Toward a High-Throughput Automated Materials Experimentation Platform: Hybrid-Automata-Inspired Modeling and Equipment Configuration Optimization

Guoxiong Ma, Haozhe Cui, Chun Yin Yip, Weibin Zhang, Molong Duan, *Member, IEEE*, and Jinglei Yang

***Abstract*—This paper presents an experimental-procedure-driven method for modeling and equipment quantity configuration of high-throughput automated materials experimentation platforms. The method integrates a hybrid-automata-inspired process model, a structured equipment dictionary, and experimental-procedure-to-equipment mapping to construct a unified platform model for timing and dependency analysis under practical constraints. Based on this model, equipment quantity configuration is formulated as a constrained integer nonlinear optimization problem and solved using an integer-coded differential evolution algorithm. A thermal insulation coating case study demonstrates the modeling procedure and resulting configuration scheme. The experiments show reduced operation time relative to manual execution and smaller standard deviations in coating transmittance. This study focuses on equipment quantity configuration rather than full platform layout and integration design.**

***Note to Practitioners*—This work presents a practical method for modeling high-throughput automated materials experimentation platforms and configuring equipment quantities under budget, footprint, and equipment-dependency constraints. The proposed framework helps practitioners translate an experimental procedure involving sample preparation, synthesis, and characterization into a platform-level representation that captures process states, equipment participation, execution sequences, and timing logic. Based on this representation, the method supports more systematic equipment quantity configuration than purely intuition-driven design. The current study focuses on equipment quantity configuration rather than full platform layout, control integration, or module assembly. The case study demonstrates how the framework can support the design of a compact and cost-aware automated platform for thermal insulation coating experiments.**

***Index Terms*—Hybrid automata, modeling, INLP optimization, high-throughput experiment platform, material experiment automation.**

Received 18 December 2025; revised 13 April 2026 and 15 May 2026; accepted 3 June 2026. Date of publication 22 June 2026; date of current version 26 June 2026. This article was recommended for publication by Associate Editor Y. Ding and Editor T. Nishi upon evaluation of the reviewers' comments. This work was supported in part by the Project of Hetao Shenzhen–Hong Kong Science and Technology Innovation Cooperation Zone under Grant HZQB-KCZYB-2020083, in part by the Innovation and Technology Commission of Hong Kong under Grant ITC-CNERC14SC01, and in part by the HKUST Shenzhen-Hong Kong Collaborative Innovation Research Institute. *(Guoxiong Ma and Haozhe Cui contributed equally to this work.) (Corresponding authors: Molong Duan; Jinglei Yang.)*

Guoxiong Ma, Chun Yin Yip, and Weibin Zhang are with the Department of Mechanical and Aerospace Engineering, The Hong Kong University of Science and Technology, Hong Kong, SAR, China (e-mail: gmaac@connect.ust.hk; cyyipai@connect.ust.hk; wzhangbx@connect.ust.hk).

Haozhe Cui is with the Academy of Interdisciplinary Studies, The Hong Kong University of Science and Technology, Hong Kong, SAR, China (e-mail: hcuiad@connect.ust.hk).

Molong Duan and Jinglei Yang are with the Department of Mechanical and Aerospace Engineering, The Hong Kong University of Science and Technology, Hong Kong, SAR, and also with the HKUST Shenzhen-Hong Kong Collaborative Innovation Research Institute, Shenzhen 518000, China (e-mail: duan@ust.hk; maeyang@ust.hk).

This article has supplementary downloadable material available at https://doi.org/10.1109/TASE.2026.3702087, provided by the authors.

Digital Object Identifier 10.1109/TASE.2026.3702087

I. Introduction

HIGH-THROUGHPUT automated experimentation has become an increasingly important paradigm in materials research, as it can accelerate sample preparation, characterization, and iterative screening while improving process repeatability and data consistency [1], [2], [3], [4]. Recent advances in robotic experimentation, autonomous laboratories, and data-driven materials discovery have further stimulated the development of customized automated platforms for specific experimental workflows [4], [5], [6], [7]. Compared with manual experimentation, such platforms provide more standardized execution, reduce operator-dependent variability, and shorten materials development and evaluation cycles. Recent studies have also emphasized system-level performance metrics, including throughput, reproducibility, autonomy, operational robustness, and robotic capability assessment, in self-driving platforms and automated laboratories [8], [9].

Automated experimental platforms in chemistry and materials science have demonstrated clear benefits in efficiency, consistency, and data quality. Representative examples include robotic synthesis platforms, autonomous laboratories for closed-loop materials discovery, and automated systems coupled with machine learning for experiment planning and optimization [10], [11], [12], [13], [14], [15], [16], [17]. These studies show that automation can increase throughput, reduce manual intervention, and generate more structured data for subsequent modeling and optimization. However, most existing efforts focus on experiment execution, reaction-condition optimization, workflow autonomy, robotic workflow execution, or closed-loop decision-making after the platform architecture has been determined, [4], [8], [18], [19]. By contrast, determining the required numbers of heterogeneous devices before platform implementation is a design-stage planning problem that has received limited explicit attention. This problem is practically important because equipment quantities directly affect achievable throughput, device utilization, investment cost, and the emergence of hidden workflow bottlenecks. In particular, early-stage equipment configuration under budget, footprint, and equipment-dependency constraints remains insufficiently studied. Related work in adjacent engineering domains and autonomous laboratory scheduling suggests that multi-type equipment selection, resource allocation, quantity optimization, and task scheduling can strongly affect system efficiency, bottleneck formation, and capacity utilization under resource constraints [20], [21]. Nevertheless, such studies typically optimize task allocation or execution schedules for given resources, whereas the design-stage determination of equipment quantities before platform implementation remains less explored. In practice, equipment planning for automated experimental platforms is still often handled heuristically, which may lead to resource waste, underutilized devices, or throughput-limiting bottlenecks.

This paper addresses the design-stage equipment quantity configuration problem for high-throughput automated materials experimentation platforms. The problem is formulated as a constrained optimization problem that balances experimental throughput and equipment utilization under budget, footprint, and inter-equipment dependency constraints. A central challenge is that platform design depends jointly on experimental procedure structure, equipment behavior, timing logic, and heterogeneous device parameters. Hybrid automata and related hybrid-system models provide a useful perspective for representing systems that combine discrete logical transitions with timing constraints or continuous dynamics [22]. Motivated by this perspective, we develop a hybrid-automata-inspired experimental-procedure-driven modeling framework that combines process abstraction with equipment-level representation. The framework describes procedure states, transitions, and timing logic through a hybrid-automata-inspired process model, and organizes heterogeneous equipment parameters through a structured equipment dictionary. These components are connected by experimental-procedure-to-equipment mapping functions to construct a unified platform model for timing and dependency analysis. Based on this model, the equipment configuration problem is formulated as an integer nonlinear programming problem and solved using an integer-coded differential evolution algorithm. Recent studies on discrete differential evolution suggest that evolutionary search can be effective for constrained and combinatorial decision spaces [23], [24]. The scope of this study is limited to equipment quantity configuration rather than full platform layout design, hardware integration, or control synthesis.

The main contributions of this work are fourfold. First, an experimental-procedure-driven modeling framework is developed to connect experimental process abstraction with equipment-level representation for high-throughput materials experimentation. Second, a structured equipment dictionary and an experimental-procedure-to-equipment mapping mechanism are introduced to support unified timing and dependency analysis. Third, equipment quantity configuration is formulated as a constrained integer nonlinear optimization problem under budget, footprint, and equipment-dependency constraints. Fourth, a thermal-insulation-coating case study is conducted to evaluate the proposed framework in terms of equipment configuration, throughput improvement, equipment utilization, and automated workflow execution.

II. Hybrid-Automata-Inspired Experimental-Procedure Modeling for High-Throughput Experimental Platforms

A. Experimental-Procedure Abstraction Based on a Hybrid-Automata-Inspired Representation

High-throughput automated experiments involve discrete process transitions, time-dependent process behavior, coordination among multiple devices, and repeated execution of experimental steps. Directly modeling all device-level interactions using a full hybrid automaton [25] would substantially increase state complexity and make subsequent optimization difficult to handle. To address this issue, we adopt a hybrid-automata-inspired experimental-procedure abstraction, in which each experimental step is represented as a process node associated with transition events, variable constraints, and continuous-time attributes. Instead of explicitly tracking every device state, the proposed abstraction focuses on process-level behavior and inter-step transitions, thereby providing a compact representation for platform-level analysis and optimization.

