n = \frac{w_{n-1}\tilde{\boldsymbol{P}}_{n-1} + \Xi \boldsymbol{x}_{n,2}}{w_{n-1} + \|\boldsymbol{x}_{n,2}\|_1}. \tag{3}$$

In the first stage, the stochastic nature of the composition state $\tilde{\boldsymbol{P}}_1$ is completely dominated by the random component matrix Ξ. Define the probability space as Ω that accommodates all possible realizations of raw material components. When the random event of the first stage is realized as $\omega_1 \in \Omega$, the uncertain matrix becomes a deterministic sample $\Xi(\omega_1)$. Accordingly, the composition state after actual detection in the first stage is calculated as:

$$\tilde{\boldsymbol{P}}_1(\omega_1) = \frac{w_0\tilde{\boldsymbol{P}}_0 + \Xi(\omega_1)\boldsymbol{x}_{1,2}}{w_0 + \|\boldsymbol{x}_{1,2}\|_1}. \tag{4}$$

By recursive analogy, suppose that real-time detection has been implemented in all previous $n-1$ stages, and the corresponding historical uncertainty realizations are recorded as the information set vector $\boldsymbol{\omega}_{n-1} = (\omega_1, \omega_2, \ldots, \omega_{n-1})$. Before the inspection operation of the n-th stage is executed, the prior stochastic composition state can be updated conditionally based on the observed historical information:

$$\tilde{\boldsymbol{P}}_n(\boldsymbol{\omega}_{n-1}) = \frac{w_{n-1}\tilde{\boldsymbol{P}}_{n-1}(\boldsymbol{\omega}_{n-1}) + \Xi \boldsymbol{x}_{n,2}}{w_{n-1} + \|\boldsymbol{x}_{n,2}\|_1}. \tag{5}$$

This prior state remains random and undetermined. After the sampling test is finished, the underlying random factor of the current stage is unveiled as ω_n. The posterior true composition state used for the next-round decision feedback is obtained as:

$$\tilde{\boldsymbol{P}}_n(\boldsymbol{\omega}_n) = \frac{w_{n-1}\tilde{\boldsymbol{P}}_{n-1}(\boldsymbol{\omega}_{n-1}) + \Xi(\omega_n) \boldsymbol{x}_{n,2}}{w_{n-1} + \|\boldsymbol{x}_{n,2}\|_1}. \tag{6}$$

Let $\boldsymbol{p}_n = (p_{n,1}, p_{n,2}, \ldots, p_{n,N})^T$ denote the test result after the end of the n-th stage, which is also an observed value of the random vector $\tilde{\boldsymbol{P}}_n$, satisfying $\boldsymbol{p}_n = \tilde{\boldsymbol{P}}_n(\boldsymbol{\omega}_n)$. According to the production quality requirements, the component proportions of the alloy at the final stage (denoted as stage τ) must strictly meet the predefined upper and lower bounds of element contents. Specifically, for each key element j ($j = 1, 2, \ldots, N$), the observed mass fraction $p_{\tau,j}$ (the j-th element of vector $\boldsymbol{p}_\tau$) must satisfy $L_j \leq p_{\tau,j} \leq U_j$, which constitutes the core quality constraint for the final product.

Accordingly, given that $\mathcal{S}$ denotes the state space, the state $\boldsymbol{s}_n$ of the n-th stage can be formulated as

$$\boldsymbol{s}_n = (\boldsymbol{p}_n, w_n, \boldsymbol{I}_n), \tag{7}$$

where it represents the component proportion and total weight at the end of the current stage, as well as the inventory of each raw material.

For the investigated multi-stage smelting blending decision process, the intermediate melt state is assumed to be fully observable after each stage. The measured elemental composition and cumulative melt weight furnish complete information for subsequent feeding and continuation or stopping decisions. Under this observation mechanism, the decision at each stage depends solely on the state observed at the end of the previous stage, independent of the unobserved composition of unfed raw materials.

2.2.3. Cost Function and Optimization Model

This study aims to minimize the expected total cost of the multi-stage smelting blending process. The total cost consists of four components: raw material cost, processing cost, stage fixed cost, and terminal composition deviation penalty.

The raw material cost incurred at stage n is formulated as

$$c_r^n = \boldsymbol{C}^T \boldsymbol{x}_{n,2}. \tag{8}$$

The processing-related cost at stage n is defined as

$$c_e^n = C_e \|\boldsymbol{x}_{n,2}\|_1 + C_e^* w_{n-1}. \tag{9}$$

Here, C_e represents the unit processing cost of newly added raw materials, and C_e^* denotes the unit holding cost of the intermediate melt.

Each operational stage incurs a fixed cost, expressed as

$$c_f^n = C_f, \tag{10}$$

where C_f is the fixed cost per stage.

Final product quality is evaluated exclusively at the terminal stage. Any deviation of elemental mass fractions from the specified ranges in the terminal state induces a penalty cost. Let $(\cdot)_+$ denote the component-wise positive operator. The terminal composition deviation penalty is formulated as:

$$\tilde{c}_v^\tau = C_v \left\| \left(\tilde{\boldsymbol{P}}_\tau - \boldsymbol{U} \right)_+ + \left(\boldsymbol{L} - \tilde{\boldsymbol{P}}_\tau \right)_+ \right\|_1. \tag{11}$$

Here, C_v represents the penalty coefficient for elemental composition deviation.

Let Π denote the set of feasible stationary policies, where each policy maps observable system states to feasible stage-wise decisions. For any policy $\pi \in \Pi$, the multi-stage stochastic smelting blending optimization problem is established as:

$$\min_{\pi \in \Pi} \mathbb{E}^\pi \left[\sum_{n=1}^{\tau} \left(c_r^n + c_e^n + c_f^n \right) + \tilde{c}_v^\tau \right]. \tag{12}$$

The expectation is taken with respect to the stochastic realizations of raw material compositions and policy-induced state transitions. The introduced

composition deviation penalty converts rigid product composition requirements into soft constraints. This formulation guarantees model validity for all reachable system states and simplifies the design of subsequent solution algorithms.

Key notations defined in this section are summarized in Table 1.

Table 1: Key Notations

Symbol	Meaning
M	Number of available raw materials, indexed by j
N	Number of considered elemental components, indexed by i
W	Target product weight of each smelting batch
T	Maximum allowable operational stage number
τ	Actual terminal stage of the multi-stage blending process
$\boldsymbol{L}$	Lower bound vector of product elemental mass fraction specifications
$\boldsymbol{U}$	Upper bound vector of product elemental mass fraction specifications
Ξ	Stochastic raw material composition matrix
Θ	Mean matrix of raw material composition parameters, with entries μ_{ij}
Σ	Variance matrix of raw material composition parameters, with entries σ_{ij}^2
ω_n	Stochastic noise induced by uncertain raw material compositions at stage n
$\tilde{\boldsymbol{P}}_n$	Pre-inspection stochastic elemental mass fraction state at stage n
$\boldsymbol{p}_n$	Post-inspection deterministic elemental mass fraction state at stage n
w_n	Cumulative weight of intermediate melt at the end of stage n
$\boldsymbol{s}_n$	Integrated system state at the end of stage n
$x_{n,1}$	Binary terminal indicator at stage n (0 = continue, 1 = terminate)
$\boldsymbol{x}_{n,2}$	Continuous raw material feeding decision vector at stage n
$\boldsymbol{I}_n$	Residual raw material inventory vector at the end of stage n
$\boldsymbol{C}$	Unit cost vector of raw materials
C_e	Unit processing cost for newly added raw materials
C_e^*	Unit holding cost of intermediate melt
C_f	Fixed operational cost per stage
C_v	Penalty coefficient for terminal elemental composition deviation
Π	Feasible set of stationary decision policies
π	Stationary policy for multi-stage stochastic decision-making

3. Two-layer Hybrid Decision-Making Method

For small-scale stochastic optimization problems requiring exact solutions, DP acts as a standard solution paradigm. Nevertheless, the recycled copper smelting blending problem investigated in this work involves high-dimensional continuous decision variables and stochastic state transitions, making conventional DP computationally intractable. Combining DP with heuristic algorithms enables complementary advantages, which has been

widely adopted in existing literature. Accordingly, this study develops a bi-level decision-making framework. Deep reinforcement learning (DRL) is employed for high-level sequential decision making and problem decomposition, while chance-constrained programming is embedded to guarantee the feasibility enforcement mechanism and risk control of low-level subproblem solutions.

3.1. Overall hybrid framework

DRL cannot guarantee solution optimality, and thus is unsuitable for directly determining raw material continuous feeding decisions. As a deep learning-based method, DRL is also dimension-sensitive, incurring prohibitive computational overhead in training and decision-making for large-scale problems. Nevertheless, DRL can efficiently produce high-quality suboptimal solutions with favorable flexibility. It has been validated that DRL can provide high-quality initial solutions to accelerate exact optimization solvers [27]. Integrating DRL with exact optimization allows complementary strengths, where DRL simplifies the solution space for exact algorithms.

Accordingly, this work proposes a hybrid framework, as shown in Figure 2, comprising an upper-level DRL agent, a lower-level exact optimization module, and a stochastic simulator. The DRL agent decomposes the original problem into tractable subproblems solved via chance-constrained programming. The required total feeding weight and confidence level α are both determined by the high-level DRL agent. With these parameters, the low-level module derives the optimal raw material feeding vector. As a dedicated training component, the stochastic simulator samples stochastic raw material compositions, updates the intermediate melt state, and feeds the updated state and immediate reward back to the DRL agent.

Starting from the initial system state $\boldsymbol{s}_0$, the DRL agent outputs sequential decisions at each stage. The smelting process terminates once a terminal indicator is activated; otherwise, the lower-level module allocates raw materials under probabilistic feasibility constraints. Following feeding and melt inspection, the simulator updates elemental composition and cumulative melt weight to form a new system state, which is fed back to the DRL agent for next-stage decision-making.

This paradigm decouples policy learning from feasibility regulation. The upper-level agent explores the hybrid discrete-continuous action space, while the lower-level module ensures all continuous feeding decisions satisfy risk

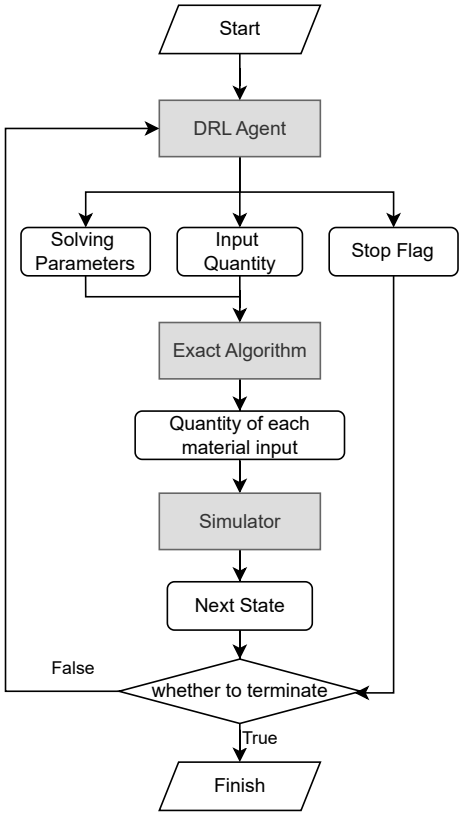


Figure 2: Algorithm workflow. The main components include the DRL agent, the exact algorithm, and the simulator. The exact algorithm can also be embedded in the simulator as part of the simulation process.

constraints. The proposed framework inherently combines the adaptive exploration of DRL with the rigorous feasibility control of exact optimization.

3.2. Upper-level policy learning: DAH-DDPG

The upper-level decision-making process involves discrete terminal indicator actions and continuous feeding-related control variables. Traditional DQN is designed exclusively for discrete action spaces [28], while DDPG is tailored to continuous control tasks [29, 23]. Although advanced actor–critic variants have improved training stability and sample efficiency [22, 30], standard DQN and DDPG are only applicable to discrete and continuous action spaces, respectively. Accordingly, the hybrid action space formulated

in this study necessitates further modifications to the conventional DDPG framework.

To address this limitation, this work develops a dual-actor hybrid deep deterministic policy gradient algorithm, termed DAH-DDPG. As illustrated in Figure 3, DAH-DDPG consists of six neural networks: F-Actor, S-Actor, and Critic, together with their corresponding target networks. The F-Actor undertakes the terminal indicator decision, whereas the S-Actor generates continuous stage-wise parametric variables. The Critic evaluates the state–action value and guides the parameter updating of both actor networks.

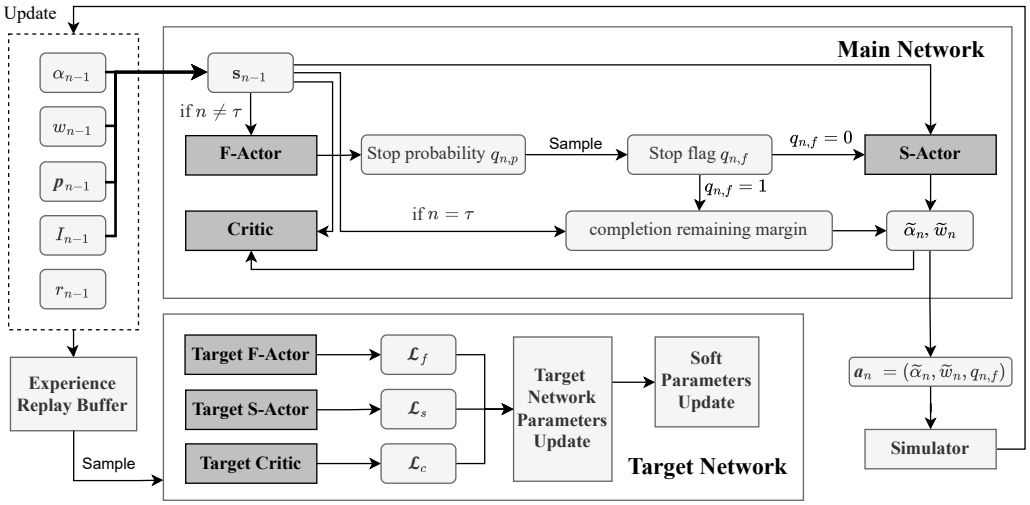


Figure 3: DRL framework. The main components include the main network and the target network. Together with the simulator and experience replay buffer, they form an iterative update process until reaching the maximum number of iterations or satisfying early stopping criteria.

Given the observed system state $\boldsymbol{s}_{n-1}$, the F-Actor outputs the stopping probability $q_{n,p}$. The binary terminal indicator $q_{n,f}$ is subsequently sampled as

$$q_{n,f} \sim \text{Bernoulli}(q_{n,p}), \tag{13}$$

where $q_{n,f} = 1$ denotes process termination after the current terminal stage, and $q_{n,f} = 0$ represents the opposite case.

Subsequently, two parameters are determined as the control actions. The first is the total weight to be fed in the current stage, which determines the total feeding amount and is denoted as $\tilde{w}_n$. The second is the confidence

increment $\tilde{\alpha}_n$, which is adopted to regulate the relaxation degree in the CCP solution process.

When the current stage corresponds to the final feeding operation, the remaining weight is supplemented such that w_τ equals W. Furthermore, to guarantee the quality of the final product, α_τ must be strictly constrained. Specifically, $\alpha_\tau = 1$ will lead to computational intractability or even infeasibility, while leaving α_τ unconstrained will result in uncontrollable results. This paper intends to control the product quality in a probabilistic manner. Accordingly, let α^* be the target confidence level with the constraint $\alpha_\tau \leq \alpha^*$. Under this condition, when $q_{n,f} = 1$, we have

$$\tilde{\alpha}_n = \alpha^* - \alpha_{n-1}, \tag{14}$$

$$\tilde{w}_n = W - w_{n-1}. \tag{15}$$

In contrast, when $q_{n,f} = 0$, the two actions mentioned above are generated by the S-Actor based on the system state of the previous stage. Thus, the S-Actor is not always invoked.