The abstract workflow model is defined as

$$F = \langle P, D_p, \Sigma, S_c, Init, Inv, \delta_c, \sigma_d \rangle, \tag{1}$$

where P denotes the set of experimental processes, D_p the labeled transitions between processes, Σ the event set, and S_c the set of continuous variables. The terms $Init$ and Inv specify initialization and process constraints, while δ_c and σ_d describe continuous evolution and discrete updates. A complete symbol definition and the case-specific instantiation are provided in the supplementary material.

B. Equipment Dictionary and Resource Representation

To represent heterogeneous laboratory resources in a unified manner, we introduce a structured equipment dictionary,

$$E = [e_s, e_o, e_e], \tag{2}$$

where e_s, e_o, and e_e denote the dictionaries for storage, operational, and experimental equipment, respectively. Storage equipment is used as sample containers or carriers, operational equipment provides transfer and positioning functions, and experimental equipment performs procedure-specific operations. For each equipment category, the dictionary records common design attributes as well as category-specific operational parameters. This unified representation supports timing evaluation, footprint estimation, cost aggregation, and dependency-constrained optimization.

For the thermal-insulation-coating case study, the dictionary includes sample containers and trays, transport and positioning devices, and process units for dispensing, stirring, drying, and characterization. Table I summarizes representative fields in

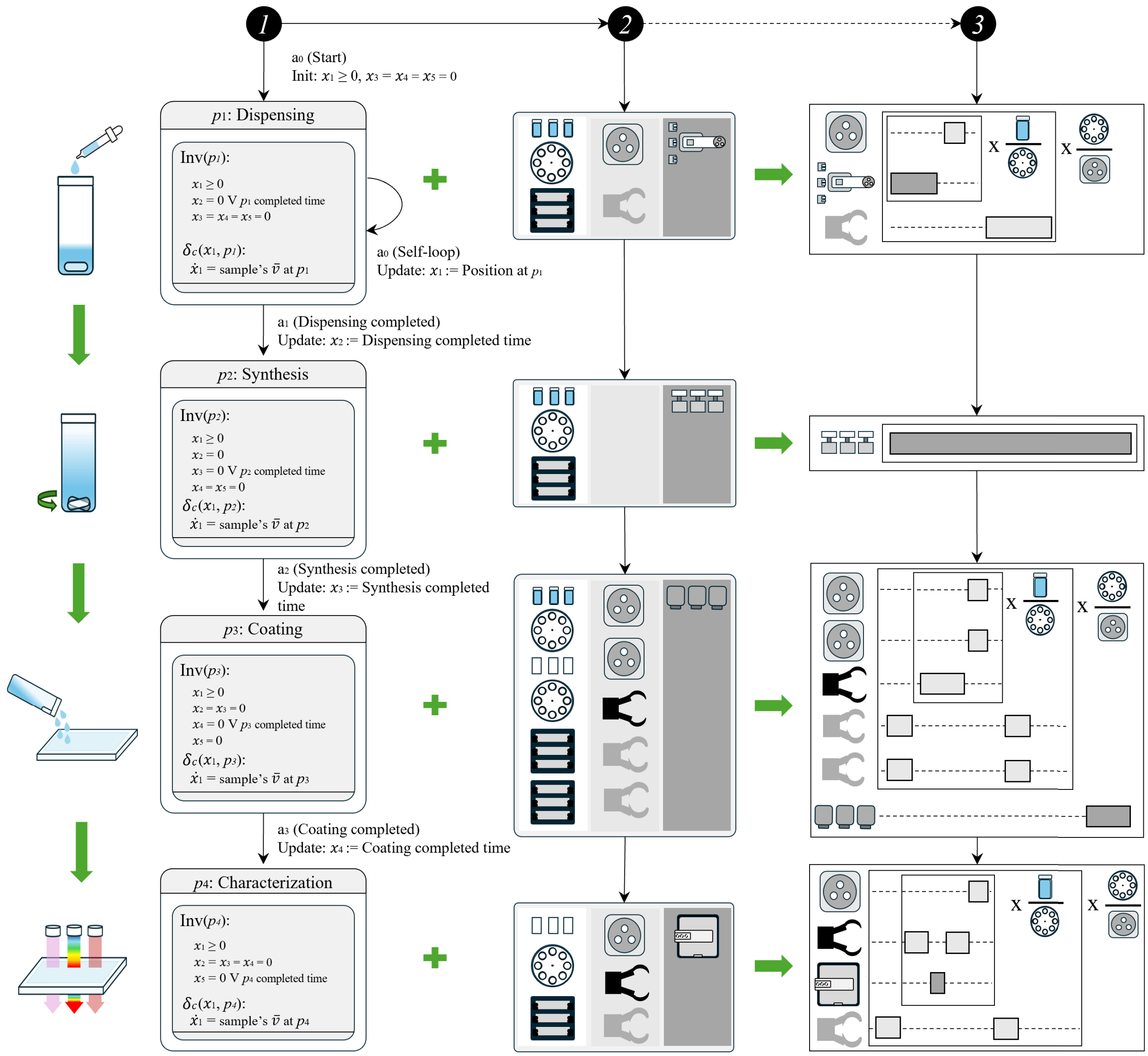


Fig. 1. Overview of the hybrid-automata-inspired experimental-procedure modeling framework for the high-throughput experimental platform.

TABLE I
REPRESENTATIVE FIELDS IN THE EQUIPMENT DICTIONARY

Category	Field	Symbol / key	Unit	Example
All	Quantity variable	q_i	count	q_1, q_7, q_{12}
All	Associated object	Obj	–	$e_2^s \to e_1^s$
Storage	Capacity	cap	ml, sample/tray, tray/case	25 ml, 3 trays/case
Operational	Maximum travel distance	$x^{\max}$	mm or rad	230 mm, 2π
Operational	Maximum speed	$v^{\max}$	mm/s or deg/s	10 mm/s, 15 deg/s
Experimental	Maximum operation time	$\tau^{\max}$	s or min	0.4 min, 180 min
Selected devices	Footprint width	l_w	mm	robot arm width
Selected devices	Footprint length	l_l	mm	feeder length
All	Unit cost	c_i	HKD/unit	bottle, turntable, feeder
All	Dependency rule	$R(q_i)$	–	$q_3 = q_1,\ q_7 = q_8 = q_9 = q_{15}$

the equipment dictionary, while the complete case-specific entries and parameter values are provided in the supplementary material. In Table I, the fields "Obj", q_i, and c_i correspond to the interaction target ε_β, quantity attribute ε_q, and cost attribute ε_c in the formal dictionary notation, respectively. The geometric descriptor ε_d includes footprint-related dimensions, such as width, length, and orientation parameters when required by the layout model. In addition, the dependency rules $R(q_i)$ encode coupled quantity constraints among equipment categories, such as shared carriers, matched trays, and synchronized module counts. The entries for storage, operational, and experimental equipment are defined as follows:

$$e_{s,i} = \langle \varepsilon_d : \{\theta_n, l_n\},\ \varepsilon_q : \mathbb{N}_{\geq 0},\ \varepsilon_c : \mathbb{R}_{\geq 0}, cap : cap_{\max} \vee$$

$$cap(\{c\}_{i=1}^{n}),\ \varepsilon_\beta : \{e\}\rangle \tag{3}$$

$$e_{o,i} = \langle \varepsilon_d : \{\theta_n, l_n\},\ \varepsilon_q : \mathbb{N}_{\geq 0},\ \varepsilon_c : \mathbb{R}_{\geq 0},\ x : x_{\max} \vee x(\{c\}_{i=1}^{n}), v : v_{\max} \vee v(\{c\}_{i=1}^{n}),\ \varepsilon_\beta : \{e\}\rangle \tag{4}$$

$$e_{e,i} = \langle \varepsilon_d : \{\theta_n, l_n\},\ \varepsilon_q : \mathbb{N}_{\geq 0},\ \varepsilon_c : \mathbb{R}_{\geq 0}, \tau : \tau_{\max} \vee \tau(\{c\}_{i=1}^{n}),\ \varepsilon_\beta : \{e\}\rangle \tag{5}$$

where cap, x, v, and τ denote storage capacity, travel distance, motion speed, and operation time, respectively.