Once the stage action is determined, the confidence level used for the current-stage subproblem and the stage-end melt weight are updated as $\alpha_n = \alpha_{n-1} + \tilde{\alpha}_n$ and $w_n = w_{n-1} + \tilde{w}_n$, respectively. Let $\boldsymbol{a}_n$ denote the action vector of the S-Actor within the action space $\mathcal{A}$, which can be formulated as

$$\boldsymbol{a}_n = (\tilde{\alpha}_n, \tilde{w}_n, q_{n,f}). \tag{16}$$

Furthermore, to bound the problem scale, the maximum number of stages is restricted during solution solving. If the predefined maximum terminal stage number T is reached, $q_{n,f}$ is forced to 1, making the current stage the final stage. The corresponding decisions follow the same formulations as Equations (14) and (15).

After generating the stage action $\boldsymbol{a}_n$, the action is transmitted to both the lower-level optimization module and the stochastic simulator. The simulator returns the updated system state $\boldsymbol{s}_n$ and immediate stage reward r_n. The state transition tuple

$$(\boldsymbol{s}_{n-1}, \boldsymbol{a}_n, r_n, \boldsymbol{s}_n, d_n) \tag{17}$$

is stored in the experience replay buffer, where d_n represents the binary terminal indicator.

The critic network is trained by minimizing the Bellman mean squared error loss, which is defined as:

$$\mathcal{L}_c = \left(Q_c(\boldsymbol{s}_{n-1}, \boldsymbol{a}_n) - [r_n + \gamma(1 - d_n)Q_c(\boldsymbol{s}_n, \boldsymbol{a}_{n+1})]\right)^2. \tag{18}$$

In this equation, $Q_c(\boldsymbol{s}_{n-1}, \boldsymbol{a}_n)$ denotes the Q-value estimated by the critic network for a given $\boldsymbol{s}_n$ and $\boldsymbol{a}_n$. The latter term in the formula corresponds to the target Q-value derived from the standard Bellman equation, where $\boldsymbol{a}_{n+1}$ is the next action obtained from state $\boldsymbol{s}_n$ by actors.

The F-Actor is trained by maximizing the expected Q-value with respect to the terminal indicator decision. Given the binary nature of the stopping action, this expectation can be explicitly calculated via enumeration. For clear distinction, $\boldsymbol{a}_n^{(1)}$ denotes the action vector when $q_{n,f} = 1$, and $\boldsymbol{a}_n^{(0)}$ denotes that when $q_{n,f} = 0$. Accordingly, the loss function formulated by the expectation can be expressed as

$$\mathcal{L}_f = -q_{n,p} Q_c\big(\boldsymbol{s}_{n-1}, \boldsymbol{a}_n^{(1)}\big) - (1 - q_{n,p}) Q_c\big(\boldsymbol{s}_{n-1}, \boldsymbol{a}_n^{(0)}\big). \tag{19}$$

The S-Actor is updated only when the smelting process does not terminate at the current stage:

$$\mathcal{L}_s = -Q_c(\boldsymbol{s}_{n-1}, \boldsymbol{a}_n). \tag{20}$$

Accordingly, the S-Actor learns continuous feeding-related parameters merely for non-terminal stages. Target networks are introduced to stabilize the training process, and soft update strategy is adopted for all target network parameters.

3.3. Lower-level chance-constrained optimization

Taking the high-level decision outputs from DAH-DDPG as inputs, the lower-level optimization module solves the detailed raw material continuous feeding decisions vector $\boldsymbol{x}_{n,2}$. Specifically, the DAH-DDPG agent transmits the confidence increment $\tilde{\alpha}_n$ and the required input weight of the current stage $\tilde{w}_n$ to the lower-level module, which is further formulated as a single-stage stochastic optimization subproblem.

This study adopts CCP to address the stage-wise probabilistic feasibility constraints induced by parameter uncertainty. As an effective optimization approach, CCP has been extensively applied to uncertain blending optimization and various metallurgical production problems [7, 31]. Compared with robust optimization, CCP exhibits lower conservatism and can fully utilize the known distributional information of stochastic parameters [32].

Within the proposed hierarchical framework, the upper-level agent optimizes the confidence increment, which determines the stage-wise confidence level incorporated into the lower-level CCP model. At each individual stage,

the deterministic formulation of the feeding optimization subproblem is expressed as

$$\min_{\boldsymbol{x}_{n,2}\in\mathcal{X}_2}\{\boldsymbol{C}^T\boldsymbol{x}_{n,2}\}$$
$$\text{s.t.}\begin{cases}\Xi^T\boldsymbol{x}_{n,2}+w_{n-1}\boldsymbol{p}_{n-1}\le\boldsymbol{U}(w_{n-1}+\mathbf{1}^T\boldsymbol{x}_{n,2}),\\ \Xi^T\boldsymbol{x}_{n,2}+w_{n-1}\boldsymbol{p}_{n-1}\ge\boldsymbol{L}(w_{n-1}+\mathbf{1}^T\boldsymbol{x}_{n,2}),\\ \mathbf{1}^T\boldsymbol{x}_{n,2}=\tilde{w}_n,\\ 0\le\boldsymbol{x}_{n,2}\le\boldsymbol{I}_{n-1}.\end{cases}\tag{21}$$

Here, $w_{n-1}\boldsymbol{p}_{n-1}$ represents the elemental mass inherited from the previous melt state. The first two constraints define the upper and lower bounds under deterministic conditions. They are the core of all constraint conditions. The total feeding weight must equal the target value $\tilde{w}_n$ provided by DDPG-DAH. Moreover, the consumption of each raw material cannot exceed the initial inventory I_{n-1} of the current stage.

Expanding the above model to elemental level yields

$$\min_{\boldsymbol{x}_{n,2}\in\mathcal{X}_2}\left\{\sum_{j=1}^{M}x_{n,2}^{j}C_j\right\}$$
$$\text{s.t.}\begin{cases}\sum_{j=1}^{M}\xi_{ij}x_{n,2}^{j}\le U_i\left(w_{n-1}+\sum_{j=1}^{M}x_{n,2}^{j}\right)-w_{n-1}p_{n-1,i},\quad\forall i=1,\dots,N,\\ \sum_{j=1}^{M}\xi_{ij}x_{n,2}^{j}\ge L_i\left(w_{n-1}+\sum_{j=1}^{M}x_{n,2}^{j}\right)-w_{n-1}p_{n-1,i},\quad\forall i=1,\dots,N,\\ \sum_{j=1}^{M}x_{n,2}^{j}=\tilde{w}_n,\\ 0\le x_{n,2}^{j}\le I_{n-1}^{j},\quad\forall j=1,\dots,M.\end{cases}\tag{22}$$

By introducing the stage confidence level α_n, the upper compositional bound is reformulated as a chance constraint for $i=1,\dots,N$:

$$\Pr\left\{\sum_{j=1}^{M}\xi_{ij}x_{n,2}^{j}\le U_i\left(w_{n-1}+\sum_{j=1}^{M}x_{n,2}^{j}\right)-w_{n-1}p_{n-1,i}\right\}\ge\alpha_n.\tag{23}$$

Define

$$Y_i=\sum_{j=1}^{M}\xi_{ij}x_{n,2}^{j}.\tag{24}$$

Under the independence and normality assumption of raw material components, Y_i follows a normal distribution with

$$E[Y_i] = \sum_{j=1}^{M} \mu_{ij} x_{n,2}^j, \quad \text{Std}[Y_i] = \sqrt{\sum_{j=1}^{M} (\sigma_{ij} x_{n,2}^j)^2}. \tag{25}$$

Let $z_{\alpha_n} = \Phi^{-1}(\alpha_n)$, where $\Phi^{-1}(\cdot)$ denotes the inverse cumulative distribution function of the standard normal distribution. Eq. (23) can then be equivalently converted into a deterministic second-order cone constraint:

$$\sum_{j=1}^{M} \mu_{ij} x_{n,2}^j + z_{\alpha_n} \sqrt{\sum_{j=1}^{M} (\sigma_{ij} x_{n,2}^j)^2} \leq U_i \left(w_{n-1} + \sum_{j=1}^{M} x_{n,2}^j \right) - w_{n-1} p_{n-1,i}. \tag{26}$$