C. Unified Platform Model for Timing and Dependency Analysis

The experimental-procedure model and the equipment dictionary are integrated into a unified platform model,

$$\mathcal{M} = \langle F, E, MOV, EQP, \Sigma', D'_p, S_e, S_o, Init', Inv', \delta'_c, \sigma'_d, S, t\rangle. \tag{6}$$

In this model, $EQP(p)$ maps each process node to the equipment participating in its execution, while $MOV(p)$ decomposes each process node into serial or parallel equipment-level subprocesses. The transformed event set Σ' and edge set D'_p characterize equipment-level transitions induced by process-level events. The variables S_e and S_o denote experiment-related and transport-related equipment variables, respectively, whereas S denotes the maximum sample capacity that can be processed by the platform over the procedure horizon. The resulting model provides a unified representation of process structure, equipment participation, execution timing, and dependency relationships, thereby supporting equipment quantity configuration analysis.

Based on this representation, the execution time of an equipment-level subprocess is computed as

$$t(e, p') = \begin{cases} e_e[\tau], & \text{if } e \in e_e, \\ \dfrac{\sigma'_d(e_o[x], (p', a, p''))}{\delta'_c(e_o[x], p')}, & \text{if } e \in e_o, \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

Here, the first case represents the operation time of experimental equipment, the second case represents the transport time of operational equipment, and the third case is used for equipment that does not directly contribute to the subprocess duration. This timing model provides the resource-coupling and equipment-dependency information required for the subsequent equipment quantity configuration problem.

For the thermal-insulation-coating case study, the procedure-to-equipment mapping explicitly expands each experimental stage into equipment-level subprocesses. For example, the coating stage is decomposed into tray transition, bottle handling, sheet handling, coating, position update, and drying-related subprocesses. Some subprocesses must be executed sequentially, whereas others can proceed in parallel. In particular, after each coating cycle, the bottle-side and sheet-side turntables can synchronously rotate to the next bottle and glass positions, while the robotic arm continues operating between fixed transfer points. This decomposition enables the unified platform model to capture both timing aggregation and inter-equipment dependencies in a form suitable for subsequent optimization.

To provide a concrete illustration of the procedure-to-equipment mapping, the coating-stage process node can be expanded into the following equipment-level subprocesses:

- bottle-tray transition → {tray, turntable A, optional carrier}, with serial or parallel-adjusted execution and a duration determined by motion distance and transport speed;
- sheet-tray transition → {sheet tray, turntable B, optional carrier}, with serial or parallel-adjusted execution and a duration determined by motion distance and transport speed;
- coating → {bottle, sheet, robotic arm}, executed serially with a duration determined by robotic transfer and action time;
- position update → {turntable A, turntable B}, executed in parallel with a duration determined by the maximum branch time; and
- surface drying → {sheet, drying unit}, treated as a batch-adjusted operation with a duration determined by the fixed drying time.

This example illustrates how a process-level node is expanded into equipment-level events and how serial, parallel, and batch-adjusted timing contributions are aggregated within the unified platform model. More detailed examples of formula application and schematic illustrations of the procedure-to-equipment expansion process are provided in the supplementary material.

D. Problem Formulation

The decision variable is the equipment-quantity vector

$$Q = \{q_1, q_2, \ldots, q_n\}, \tag{8}$$

where each $q_i \in \mathbb{Z}_{\geq 0}$ denotes the selected quantity of an equipment category and corresponds to the quantity attribute ε_q in the equipment dictionary.

The optimization objective is defined as

$$f(Q) = w_1 \left(1 - e^{-k \cdot T_h(Q)}\right) + w_2 R_u(Q), \tag{9}$$

where w_1 and w_2 are weighting coefficients, $T_h(Q)$ denotes the throughput term, $R_u(Q)$ denotes the utilization term, and k controls the saturation rate of the throughput contribution. The exponential transformation maps throughput to a bounded range while preserving the monotonic preference for higher throughput. Since the transformed term is strictly concave for $k > 0$, the marginal gain of throughput decreases as throughput increases. Thus, k regulates the sensitivity of the objective to throughput variation and helps balance processing efficiency and resource utilization. The weights w_1 and w_2 reflect the design preference between throughput and utilization. In this study, $w_1 = w_2 = 0.5$ is adopted as a neutral default setting in the absence of task-specific Preferences, and $k = 0.5$ is selected by empirical calibration to maintain sensitivity over the expected throughput range. The effect of these parameter choices is evaluated in Section IV-D.

The throughput term is defined as

$$T_h(Q) = S \cdot \left[\sum_{p \in P} T_p(Q)\right]^{-1}, \tag{10}$$

TABLE II
REPRESENTATIVE EQUIPMENT-DEPENDENCY RULES IN THE COATING CASE STUDY

Relation	Interpretation
$q_3 = q_1$	One sheet per sample
$q_4 = q_2$	Matched tray quantities
$q_7 = q_8 = q_9 = q_{15}$	Coupled transport/characterization
$q_{13} = q_1,\ q_{14} = q_1$	Stirring/drying scale with samples

where S is the maximum number of samples processed over the procedure horizon, and $T_p(Q)$ is the execution time of process node p under equipment configuration Q. The process time is obtained by aggregating equipment-level subprocess durations under serial, parallel, and batch-adjusted execution modes. Thus, the throughput term reflects the overall sample-processing capacity of the configured platform.

The utilization term is defined as

$$R_u(Q) = \frac{1}{3}U(Q), \tag{11}$$

where $U(Q)$ aggregates the utilization of storage, operational, and experimental equipment over the experimental procedure. The factor $\frac{1}{3}$ normalizes the aggregated utilization over the three equipment classes. This term is introduced to discourage configurations with severely underutilized equipment while maintaining a throughput-oriented optimization objective.

E. Constraints

The equipment quantity configuration problem is subject to three classes of constraints. First, equipment-dependency constraints enforce feasible quantitative relationships among interacting equipment categories:

$$Q \in \mathcal{Q}_d, \tag{12}$$

where $\mathcal{Q}_d$ denotes the feasible set induced by equipment interaction, matching, and synchronization requirements.

Second, the total platform footprint must not exceed the available workspace:

$$A(Q) \leq A_{\max}, \tag{13}$$

where $A(Q)$ denotes the total occupied area determined by the selected equipment quantities.

Third, the total equipment cost must satisfy the budget limit:

$$C(Q) \leq C_{\max}, \tag{14}$$

where $C(Q)$ denotes the total investment cost of the platform configuration.

Together, these constraints ensure that the optimized solution is not only high-performing in terms of throughput and utilization, but also physically deployable and economically feasible. In the thermal-insulation-coating case study, the available laboratory table area is set to 0.54 m^2, and the total budget limit is set to \$11,000.

Table II lists representative equipment-dependency rules used in the case study. These rules capture matching relations between samples and substrates, tray coordination requirements, and coupling relations among transport, processing, and characterization resources. The complete constraint set and parameter definitions are provided in the supplementary material.

TABLE III
DIFFERENTIAL EVOLUTION PARAMETER SETTINGS

Parameter	Setting
Strategy	best2bin
Decision dimension	15
Population multiplier	15
Approximate population size	225
Mutation factor	[0.5, 1.0]
Crossover probability	0.7
Maximum generations	30000
Stopping tolerance	10^{-5}
Polishing	Disabled
Independent runs	20

TABLE IV
REPEATED-RUN STATISTICS OF THE DE-BASED OPTIMIZER

Metric	Value
Independent runs	20
Feasible runs	20
Feasible rate	100%
Unique final solutions	1
Best objective	0.70039
Mean objective	0.70039
Std. objective	1.14×10^{-16}
Best throughput	3.19810
Mean throughput	3.19810
Std. throughput	4.56×10^{-16}
Mean runtime (s)	89.61
Std. runtime (s)	56.11

F. Solution Method

The resulting optimization problem is discrete, nonlinear, and constrained by coupled resource relationships. We therefore adopt a differential-evolution-based heuristic [27] with integer quantity encoding to search for high-quality feasible equipment configurations.