Similarly, the lower-bound chance constraint is transformed into

$$\sum_{j=1}^{M} \mu_{ij} x_{n,2}^j + z_{1-\alpha_n} \sqrt{\sum_{j=1}^{M} (\sigma_{ij} x_{n,2}^j)^2} \geq L_i \left(w_{n-1} + \sum_{j=1}^{M} x_{n,2}^j \right) - w_{n-1} p_{n-1,i}. \tag{27}$$

Eqs. (26) and (27) contain second-order cone terms, which increase computational burden in simulation-based training. To improve training efficiency, this study adopts a linear approximation scheme [33]. For $\alpha_n \geq 0.5$, it holds that $z_{\alpha_n} \geq 0$. By applying the triangle inequality, one obtains

$$\sqrt{\sum_{j=1}^{M} (\sigma_{ij} x_{n,2}^j)^2} \leq \sum_{j=1}^{M} \sigma_{ij} x_{n,2}^j. \tag{28}$$

Accordingly, a sufficient linear approximation for the upper-bound constraint is derived as

$$\sum_{j=1}^{M} (\mu_{ij} + z_{\alpha_n} \sigma_{ij}) x_{n,2}^j \leq U_i \left(w_{n-1} + \sum_{j=1}^{M} x_{n,2}^j \right) - w_{n-1} p_{n-1,i}. \tag{29}$$

The lower-bound constraint can be approximated in a consistent manner as

$$\sum_{j=1}^{M} (\mu_{ij} + z_{1-\alpha_n} \sigma_{ij}) x_{n,2}^j \geq L_i \left(w_{n-1} + \sum_{j=1}^{M} x_{n,2}^j \right) - w_{n-1} p_{n-1,i}. \tag{30}$$

Combining Eqs. (29) and (30), the complete linearized CCP subproblem at each stage is formulated as

$$\min_{x_{n,2}\in\mathcal{X}_2}\left\{\sum_{j=1}^{M}x_{n,2}^{j}C_j\right\}$$
$$\text{s.t.}\begin{cases}\displaystyle\sum_{j=1}^{M}(\mu_{ij}+z_{\alpha_n}\sigma_{ij})x_{n,2}^{j}\le U_i\left(w_{n-1}+\sum_{j=1}^{M}x_{n,2}^{j}\right)-w_{n-1}p_{n-1,i}\\ \qquad\qquad,\forall i=1,\dots,N,\\ \displaystyle\sum_{j=1}^{M}(\mu_{ij}+z_{1-\alpha_n}\sigma_{ij})x_{n,2}^{j}\ge L_i\left(w_{n-1}+\sum_{j=1}^{M}x_{n,2}^{j}\right)-w_{n-1}p_{n-1,i}\\ \qquad\qquad,\forall i=1,\dots,N,\\ \sum_{j=1}^{M}x_{n,2}^{j}=\tilde{w}_n,\\ 0\le x_{n,2}^{j}\le I_{n-1}^{j},\quad\forall j=1,\dots,M.\end{cases}\tag{31}$$

The resultant linearized model can be efficiently solved by off-the-shelf optimization solvers, including CVXPY, Gurobi, and COPT.

When $\alpha_n < 0.5$, the proposed linearization remains applicable as an approximate strategy, even though its sufficient-condition property is no longer valid. This approximation ensures computational tractability for the entire simulation-based policy learning framework.

3.4. Interaction between learning and optimization

The proposed hybrid framework combines DAH-DDPG and CCP via the stage-wise action $\boldsymbol{a}_n$. At each decision stage, DAH-DDPG does not directly output the complete raw material continuous feeding decisions vector. It instead provides high-level decisions, including the confidence increment, total feeding amount and terminal indicator. These decisions are used to build the lower-level CCP subproblem and judge the process termination.

Given the α_n and w_n, the linearized CCP model in Eq. (31) yields the optimal detailed continuous feeding decisions vector $\boldsymbol{x}_{n,2}$. The stochastic simulator then samples the realized raw material composition matrix Ξ. It further updates the system state following the stage-wise transition dynamics in Section 2. The immediate stage reward is calculated according to the cost structure defined in Section 2. The relevant cost components include the raw material cost c_r^n, processing cost c_e^n, fixed stage cost c_f^n, and terminal composition deviation penalty c_v^τ.

This hierarchical interaction allows the upper-level DRL module to explore long-term sequential policies. Meanwhile, the lower-level optimization module ensures local feasibility at each stage via the feasibility enforcement mechanism. The DRL agent can thus optimize the endogenous stopping time, feeding quantity and risk conservatism level. It avoids direct exploration of the high-dimensional continuous space for raw material allocation.

3.5. Training procedure

By integrating the upper-level DAH-DDPG module and the lower-level CCP module, the complete hybrid training algorithm is summarized in Algorithm 1.

The key training hyperparameters involve the discount factor γ, soft update coefficient ψ, and exploration rate ϵ. The learning rates of the F-Actor network, S-Actor network, and Critic network are denoted by l_f, l_s, and l_c, respectively. Other major settings contain the training interval f_t, batch size n_{bc}, replay buffer capacity n_{bf}, maximum training episodes n_r, and guided sample size n_{in}. In addition, θ_f^-, θ_s^-, and θ_c^- represent the parameters of the corresponding target networks.

Guided samples are preloaded into the replay buffer before formal training. These samples cover typical two-stage and three-stage smelting processes, with randomized decisions provided as guidance. When a model is randomly initialized, it is insensitive to various states. This often leads to poor action exploration at the early training stage. Guided samples can effectively accelerate convergence [22].

In the proposed model, two action networks need to cooperate. The S-Actor learns effective information more easily than the F-Actor. This imbalance easily causes the S-Actor to overfit, while the F-Actor diverges. Introducing guided samples promotes effective optimization in the early stage. It alleviates the instability caused by mismatched network updates. This is a common strategy in multi-network training to avoid divergence from poor coordination [34].

During the training phase, each sample is formulated as a five-tuple and stored in the replay buffer $\mathcal{B}$. Samples are not stored in units of complete decision sequences. This design eliminates the inherent influence of stage order on decision-making.

At each decision stage, DAH-DDPG cooperates with the CCP model to generate decisions. The stochastic simulator updates the melt state and

Algorithm 1: Hybrid DAH-DDPG and CCP algorithm

Input: γ, ψ, ϵ, l_f, l_s, l_c, f_t, n_{bc}, n_{bf}, n_r, n_{in}, and problem parameters defined in Section 2

Output: Trained DAH-DDPG parameters

Initialize θ_f, θ_s, θ_c, θ_f^-, θ_s^-, and θ_c^-;
Initialize replay buffer $\mathcal{B}$ and populate it with guided samples;
for $r = 1$ **to** n_r **do**
    Reset the simulator, set $\mathcal{S}_0$, set $\alpha_0 \leftarrow 0$, and let $n \leftarrow 1$;
    while $n \leq T$ **do**
        Generate the stopping probability $q_{n,p}$ using the F-Actor;
        Sample $q_{n,f} \sim \text{Bernoulli}(q_{n,p})$;
        if $rand(0,1) \leq \epsilon$ **then**
            Generate a random exploratory action within the feasible action range;
        else
            if $q_{n,f} = 1$ ***or*** $n = T$ **then**
                Compute $\tilde{\alpha}_n = 1 - \alpha_{n-1}$ and $\tilde{w}_n = W - w_{n-1}$;
                Set $\boldsymbol{a}_n \leftarrow (\tilde{\alpha}_n, \tilde{w}_n, 1)$;
            else
                Generate $(\tilde{\alpha}_n, \tilde{w}_n)$ using the S-Actor;
                Set $\boldsymbol{a}_n \leftarrow (\tilde{\alpha}_n, \tilde{w}_n, 0)$;
        Compute $\alpha_n \leftarrow \alpha_{n-1} + \tilde{\alpha}_n$ and $w_n \leftarrow w_{n-1} + \tilde{w}_n$;
        Solve the lower-level CCP subproblem and obtain $\boldsymbol{x}_{n,2}$;
        $\boldsymbol{s}_n, r_n, d_n \leftarrow \text{Simulator}(\boldsymbol{s}_{n-1}, \boldsymbol{a}_n, \boldsymbol{x}_{n,2})$;
        Store $(\boldsymbol{s}_{n-1}, \boldsymbol{a}_n, r_n, \boldsymbol{s}_n, d_n)$ in $\mathcal{B}$;
        if $\text{length}(\mathcal{B}) > n_{bf}$ **then**
            Remove the oldest samples exceeding the buffer capacity;
        if $r \equiv 0 \pmod{f_t}$ **then**
            Sample a minibatch from $\mathcal{B}$;
            Update θ_c, θ_f, and θ_s according to $\mathcal{L}_c$, $\mathcal{L}_f$, and $\mathcal{L}_s$;
            Perform soft updates of the target networks with coefficient ψ;
        if $q_{n,f} = 1$ **then**
            Break;
        $n \leftarrow n + 1$;

returns the immediate reward. The generated five-tuple sample is then stored into $\mathcal{B}$ for subsequent network training.