Each candidate solution is represented by an equipment-quantity vector $Q = [q_1, q_2, \ldots, q_{15}]$, where each component denotes the number of units selected for one equipment category. Before optimization, each variable q_i is assigned integer lower and upper bounds. The lower bounds are determined by process-feasibility and dependency requirements, whereas the upper bounds are restricted to practically meaningful ranges based on sample scale, budget, footprint, and preliminary trial considerations. During optimization, the differential evolution operators are applied to a real-valued intermediate vector. The resulting vector is then converted into an integer equipment configuration by floor-based discretization and bound clipping before objective evaluation.

Constraint handling is performed using an exterior penalty strategy. After integer conversion, each candidate configuration is checked against the dependency, footprint, and budget constraints. Infeasible candidates are assigned a large penalized objective value, whereas feasible candidates are evaluated using the unified platform model. Since the differential evolution routine is implemented in minimization form, the negative value of the maximization objective is returned for feasible candidates. Table III summarizes the differential evolution settings, and Table IV reports repeated-run statistics under different random seeds.

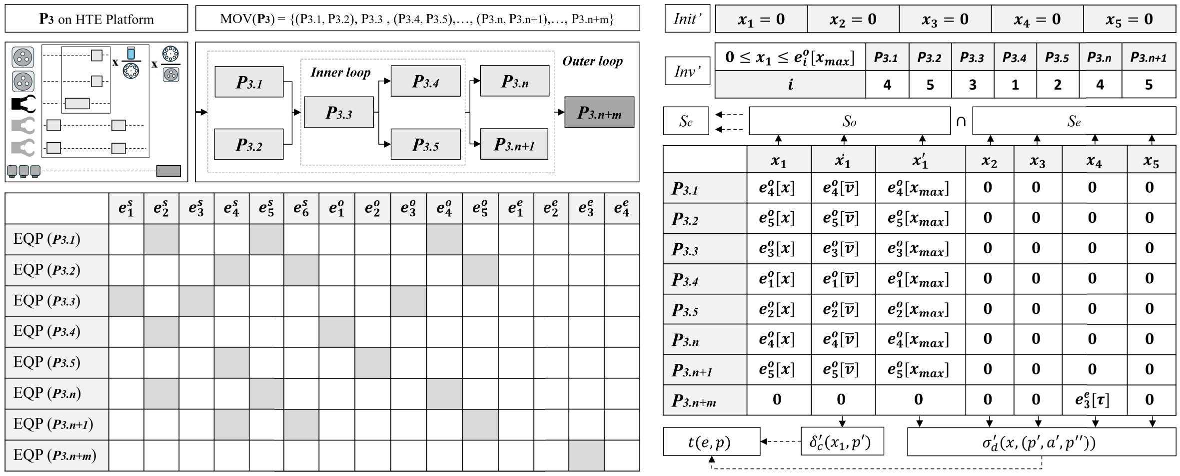


Fig. 2. Mapping of the coating workflow to the unified platform model $\mathcal{M}$, showing process-level abstraction, equipment assignment, and timing-related interactions.

TABLE V

REPRESENTATIVE HIGH-QUALITY FEASIBLE EQUIPMENT CONFIGURATIONS

Solution	$f(Q)$	T_h	Cost	Area
1	0.7004	3.1981 samples/h	10898.0	0.5215
2	0.6857	2.9468 samples/h	10849.5	0.4978
3	0.6688	2.6929 samples/h	10801.0	0.4754
4	0.6493	2.4363 samples/h	10752.5	0.4543
5	0.6268	2.1770 samples/h	10704.0	0.4347

Solution	Equipment configuration $[q_1, q_2, \ldots, q_{15}]$
1	[12, 1, 12, 1, 0, 0, 1, 1, 1, 0, 0, 1, 12, 12, 1]
2	[11, 1, 11, 1, 0, 0, 1, 1, 1, 0, 0, 1, 11, 11, 1]
3	[10, 1, 10, 1, 0, 0, 1, 1, 1, 0, 0, 1, 10, 10, 1]
4	[9, 1, 9, 1, 0, 0, 1, 1, 1, 0, 0, 1, 9, 9, 1]
5	[8, 1, 8, 1, 0, 0, 1, 1, 1, 0, 0, 1, 8, 8, 1]

III. OPTIMIZATION RESULTS AND DESIGN INSIGHTS

Based on the proposed INLP formulation and the differential-evolution solver, representative high-quality feasible equipment configurations were obtained for the target high-throughput experimental platform. These results illustrate how throughput changes under coupled space and budget constraints, and reveal the design trade-off between sample parallelization and investment in supporting equipment.

A. Representative Feasible Configurations

Table V summarizes five representative high-quality feasible solutions. The highest-ranked solution achieves the largest throughput among the listed configurations while satisfying both the budget and footprint constraints. The results show that the numbers of stirring and drying units increase with the number of sample containers, enabling greater parallel sample processing during the corresponding stages. In contrast, several auxiliary devices, such as the robotic arm and the two electric turntables, remain single-unit resources. This suggests that these auxiliary devices are not the dominant throughput bottlenecks under the current procedure structure and timing conditions.

B. Trade-off Analysis

A notable observation is that the tray-case and tray-carrier variables are consistently assigned zero in the optimized

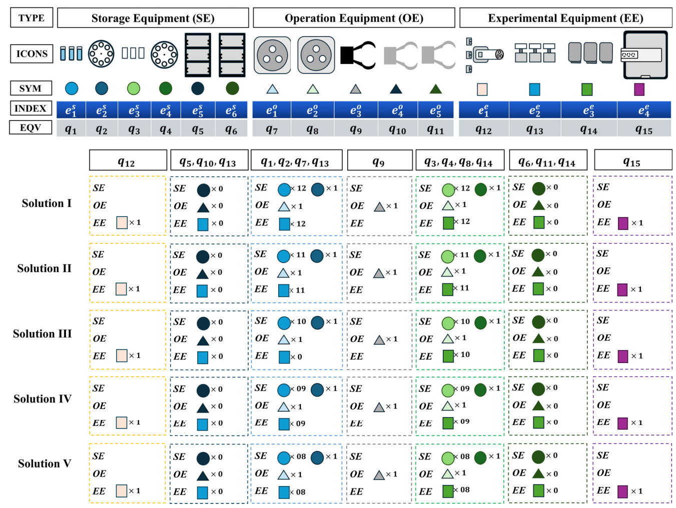
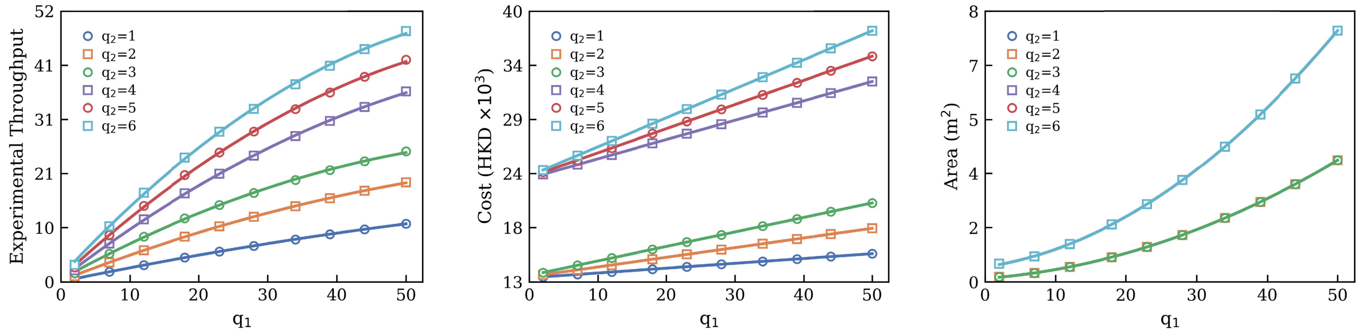


Fig. 3. Visualization of representative feasible equipment configurations obtained by the proposed optimization framework.

configurations. This indicates that, under the current budget and space constraints, introducing the tray-based design does not provide sufficient throughput improvement to justify its additional equipment cost and footprint. Instead, the optimizer improves throughput primarily by increasing the number of sample containers and the corresponding stirring and drying units. The monotonic decrease in throughput from Solution 1 to Solution 5 further confirms that the achievable throughput is strongly associated with the degree of sample parallelization. Within the current resource envelope, investing in sample-linked processing capacity is therefore more effective than expanding tray-related supporting resources.