Network parameters are updated periodically every f_t episodes. Notably, the two actor networks are updated asynchronously. The F-Actor converges more slowly than the S-Actor. Thus, the F-Actor adopts a higher update frequency to facilitate stable coordinated optimization.

The entire training process terminates when the maximum episode number is reached or the early stopping criterion is satisfied.

4. Numerical Study

This section conducts numerical experiments to validate the performance of the proposed hybrid framework. The numerical analysis is organized into four parts, which comprehensively evaluate the framework from experimental settings, algorithm convergence, optimization performance, cost composition, and ablation mechanism.

4.1. Experimental setup

4.1.1. Problem generation

To fully validate the effectiveness and robustness of the proposed method, multiple heterogeneous numerical examples are established. The designed examples involve various problem scales and feature diverse distribution characteristics of raw material uncertainty.

Apart from the basic parameters M, N and W defined in the multi-stage optimization model, three binary indicators are further proposed to classify raw material uncertainty levels. Specifically, $\mathcal{F}_L$, $\mathcal{F}_M$ and $\mathcal{F}_H$ are defined to represent scenarios with low, medium and high raw material uncertainty, respectively. The uncertainty degree of each raw material is quantified by the mean coefficient of variation of elemental components CV_j. The interval division of the coefficient of variation for each uncertainty level is presented in Table 2.

Table 2: Parameter definitions for uncertainty levels

Parameter	CV interval
$\mathcal{F}_L = 1$	$[0.005,\ 0.05]$
$\mathcal{F}_M = 1$	$[0.05,\ 0.1]$
$\mathcal{F}_H = 1$	$[0.1,\ 0.2]$

Materials with different uncertainty levels usually have distinct prices, and a negative correlation generally exists between them. To characterize the inherent relationship between material composition fluctuation and procurement cost, the following mapping relationship is constructed:

$$C_j = \frac{k_1}{\log_2(CV_j + k_2)}, \tag{32}$$

where $k_1 = 1000$ and $k_2 = 1.5$ are calibrated model parameters. In practice, the uncertainty of raw material composition is negatively correlated with material quality. A significant marginal diminishing effect exists when CV_j is at a low level, where a small increase in uncertainty leads to a sharp drop in cost. Therefore, this study adopts a logarithmic reciprocal function to represent the relationship between uncertainty and material cost. The saturation property of the logarithmic function is used to describe the marginal attenuation. This design can well fit the nonlinear characteristics between quality fluctuation and pricing of recycled copper raw materials.

Accordingly, the problem scale is determined by the number of elemental components and raw material types. The uncertainty level is controlled by combining raw materials with low, medium, and high uncertainty. This experimental design enables the numerical cases to effectively reflect diverse practical production scenarios.

4.1.2. Benchmark methods and evaluation metrics

A simplified orthogonal experimental scheme is designed to compare the performance of different algorithms under diverse conditions. Four experimental factors are considered, namely $\mathcal{F}_L$, $\mathcal{F}_M$, $\mathcal{F}_H$ and problem scale. Four typical scale settings are adopted: 5E24M, 10E48M, 15E60M and 20E100M. For example, 5E24M denotes the case with 5 elemental components and 24 raw material types.

For the combined mode of uncertainty indicators $(\mathcal{F}_L, \mathcal{F}_M, \mathcal{F}_H)$, four typical heterogeneous combinations are selected:

$$(1,1,0), \quad (1,0,1), \quad (0,1,1), \quad (1,1,1).$$

These correspond to low – medium, low – high, medium – high, and low – medium – high hybrid uncertainty structures, respectively. Cases containing only a single uncertainty level are excluded, since such scenarios have limited optimization potential and cannot reflect the heterogeneous characteristics of actual raw material supply, and thus are not the focus of this study.

For each uncertainty combination and problem scale, ten independent repeated experiments are carried out. Single-stage linear programming (LP) and single-stage CCP are adopted as benchmark methods for fair comparison. The proposed hybrid method is abbreviated as DRL–CCP.

Single-stage LP ignores the inherent uncertainty of the problem and solves it merely as a conventional combinatorial optimization model. In contrast, single-stage CCP explicitly accounts for parameter uncertainty under the same combinatorial optimization framework. It can be regarded as a special case of the proposed algorithm with $\tau = 1$, where the blending process terminates directly at the first stage, and the required confidence level and total feeding weight are determined in a one-off manner.

The core evaluation metric is the cost ratio between the proposed method and the single-stage CCP benchmark:

$$\rho = \frac{C_{\text{DRL-CCP}}}{C_{\text{CCP}}}, \tag{33}$$

where $C_{\text{DRL-CCP}}$ denotes the average total cost of the multi-stage hybrid framework, and C_{CCP} represents the cost obtained by the single-stage CCP strategy. A value of $\rho < 1$ implies that the proposed method achieves a superior economic performance with lower overall cost.

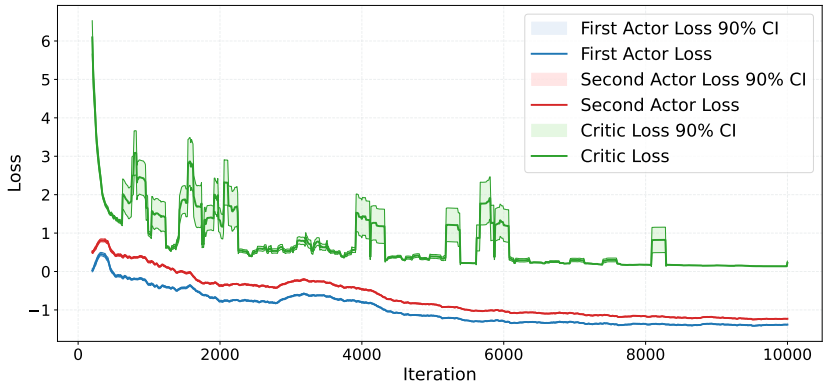
Hyperparameters of the algorithm are tuned via an iterative Bayesian optimization strategy. Since hyperparameter sensitivity is not the core focus of this work, the detailed tuning process is omitted for brevity.

4.2. Training convergence

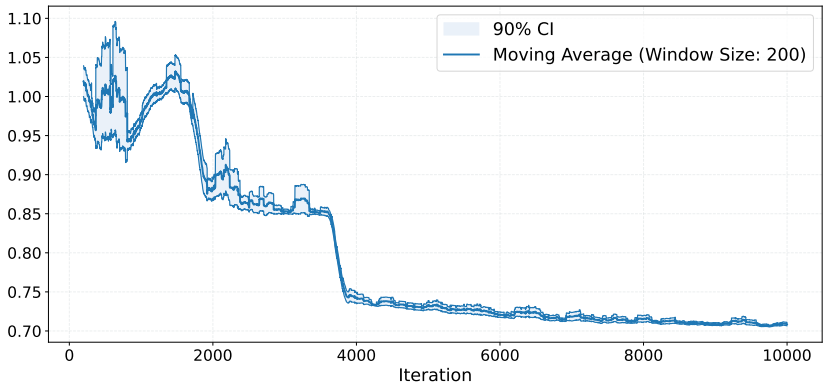
Before conducting performance comparison, the training stability and convergence behavior of the framework are first verified. Figure 4 illustrates the training evolution curves of key network components.

The loss values of the dual Actor networks and Critic network gradually decline and eventually stabilize within a narrow range during the training process. No persistent oscillation or divergence is observed across repeated experiments, which demonstrates that the designed dual-Actor network structure possesses favorable training stability.

The optimization metric fluctuates violently in the early training phase, which corresponds to the random exploration stage of policy learning. With sufficient interactive exploration, the metric gradually declines and converges to a stable level, indicating that the DRL agent continuously optimizes the multi-stage intermediate melt blending policy.



((a)) Convergence curves of the actor network loss in the second stage.



((b)) Convergence curve of the optimization metric.

Figure 4: Convergence curves of the model components, including the actor network losses, the Critic network loss, and the overall optimization metric.

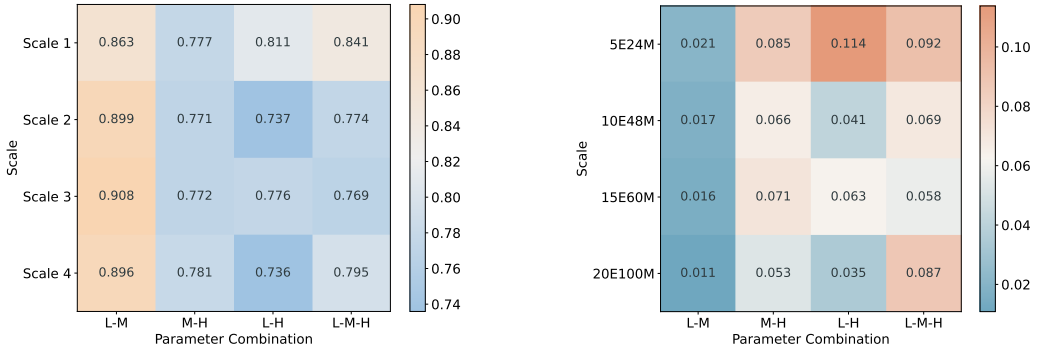
Further decomposition of individual cost components reveals that the training process essentially reflects the trade-off between raw material cost and terminal constraint violation penalty, which aligns with the overall cost minimization objective of the established multi-stage model. As shown in Figure 5, raw material cost and violation penalty both decline steadily after the initial exploration phase. By contrast, energy consumption cost and sampling cost remain relatively stable. This is because the decisions gradually converge to a steady state with algorithm iteration in the later training stage.

Figure 5: Cost component evolution during training. Material and violation costs decline after initial exploration, while energy and sampling costs remain stable.

4.3. Overall performance under orthogonal experiments

Based on the verified training stability, the overall optimization performance is evaluated under diverse problem scales and raw material uncertainty combinations. Figure 6 presents the mean and standard deviation of the optimization metric ρ defined in Eq. (33).

As observed from Figure 6(a), the proposed method achieves $\rho < 1$ under nearly all experimental configurations, demonstrating that the multi-stage closed-loop framework can effectively reduce the comprehensive cost compared with the static single-stage CCP strategy. In most cases, the cost ratio drops below 0.8, which validates the prominent economic advantage of the multi-stage sequential decision-making structure.



((a)) Heatmap of the mean optimization metric

((b)) Heatmap of the standard deviation of the optimization metric

Figure 6: Distribution of the optimization metric. For different problem scales and material uncertainty combinations, the ratio of the multi-stage total cost obtained by DRL–CCP to the single-stage CCP total cost is reported, together with its mean and standard deviation.

The performance gain varies with different uncertainty combinations. For low–medium hybrid uncertainty scenarios, the cost reduction margin is relatively limited. The underlying reason is that the overall uncertainty fluctuation is mild, leaving little room for multi-stage corrective adjustment of intermediate melt composition. By contrast, the proposed framework exhibits the most prominent superiority under low–high uncertainty configuration. Large heterogeneity among raw materials enlarges the value of sequential inspection and dynamic adjustment, thereby fully releasing the potential of the multi-stage decision mechanism.

Figure 6(b) further reflects the performance volatility of the optimization metric. The standard deviation maintains a low level under most settings, implying that the proposed method possesses stable generalization across different problem scales and uncertainty structures. In particular, low–high uncertainty cases achieve both excellent average optimization effect and small performance fluctuation. The average coefficient of variation over all experimental groups is 0.071, which further verifies the robustness of the proposed framework.

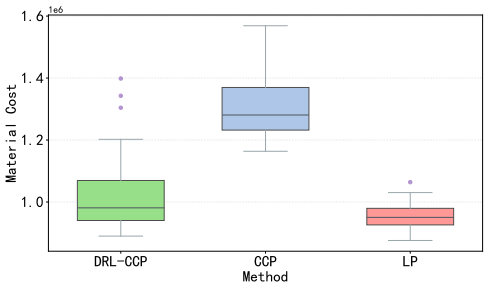
4.4. Cost-structure analysis

The above results verify the cost reduction advantage of the proposed method from a global perspective. This subsection further explores the internal source of performance improvement by decomposing the total cost into

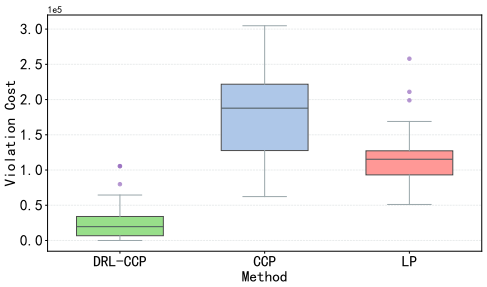
raw material purchase cost and constraint violation penalty cost. A typical case containing both low- and high-uncertainty raw materials is selected for detailed comparative analysis.

As illustrated in Figure 7, the raw material cost of the proposed method is slightly higher than that of single-stage linear programming, but remarkably lower than that of single-stage CCP. This indicates that the single-stage CCP method tends to adopt overly conservative and high-cost raw material allocation schemes to satisfy probabilistic feasibility constraints.

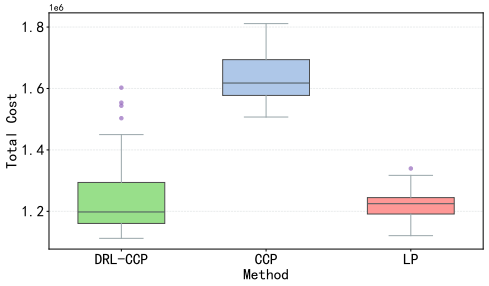
In terms of penalty cost induced by composition violation, the proposed method achieves the lowest level under stochastic intermediate melt state transition. Single-stage linear programming lacks explicit risk awareness and probabilistic feasibility control, thus resulting in severe constraint violation and high penalty cost. Single-stage CCP also suffers from non-negligible penalty cost in the representative case, owing to its static one-shot decision structure without the ability to dynamically adjust after uncertainty realiza-



((a)) Boxplot of material costs across different methods.



((b)) Boxplot of penalty costs across different methods.



((c)) Boxplot of total costs across different methods.

Figure 7: Comparison of material costs, penalty costs, and total costs across methods for a representative case.

tion.

The total cost comparison confirms that the proposed hybrid framework achieves the best comprehensive economic performance among the three methods. Although single-stage linear programming may obtain lower nominal material cost occasionally, such solutions usually accompany severe composition violation and thus cannot be applied in practical production. Instead, the proposed method establishes an effective trade-off between raw material cost and violation penalty via multi-stage dynamic regulation.

To intuitively compare the constraint violation characteristics of different methods, Figure 8 presents the elemental violation levels in radar chart form.

The radar charts demonstrate that the proposed method restricts elemental constraint violations within a small range while maintaining low total cost. Single-stage linear programming may perform well on several individual components but lacks systematic risk control, easily causing excessive violation on other dimensions. Single-stage CCP shows high decision conservatism but insufficient flexibility facing stochastic melt state evolution.