To further interpret this design tendency, a parameter study was conducted by varying the sample-container quantity q_1 and tray quantity q_2, while keeping the remaining equipment quantities at their minimum feasible levels. Figure 4 shows the corresponding changes in throughput, cost, and footprint. The results indicate that increasing the tray quantity beyond a moderate range substantially increases cost and occupied area, whereas the throughput gain remains limited. By contrast, increasing the number of sample containers provides a



(a) (b) (c)

Fig. 4. Effect of sample-container quantity and tray quantity on (a) throughput, (b) cost, and (c) footprint.

TABLE VI

COMPARISON BETWEEN THE GREEDY BASELINE AND THE DE-BASED METHOD FOR THE PRESENT COATING-PLATFORM CASE

Metric	Greedy baseline	DE-based method
Objective value	0.7004	0.7004
Throughput	3.1981	3.1981
Utilization	0.6029	0.6029
Footprint	0.5215	0.5215
Cost	10898.0	10898.0
Mean runtime (s)	0.00124	89.61

TABLE VII

EXPANSION HISTORY OF THE GREEDY BASELINE FOR THE PRESENT COATING-PLATFORM CASE

Step	Action	q_1	Obj.	T_h	Cost
1	Sample-line	4	0.4918	1.1119	10510.0
2	Sample-line	5	0.5343	1.3825	10558.5
3	Sample-line	6	0.5701	1.6501	10607.0
4	Sample-line	7	0.6006	1.9149	10655.5
5	Sample-line	8	0.6268	2.1770	10704.0
6	Sample-line	9	0.6493	2.4363	10752.5
7	Sample-line	10	0.6688	2.6929	10801.0
8	Sample-line	11	0.6857	2.9468	10849.5
9	Sample-line	12	0.7004	3.1981	10898.0

more favorable throughput improvement per unit of resource consumption.

C. Comparison With a Baseline Design Heuristic

To provide a non-evolutionary reference, we constructed a constraint-aware greedy baseline inspired by fix–propagate–repair heuristics for feasibility-preserving search under coupled constraints [28]. The baseline was implemented using the same objective function, procedure model, dependency constraints, budget limit, and area limit as the proposed DE-based method. It serves as a structured benchmark for the present design problem rather than as a general-purpose optimizer.

The baseline starts from a minimum feasible configuration and expands the design monotonically through a small predefined action set. After each tentative move, dependent decision variables are updated through a fix–propagate–repair procedure to preserve structural consistency and feasibility. Among all feasible candidate moves, the move with the largest objective improvement is accepted. Because the action set is intentionally restricted, the baseline represents a locally guided expansion heuristic rather than a global search method.

TABLE VIII

REPRESENTATIVE COMPARISON BETWEEN BASELINE AND DE METHODS UNDER EXTENDED RESOURCE LIMITS

$A_{\max}$	$C_{\max}$	Baseline			DE		
		Obj.	T_h	Cost	Obj.	T_h	Cost
0.8	15000	0.7503	4.4157	11140.5	0.7950	8.6600	13402.0
1.2	15000	0.7800	5.8007	11431.5	0.8021	11.0429	13984.0
1.6	15000	0.7926	6.8914	11674.0	0.8049	13.1461	14566.0
1.6	30000	0.7926	6.8914	11674.0	0.7995	17.0344	26804.0

For the present coating-platform case, the greedy baseline converged to the same final equipment configuration as the DE-based method. Table VI summarizes the final comparison, and Table VII reports the corresponding stepwise expansion path of the greedy baseline. In this case, the baseline selected the sample-line scaling action at every iteration, and no tray-line scaling step was chosen before termination. The occupied area also increased monotonically along the same path. These results indicate that the feasible region of the present case is highly structured and nearly monotonic, allowing a simple expansion heuristic to recover the same final design. Accordingly, the baseline achieved the same objective value, throughput, utilization, area, and cost as the DE-based method, while requiring substantially less computation time.

However, this result should not be interpreted as evidence that the baseline and the DE-based method are generally equivalent. The current case is strongly simplified by a narrow feasible region and a dominant monotonic expansion path. Under such conditions, the final configuration can be reached by repeatedly applying a small number of intuitive scaling moves. By contrast, the DE-based method is designed for more general configuration spaces, where multiple heterogeneous devices, auxiliary units, and coupling relations may create competing structural design patterns.

This distinction becomes clearer in the extended scenario study. Table VIII reports representative comparisons between the baseline and the DE-based method under selected area and cost limits. In this table, throughput is reported as the recovered original value T_h. The results show that, as the feasible region becomes larger, the DE-based method consistently identifies higher-quality configurations than the baseline. More importantly, the DE solutions are not merely scaled-up versions of the baseline design. As shown in Table IX, under $A_{\max} = 0.8$ and $C_{\max} = 15000$, the baseline solution is $[17, 1, 17, 1, 0, 0, 1, 1, 1, 0, 0, 1, 17, 17, 1]$,

TABLE IX

REPRESENTATIVE STRUCTURAL COMPARISON BETWEEN THE BASELINE AND DE-BASED CONFIGURATIONS UNDER $A_{\max} = 0.8$ AND $C_{\max} = 15000$

Method	Solution vector
Baseline	$[17, 1, 17, 1, 0, 0, 1, 1, 1, 0, 0, 1, 17, 17, 1]$
DE-based	$[36, 3, 36, 3, 1, 1, 1, 1, 1, 1, 1, 1, 36, 36, 1]$

TABLE X

BASELINE PARAMETER SETTING FOR THE SENSITIVITY ANALYSIS

Parameter	Baseline value
w_1	0.5
w_2	0.5
k	0.5
Footprint limit (m^2)	0.54
Cost limit (HKD)	11000

whereas the DE-based method identifies $[36, 3, 36, 3, 1, 1, 1, 1, 1, 1, 1, 1, 36, 36, 1]$. The latter corresponds to a multi-tray configuration in which 36 vials are distributed over three trays, together with the activation of tray-buffering and storage-related units. By contrast, the baseline mainly follows a single monotonic expansion path and therefore tends to increase the carrying capacity of the existing line without introducing an alternative tray-level design structure.

It is important to emphasize that the proposed DE-based framework does not aim to guarantee a mathematically proven global optimum for every instance. Rather, its role is to identify high-quality feasible configurations under coupled structural and resource constraints. This capability becomes more important when the feasible space is larger and structurally more diverse, because handcrafted monotonic expansion heuristics may become restrictive and miss promising configuration patterns. Therefore, although the greedy baseline is sufficient for the present tightly structured coating-platform case, the DE-based framework provides stronger exploration capability and better configuration quality in broader design spaces.

D. Sensitivity Analysis of Design Parameters

To further assess the robustness of the proposed optimization framework, a sensitivity analysis was conducted for four groups of influential parameters: the objective weights (w_1, w_2), the nonlinear throughput scaling parameter k, the area limit, and the cost limit. For each scenario, the DE algorithm was independently executed 20 times using different random seeds. The reported statistics include the feasibility rate, objective value, throughput, utilization, area, cost, runtime, and the number of distinct optimized solutions. For scenarios with infeasible runs, performance metrics were averaged over feasible runs only, while the feasibility rate was computed over all 20 independent runs. The baseline parameter setting used in this analysis is summarized in Table X.