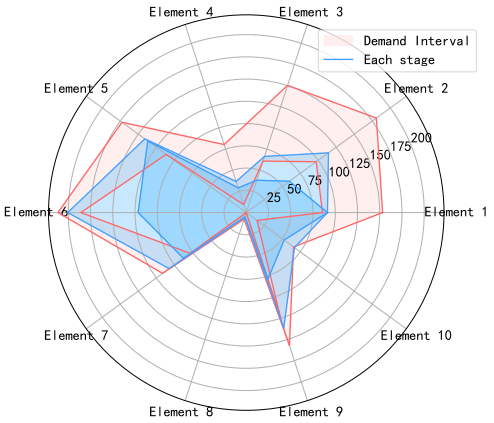
In summary, the superiority of the proposed method does not lie in minimizing a single type of cost alone, but in achieving a desirable cost–risk balance by virtue of multi-stage corrective decision-making based on intermediate melt state feedback.

4.5. Ablation study on closed-loop feedback

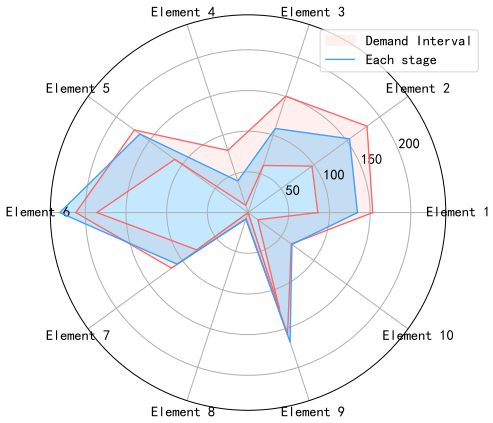
The preceding numerical results validate the overall effectiveness of the proposed framework. This subsection further conducts ablation experiments to identify whether the performance improvement stems from the closed-loop feedback mechanism of intermediate melt inspection information. The core research question is whether the lower-level optimization subproblem should incorporate the updated historical state $\boldsymbol{s}_{n-1}$ when constructing stage-wise allocation decisions.

To address this issue, an open-loop ablation variant is constructed, where the stage subproblem abandons the updated melt state feedback and only adopts static prior information. The deterministic open-loop formulation is expressed as

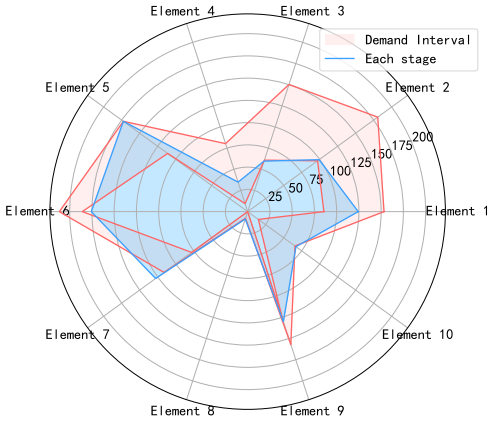
$$\begin{aligned} &\min_{\boldsymbol{x}_{n,2}\in\mathcal{X}_2}\{\boldsymbol{C}^T\boldsymbol{x}_{n,2}\} \\ &\text{s.t.}\begin{cases}\Xi^T\boldsymbol{x}_{n,2}\le \boldsymbol{U}\tilde{w}_n,\\ \Xi^T\boldsymbol{x}_{n,2}\ge \boldsymbol{L}\tilde{w}_n,\\ 0\le \boldsymbol{x}_{n,2}\le \boldsymbol{I}_{n-1}.\end{cases} \end{aligned} \tag{34}$$



((a)) Radar chart of DRL–CCP



((b)) Radar chart of CCP



((c)) Radar chart of LP

Figure 8: Comparison of radar charts of constraint violations under different methods for a representative case.

Correspondingly, the linearized solvable approximation model for open-

loop control is derived as

$$\min_{\boldsymbol{x}_{n,2}\in\mathcal{X}_2}\{\boldsymbol{C}^T\boldsymbol{x}_{n,2}\}$$
$$\text{s.t.}\begin{cases} U_i\tilde{w}_n-\boldsymbol{\mu}_i^T\boldsymbol{x}_{n,2}+U_i\sum_{j=1}^{M}x_{n,2}^j-z_{\alpha_n}\sum_{j=1}^{M}x_{n,2}^j\sigma_{ij}\geq 0, & \forall i=1,\ldots,N,\\ L_i\tilde{w}_n-\boldsymbol{\mu}_i^T\boldsymbol{x}_{n,2}+L_i\sum_{j=1}^{M}x_{n,2}^j+z_{\alpha_n}\sum_{j=1}^{M}x_{n,2}^j\sigma_{ij}\leq 0, & \forall i=1,\ldots,N,\\ 0\leq x_{n,2}^j\leq I_{n-1}^j, \\ j=1,\ldots,M. \end{cases} \tag{35}$$

Figure 9 illustrates the statistical results of the optimization metric under open-loop control across different problem scales and uncertainty combinations.

Compared with the full closed-loop framework, the open-loop strategy presents inferior optimization performance under all experimental settings, which confirms that intermediate melt state feedback is a crucial contributor to performance improvement. Nevertheless, the open-loop variant still outperforms the single-stage benchmark method in all cases, with an average optimization metric of 0.866. This reveals that the performance gain of the proposed framework originates from two aspects: the inherent structural advantage of multi-stage decision-making, and the additional benefit brought by closed-loop state feedback. The multi-stage structure alone already surpasses

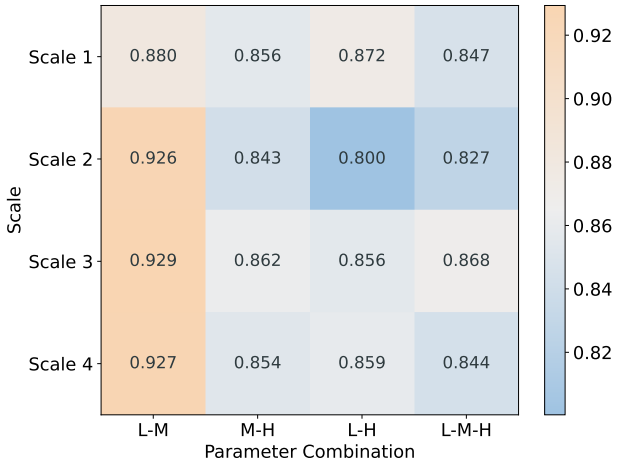


Figure 9: Distribution of the optimization metric under open-loop control. The figure reports the mean optimization metric under different problem scales and material uncertainty combinations when the agent does not use historical information.

static single-stage scheduling, while real-time feedback further amplifies the optimization margin.

The performance gap between closed-loop and open-loop strategies varies across scenarios. For low – medium uncertainty combinations, the performance difference is slight, implying that the marginal benefit of state feedback is limited under mild uncertainty fluctuation. By contrast, the closed-loop framework achieves remarkable superiority when high-uncertainty raw materials are involved, especially in medium- and large-scale instances. This phenomenon further verifies that inspection feedback and dynamic correction are most valuable under strong and heterogeneous raw material uncertainty.

4.6. Summary of numerical findings

The numerical experiments lead to three conclusive findings:

First, the proposed DAH-DDPG–CCP hybrid framework possesses stable training characteristics. All network components achieve steady and stable convergence throughout the training process; the losses of Actor and Critic networks converge steadily without persistent oscillation, and the optimization metric gradually declines and stabilizes within a narrow fluctuation range after the initial random exploration phase.

Second, the proposed method consistently outperforms single-stage CCP and linear programming benchmarks under various problem scales and raw material uncertainty structures, reducing the overall production cost by an average of 18.8%. The performance advantage is particularly prominent in scenarios with highly heterogeneous uncertainty, such as low – high hybrid uncertainty combinations. By contrast, only marginal cost reduction can be gained under mild overall uncertainty or small-scale problem instances. Even in the open-loop ablation scenario without real-time state feedback, the developed approach still surpasses single-stage baseline methods, and the full closed-loop feedback structure can further strengthen the economic benefit.

Third, the core competitiveness of the framework lies in its capability to balance raw material cost and constraint violation penalty via multi-stage closed-loop dynamic adjustment. Through iterative policy learning, the framework effectively coordinates raw material procurement cost and constraint violation penalty cost, achieving a more favorable cost–risk equilibrium under stochastic operational conditions. The ablation study further demonstrates that both the multi-stage decision structure and real-time intermediate melt state feedback jointly contribute to the overall performance improvement.