Table XI summarizes the sensitivity results for the objective-related parameters, including the objective weights (w_1, w_2) and the throughput scaling parameter k. In all tested scenarios, the feasibility rate remained 100%, and the same optimized configuration was obtained in all repeated runs. The throughput, utilization, area, and cost values remained unchanged, while only the scalar objective value varied due to the different weighting schemes or nonlinear throughput rescaling. This result suggests that the proposed method is robust to moderate variations in the objective aggregation formulation. For the studied case, the ranking of feasible candidate designs appears to be preserved under the tested objective preferences and throughput scaling settings.

TABLE XI

SENSITIVITY TO OBJECTIVE WEIGHTS AND SCALING PARAMETER

Scenario	Feas.	Obj.	T_h	Util.
$(w_1 = 0.1, w_2 = 0.9)$	1.00	0.6224	3.1981	0.6029
$(w_1 = 0.3, w_2 = 0.7)$	1.00	0.6614	3.1981	0.6029
$(w_1 = 0.5, w_2 = 0.5)$	1.00	0.7004	3.1981	0.6029
$(w_1 = 0.7, w_2 = 0.3)$	1.00	0.7394	3.1981	0.6029
$(w_1 = 0.9, w_2 = 0.1)$	1.00	0.7784	3.1981	0.6029
$k = 0.2$	1.00	0.5377	3.1981	0.6029
$k = 0.5$	1.00	0.7004	3.1981	0.6029
$k = 1.0$	1.00	0.7810	3.1981	0.6029
$k = 2.0$	1.00	0.8006	3.1981	0.6029

TABLE XII

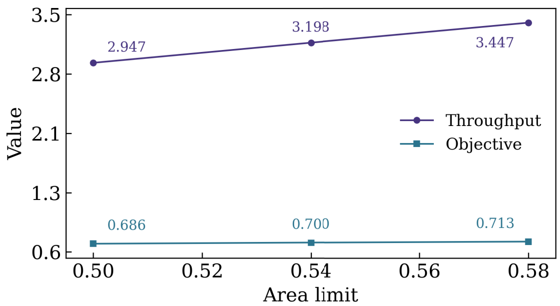
SENSITIVITY TO AREA AND COST CONSTRAINTS

Scenario	Feas.	Obj.	T_h	Util.	Footprint	Cost
$A_{\max} = 0.50$	1.00	0.6857	2.9468	0.6005	0.4978	10849.5
$A_{\max} = 0.54$	1.00	0.7004	3.1981	0.6029	0.5215	10898.0
$A_{\max} = 0.58$	1.00	0.7132	3.4468	0.6049	0.5466	10946.5
$C_{\max} = 10500$	0.30	0.4405	0.8384	0.5385	0.3591	10461.5
$C_{\max} = 11000$	1.00	0.7004	3.1981	0.6029	0.5215	10898.0
$C_{\max} = 11500$	1.00	0.7004	3.1981	0.6029	0.5215	10898.0

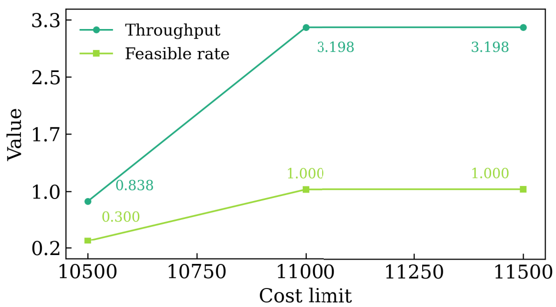
By contrast, the optimization outcome is more sensitive to resource constraints, as shown in Table XII. When the area limit increased from 0.50 to 0.58, the mean throughput increased monotonically from 2.9468 to 3.4468, accompanied by moderate increases in utilization, occupied area, and cost. Inspection of the optimized solutions further showed that the overall architecture pattern remained unchanged, while the main scale-related decision variables increased monotonically with the available area budget. This suggests that the area constraint does not alter the preferred system structure in the present case, but instead determines the admissible scale level of that structure.

The cost limit exhibited an even stronger impact. When the budget was reduced to 10500, the feasibility rate dropped to 30%, and the mean throughput decreased to 0.8384. By contrast, increasing the budget from 11000 to 11500 did not change the optimized configuration or the achieved performance. This indicates the presence of a practical cost threshold: once the budget exceeds the cost required by the preferred design, additional budget does not lead to further improvement. Therefore, compared with the objective-related parameters, the admissible resource budget, especially the cost bound, plays a more decisive role in shaping the final design outcome.

Figure 5 provides a visual summary of the resource-constraint sensitivity analysis. Figure 5(a) shows that the throughput and objective value increase steadily as the area limit is relaxed, confirming that additional area budget enables a larger-scale yet structurally similar design. Figure 5(b) shows that the cost limit produces a threshold effect: tightening the budget to 10500 substantially reduces both throughput and feasibility rate, whereas further budget increases beyond 11000 yield no additional gain.



(a)



(b)

Fig. 5. Sensitivity of optimization performance to resource constraints: (a) effect of area limit on throughput and objective value; (b) effect of cost limit on throughput and feasibility rate.

Overall, the sensitivity analysis shows that the proposed optimization framework is robust to the specification of the objective weights and the throughput scaling parameter in the present case. By contrast, the optimized configuration is more sensitive to the admissible resource budget, particularly the cost limit. These results suggest that, for the studied design problem, resource constraints play a more decisive role than moderate variations in objective aggregation.

E. Discussion of Container-Based Vs. Tray-Based Parallelization

The trade-off analysis provides a direct explanation for the optimizer's preference for container-based parallelization over tray-based expansion in the present case. Increasing the number of trays requires not only additional tray-handling capacity, but also matched supporting resources such as turntables, carriers, and storage-related units. These additions increase both cost and footprint through coupled structural requirements, while the resulting throughput improvement remains limited under the current resource bounds. By contrast, increasing the number of sample containers makes more effective use of the core processing resources and supports a higher level of parallel sample handling with lower marginal resource consumption.

This result suggests that, for the studied coating workflow, throughput improvement is influenced more strongly by the number of simultaneously processed samples than by the expansion of tray-handling units alone. More broadly, it illustrates the value of the proposed modeling and optimization framework in revealing design preferences that may not be obvious from intuition alone. In particular, when multiple resource couplings and auxiliary-unit dependencies are present, intuitive expansion strategies may overestimate the benefit of adding visible transport or tray-handling components, whereas the optimization model can identify configuration directions that deliver better overall system performance under explicit cost and footprint constraints.

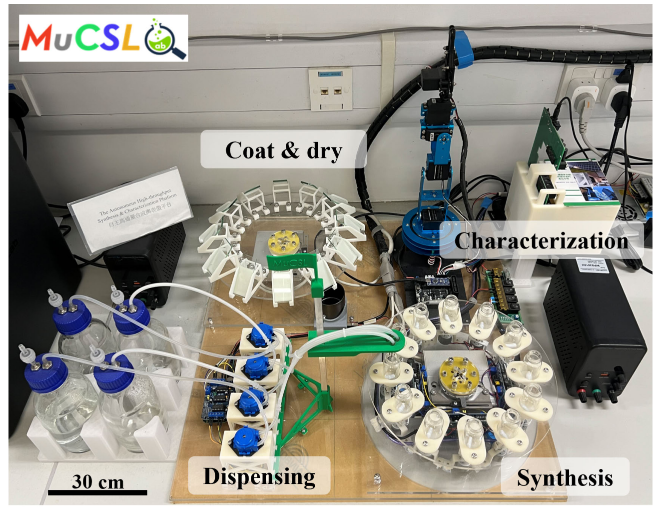


Fig. 6. Automated high-throughput platform for thermal insulation coating preparation and characterization.

IV. Experimental Validation on Thermal Insulation Film Coating

To evaluate the practical performance of the optimized platform in a representative coating workflow, thermal-insulation film coating experiments were conducted using both manual operation and the automated platform under the same formulation and measurement conditions. The validation focused on two aspects: workflow efficiency and optical transmittance. The main evaluation metrics included active operation time, process hold time, and transmittance at selected wavelengths.

A. Experimental Setup

Figure 6 shows the automated high-throughput platform used for thermal-insulation coating preparation and characterization. In the manual workflow, an experienced operator prepared coating samples using the doctor-blade method and completed the subsequent characterization steps. In the automated workflow, the optimized platform executed the corresponding dispensing, coating, drying, and characterization sequence. For direct comparison, both workflows were evaluated using three samples under the same formulation, measurement conditions, and time-accounting rule.