5. Conclusions and Future Work

This study investigates the multi-stage stochastic blending problem of recycled copper alloys. Considering raw composition uncertainty, sequential inspection feedback and endogenous stopping characteristics, a stochastic sequential decision model with chance constraints is established, which accurately describes the dynamic evolution and quality control requirements of practical smelting blending processes.

A DAH-DDPG algorithm is proposed to accommodate hybrid actions consisting of discrete shutdown and continuous feeding decisions. A two-layer framework integrating reinforcement learning and chance-constrained programming is further constructed. The upper layer handles sequential decision and terminal judgment, while the lower layer guarantees the probabilistic feasibility of each blending scheme, achieving an effective integration of policy learning and risk constraint control.

Numerical experiments verify the stable training performance and strong generalization ability of the proposed method. It evidently reduces the overall cost compared with benchmarks, and presents more prominent superiority under high uncertainty conditions. Ablation results confirm that the multi-stage structure and closed-loop state feedback are critical to performance improvement.

This work provides a reliable decision-making reference for the blending practice of recycled copper smelting, and supports the efficient utilization of low-cost recycled materials and reasonable cost control. Future research may introduce robust optimization strategies and develop dedicated efficient algorithms for such sequential stochastic blending problems to further extend the applicability of the proposed method in metallurgical production.

References

[1] Ü. Sakallı, Ö. Baykoç, B. Birgören, Stochastic optimization for blending problem in brass casting industry, Annals of Operations Research 186 (1) (2011) 141–157.

[2] G. Gaustad, P. Li, R. Kirchain, Modeling methods for managing raw material compositional uncertainty in alloy production, Resources, Conservation and Recycling 52 (2) (2007) 180–207.

[3] A. Noshadravan, G. Gaustad, R. Kirchain, E. Olivetti, Operational strategies for increasing secondary materials in metals production under uncertainty, Journal of Sustainable Metallurgy 3 (2) (2017) 350–361.

[4] M. Fröhling, O. Rentz, A case study on raw material blending for the recycling of ferrous wastes in a blast furnace, Journal of Cleaner Production 18 (2) (2010) 161–173.

[5] A. Prékopa, Stochastic Programming, Vol. 324 of Mathematics and Its Applications, Kluwer Academic Publishers, Dordrecht, 1995.

[6] A. Nemirovski, A. Shapiro, Convex approximations of chance constrained programs, SIAM Journal on Optimization 17 (4) (2006) 969–996.

[7] A. Rong, R. Lahdelma, Fuzzy chance constrained linear programming model for optimizing the scrap charge in steel production, European Journal of Operational Research 186 (3) (2008) 953–964.

[8] A. Ben-Tal, A. Nemirovski, Robust optimization–methodology and applications, Mathematical Programming 92 (3) (2002) 453–480.

[9] Y. Yang, W. Chen, L. Wei, X. Chen, Robust optimization for integrated scrap steel charge considering uncertain metal elements concentrations and production scheduling under time-of-use electricity tariff, Journal of Cleaner Production 176 (2018) 800–812.

[10] N. H. Lappas, C. E. Gounaris, Multi-stage adjustable robust optimization for process scheduling under uncertainty, AIChE Journal 62 (5) (2016) 1646–1667.

[11] D. Bertsimas, I. Dunning, Multistage robust mixed-integer optimization with adaptive partitions, Operations Research 64 (4) (2016) 980–998.

[12] M. Goerigk, M. Hartisch, Multistage robust discrete optimization via quantified integer programming, Computers & Operations Research 135 (2021) 105434.

[13] R. Paradiso, A. Georghiou, S. Dabia, D. Tönissen, Exact and approximate schemes for robust optimization problems with decision-dependent information discovery, INFORMS Journal on Computing 37 (6) (2025) 1457–1477.

[14] D. P. Bertsekas, Dynamic Programming and Optimal Control, Volume I, 4th Edition, Athena Scientific, Belmont, MA, 2012.

[15] G. Peskir, A. Shiryaev, Optimal Stopping and Free-Boundary Problems, Lectures in Mathematics ETH Zürich, Birkhäuser, Basel, 2006.

[16] J. Li, J.-C. Lee, A δv-learning approach for optimal stopping problems, European Journal of Operational Research 306 (1) (2023) 201–212.

[17] D. Russo, B. Van Roy, J. Zheng, Learning to stop with surprisingly few samples, Mathematics of Operations Research 46 (4) (2021) 1297–1319.

[18] F. Trevizan, S. Thiébaux, P. Santana, B. C. Williams, Heuristic search in dual space for constrained stochastic shortest path problems, in: Proceedings of the International Conference on Automated Planning and Scheduling, Vol. 26, 2016, pp. 326–334.

[19] S. Hong, B. C. Williams, An anytime algorithm for constrained stochastic shortest path problems with deterministic policies, Artificial Intelligence 316 (2023) 103846.

[20] C. Schmalz, F. Trevizan, Efficient constraint generation for stochastic shortest path problems, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 38, 2024, pp. 20150–20157. doi:10.1609/aaai.v38i18.30007.

[21] Z. Fan, R. Su, W. Zhang, Y. Yu, Hybrid actor-critic reinforcement learning in parameterized action space, in: Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI-19), 2019, pp. 2279–2285.

[22] S. Fujimoto, H. van Hoof, D. Meger, Addressing function approximation error in actor-critic methods, in: Proceedings of the 35th International Conference on Machine Learning, Vol. 80 of Proceedings of Machine Learning Research, 2018, pp. 1587–1596.

[23] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, D. Wierstra, Continuous control with deep reinforcement learning, published as a conference paper at ICLR 2016 (2015). arXiv:1509.02971.

[24] Y. Jiang, Q. Liu, L. Tang, W. Yu, Q. Li, W. Shi, Deep reinforcement learning approach with hybrid action space for mobile charging in wireless rechargeable sensor networks, Expert Systems with Applications 238 (2024) 121908.

[25] T. Wang, Y. Deng, Z. Yang, Y. Wang, H. Cai, Parameterized deep reinforcement learning with hybrid action space for edge task offloading, IEEE Internet of Things Journal 11 (6) (2024) 10754–10767.

[26] Y. Xu, Y. Wei, K. Jiang, L. Chen, D. Wang, H. Deng, Action decoupled sac reinforcement learning with discrete-continuous hybrid action spaces, Neurocomputing 537 (2023) 141–151.

[27] C. O'Malley, P. de Mars, L. Badesa, G. Strbac, Reinforcement learning and mixed-integer programming for power plant scheduling in low carbon systems: Comparison and hybridisation, Applied Energy 349 (2023) 121659.

[28] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, et al., Human-level control through deep reinforcement learning, nature 518 (7540) (2015) 529–533.

[29] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, M. Riedmiller, Deterministic policy gradient algorithms, in: Proceedings of the 31st International Conference on Machine Learning, Vol. 32 of Proceedings of Machine Learning Research, 2014, pp. 387–395.

[30] T. Haarnoja, A. Zhou, P. Abbeel, S. Levine, Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor, in: Proceedings of the 35th International Conference on Machine Learning, Vol. 80 of Proceedings of Machine Learning Research, 2018, pp. 1861–1870.

[31] Y. Chen, Y. Li, B. Sun, Y. Li, H. Zhu, Z. Chen, A chance-constrained programming approach for a zinc hydrometallurgy blending problem under uncertainty, Computers & Chemical Engineering 140 (2020) 106893.

[32] E. Roos, D. den Hertog, Reducing conservatism in robust optimization, INFORMS Journal on Computing 32 (4) (2020) 1109–1127.

[33] K. Ağpak, H. Gökçen, A chance-constrained approach to stochastic line balancing problem, European Journal of Operational Research 180 (3) (2007) 1098–1115.

[34] G. V. J. de la Cruz, Y. Du, M. E. Taylor, Pre-training with non-expert human demonstration for deep reinforcement learning, The Knowledge Engineering Review 34 (2019) e10.