The reported active operation time included all human intervention required before and during execution, including setup, consumable handling, instrument-readiness confirmation, reagent handling, sample transfer, and measurement-related operations. Passive waiting periods were excluded from this metric. In the manual workflow, the preparation stage included bench arrangement, instrument-readiness confirmation, consumable preparation, and reagent preparation,

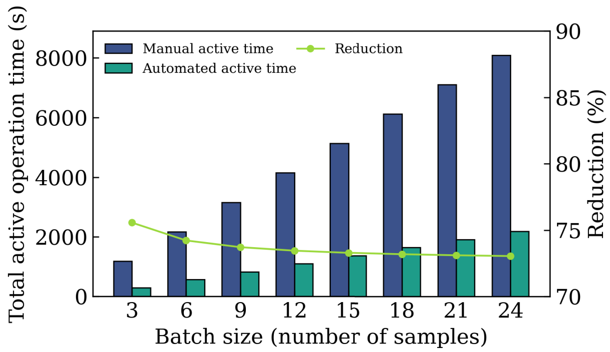


Fig. 7. Active operation time as a function of batch size for manual processing and the automated platform.

TABLE XIII

PER-SAMPLE COMPARISON OF MANUAL AND AUTOMATED WORKFLOW TIMING UNDER THE SAME ACCOUNTING RULE

Step (s/sample)	Manual	Automated
Preparation	201.3	20.0
Dispensing	237.3	33.0 ± 0.1
Coating and drying-related manipulation	71.3	20.3 ± 0.6
Characterization-related operation	20.0	37.3 ± 0.6
Total active operation time	529.9	110.6 ± 0.6
Process hold time	36000	36000

corresponding to 201.3 s per sample. In the automated workflow, preparation consisted mainly of platform startup and consumable placement before execution, corresponding to approximately 20.0 s per sample under the present three-sample validation setting.

The execution-stage time included active intervention associated with dispensing, coating- and drying-related manipulation, sample transfer, and characterization. The automated workflow was repeated three times, and the corresponding timing values are reported on a per-sample basis as mean and standard deviation. The stirring stage, which lasted approximately 36000 s, did not require continuous operator intervention and was therefore reported separately as process hold time rather than being included in the total active operation time. Optical measurements were performed using a UV–Vis spectrometer at wavelengths of 365 nm, 760 nm, and 940 nm.

B. *Workflow Efficiency Comparison*

Table XIII summarizes the workflow timing results under the same accounting rule. Compared with manual processing, the automated platform required substantially less active operator intervention during preparation, dispensing, and coating-related manipulation. For the three-sample validation batch, the total active operation time was reduced from 1589.7 s for manual processing to 331.8 s for the automated workflow, corresponding to an approximately 79.1% reduction. On a per-sample basis, this corresponds to a decrease from 529.9 s to 110.6 s. From the batch-size scaling shown in Figure 7, the relative time saving gradually converges to about 72.6% as the batch size increases. The process hold time remained identical for both workflows at 36000 s and was therefore excluded from the active-operation-time reduction.

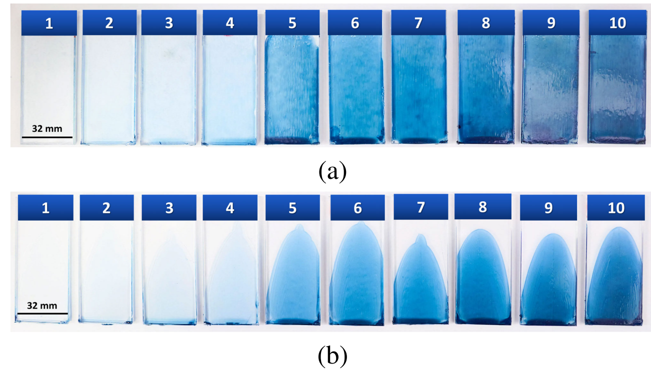


Fig. 8. Representative coating results obtained by (a) manual preparation and (b) automated platform preparation.

The reduction mainly originated from the automation of repetitive dispensing and coating-related handling steps, together with the lower preparation burden enabled by batch-based operation. The small variation across repeated automated runs further indicates good execution consistency.

C. *Coating and Optical Characterization Results*

To evaluate the effect of manual and automated workflow execution on the optical performance of the coating process, an additional formulation study was conducted by varying the infrared filler concentration in steps of 0.25 g mL^{-1} over the tested range. The formulations consisted of UV filler, IR filler, matrix, and diluent. In total, ten coated formulation conditions were investigated, together with one uncoated glass reference. For each workflow, the full set of ten coated formulations was prepared in three independent experimental runs, yielding three samples for each formulation condition and 30 coated samples for each preparation mode. Each sample was characterized at 365 nm, 760 nm, and 940 nm using a UV–Vis spectrometer. The uncoated glass substrate was measured separately as the baseline reference. Figure 8 shows representative coatings produced by manual and automated preparation, while Figure 9 presents the corresponding transmittance results.

As shown in Figure 9, both preparation modes produced the expected wavelength-dependent transmittance trends across the investigated formulation range. The uncoated glass reference exhibited the baseline optical response of the substrate. Overall, coatings prepared by the automated platform showed higher mean transmittance than those prepared manually. In addition, the automated platform generally yielded smaller standard deviations across most tested conditions, indicating a narrower spread among independently prepared samples. These results suggest that the automated approach improves coating-preparation uniformity while maintaining the intended optical response of the formulation system. Taken together, the experimental validation demonstrates that the optimized automated platform can substantially reduce active human intervention while producing coatings with higher average transmittance and smaller sample-to-sample variation than manual operation in the studied thermal-insulation coating workflow.

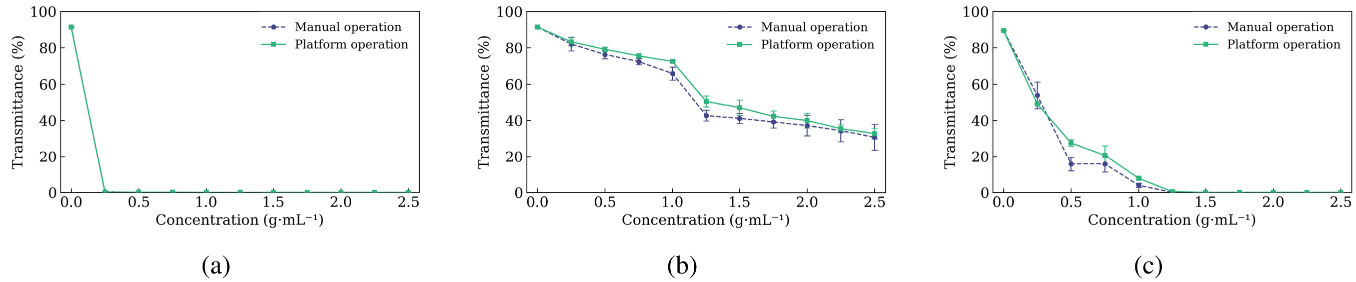


Fig. 9. Optical transmittance at (a) 365nm, (b) 760nm, and (c) 940nm. Error bars represent standard deviations.

V. Conclusion

This paper presented an experimental-procedure-driven modeling and equipment quantity configuration framework for high-throughput automated materials experiments. The framework integrates a hybrid-automata-inspired process abstraction, a structured equipment dictionary, and a procedure-to-equipment mapping mechanism to construct a unified platform model for timing and dependency analysis. Based on this model, equipment quantity configuration was formulated as a constrained integer nonlinear optimization problem under budget, footprint, and equipment-dependency constraints.

A thermal-insulation coating case study was conducted to demonstrate the proposed framework. The optimization results showed that the framework can identify high-quality feasible equipment configurations and reveal design trade-offs between sample parallelization and supporting-equipment investment. In the studied case, the optimized design favored container-based parallelization over tray-based expansion under the given resource constraints. Sensitivity analysis further indicated that the optimized configuration was robust to moderate variations in objective weights and throughput scaling, while being more strongly influenced by admissible resource budgets, especially the cost limit.

Experimental validation on a representative thermal-insulation coating workflow showed that the optimized automated platform substantially reduced active human intervention compared with manual operation. In addition, the automated workflow yielded higher average optical transmittance and smaller standard deviations across the tested formulation conditions, indicating improved preparation consistency while maintaining the intended optical response.

This work focuses on equipment quantity configuration rather than full platform layout, hardware integration, or control synthesis. Future work will extend the framework to a wider range of experimental procedures, incorporate layout-aware configuration and scheduling, and investigate integration with automated experiment orchestration and closed-loop materials discovery.

Acknowledgment

The authors acknowledge the use of AI-assisted tools for language polishing, grammar checking, and LaTeX formatting support. They are solely responsible for the scientific content, analysis, interpretations, and conclusions of this work.

References

[1] K. T. Butler, D. W. Davies, H. Cartwright, O. Isayev, and A. Walsh, “Machine learning for molecular and materials science,” *Nature*, vol. 559, no. 7715, pp. 547–555, 2018.

[2] J. Schmidt, M. R. G. Marques, S. Botti, and M. A. L. Marques, “Recent advances and applications of machine learning in solid-state materials science,” *npj Comput. Mater.*, vol. 5, no. 1, Aug. 2019, Art. no. 83.

[3] N. J. Szymanski et al., “An autonomous laboratory for the accelerated synthesis of inorganic materials,” *Nature*, vol. 624, no. 7990, pp. 86–91, Dec. 2023, doi: 10.1038/s41586-023-06734-w.

[4] A. K. Y. Low, J. J. W. Cheng, K. Hippalgaonkar, and L. W. T. Ng, “Self-driving laboratories: Translating materials science from laboratory to factory,” *ACS Omega*, vol. 10, no. 28, pp. 29902–29908, Jul. 2025, doi: 10.1021/acsomega.5c02197.

[5] D. P. Tabor et al., “Accelerating the discovery of materials for clean energy in the era of smart automation,” *Nature Rev. Mater.*, vol. 3, no. 5, pp. 5–20, Apr. 2018.

[6] G. Tom et al., “Self-driving laboratories for chemistry and materials science,” *Chem. Rev.*, vol. 124, no. 16, pp. 9633–9732, Aug. 2024, doi: 10.1021/acs.chemrev.4c00055.

[7] H. Lee, H. J. Yoo, H. S. Jang, B. Park, Y. J. Park, and S. S. Han, “Toward self-driving laboratory 2.0 for chemistry and materials discovery,” *Mater. Horizons*, vol. 13, no. 10, pp. 4712–4739, 2026, doi: 10.1039/d5mh01984b.

[8] A. A. Volk and M. Abolhasani, “Performance metrics to unleash the power of self-driving labs in chemistry and materials science,” *Nature Commun.*, vol. 15, no. 1, Feb. 2024, Art. no. 1378, doi: 10.1038/s41467-024-45569-5.

[9] P. Salazar-Villacis and B. Benyahia, “The ADePT framework for assessing autonomous laboratory robotics,” *Commun. Chem.*, vol. 9, no. 1, Feb. 2026, Art. no. 99, doi: 10.1038/s42004-026-01932-9.

[10] C. W. Coley et al., “A robotic platform for flow synthesis of organic compounds informed by AI planning,” *Science*, vol. 365, no. 6453, Aug. 2019, Art. no. eaax1566, doi: 10.1126/science.aax1566.

[11] B. P. MacLeod et al., “Self-driving laboratory for accelerated discovery of thin-film materials,” *Proc. Sci. Adv.*, vol. 6, 2020, Art. no. eaaz8867, doi: 10.1126/sciadv.aaz8867.

[12] A. E. Gongora et al., “A Bayesian experimental autonomous researcher for mechanical design,” *Sci. Adv.*, vol. 6, no. 15, Apr. 2020, Art. no. eaaz1708, doi: 10.1126/sciadv.aaz1708.

[13] Y. Jiang, D. Salley, A. Sharma, G. Keenan, M. Mullin, and L. Cronin, “An artificial intelligence enabled chemical synthesis robot for exploration and optimization of nanomaterials,” *Sci. Adv.*, vol. 8, no. 40, Oct. 2022, Art. no. eabo2626, doi: 10.1126/sciadv.abo2626.

[14] H. Zhao et al., “A robotic platform for the synthesis of colloidal nanocrystals,” *Nature Synth.*, vol. 2, no. 6, pp. 1–12, Mar. 2023, doi: 10.1038/s44160-023-00250-5.

[15] T. Ha et al., “AI-driven robotic chemist for autonomous synthesis of organic molecules,” *Sci. Adv.*, vol. 9, no. 44, Nov. 2023, Art. no. eadj0461, doi: 10.1126/sciadv.adj0461.

[16] J. Park, Y. M. Kim, S. Hong, B. Han, K. T. Nam, and Y. Jung, “Closed-loop optimization of nanoparticle synthesis enabled by robotics and machine learning,” *Matter*, vol. 6, no. 3, pp. 677–690, Mar. 2023, doi: 10.1016/j.matt.2023.01.018.

[17] B. A. Koscher et al., “Autonomous, multiproperty-driven molecular discovery: From predictions to measurements and back,” *Science*, vol. 382, no. 6677, Dec. 2023, Art. no. eadi1407, doi: 10.1126/science.adi1407.

[18] Z. Zhou, S. Veeramani, F. Munguia-Galeano, H. Fakhruldeen, and A. I. Cooper, "Localization, inspection, and reasoning (LIRA) module for autonomous workflows in self-driving laboratories," *Commun. Chem.*, vol. 8, no. 1, Nov. 2025, Art. no. 384, doi: 10.1038/s42004-025-01770-1.

[19] J. Huang, H. Liu, S. Junginger, and K. Thurow, "Mobile robots in automated laboratory workflows," *SLAS Technol.*, vol. 30, Feb. 2025, Art. no. 100240.

[20] Q. Zhao, X. Zhang, and P. Wang, "Multi-type equipment selection and quantity decision optimization in intelligent warehouse," *IEEE Access*, vol. 12, pp. 63515–63527, 2024.

[21] J. Zhou et al., "A multi-robot–multi-task scheduling system for autonomous chemistry laboratories," *Digit. Discovery*, vol. 4, no. 3, pp. 636–652, 2025, doi: 10.1039/d4dd00313f.

[22] L. Lu, Y. Shi, M. Pan, and X. Li, "Verifying hybrid automata networks guided by task scenarios," *Formal Methods Syst. Design*, vol. 68, no. 1, Feb. 2026, Art. no. 5, doi: 10.1007/s10703-026-00492-x.

[23] Y. Zhang and M. Zhang, "Optimization of machine tool processing scheduling based on differential evolution algorithm," *PLoS ONE*, vol. 20, no. 10, Oct. 2025, Art. no. e0333691.

[24] O. A. Alvarez-Flores, R. Rivera-Blas, L. A. Flores-Herrera, E. Z. Rivera-Blas, M. A. Funes-Lora, and P. A. Niño-Suárez, "A novel modified discrete differential evolution algorithm to solve the operations sequencing problem in CAPP systems," *Mathematics*, vol. 12, no. 12, p. 1846, Jun. 2024.

[25] J.-F. Raskin, "An introduction to hybrid automata," in *Handbook of Networked and Embedded Control Systems*. Cham, Switzerland: Springer, 2005, pp. 491–517.

[26] P. Belotti, C. Kirches, S. Leyffer, J. Linderoth, J. Luedtke, and A. Mahajan, "Mixed-integer nonlinear optimization," *Acta Numerica*, vol. 22, pp. 1–131, May 2013.

[27] R. Storn and K. Price, "Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optim.*, vol. 11, no. 4, pp. 341–359, Dec. 1997, doi: 10.1023/a:1008202821328.

[28] D. Salvagnin, R. Roberti, and M. Fischetti, "A fix-propagate-repair heuristic for mixed integer programming," *Math. Program. Comput.*, vol. 17, no. 1, pp. 111–139, Mar. 2025, doi: 10.1007/s12532-024-00269-5.

Guoxiong Ma received the B.S. degree in mechanical engineering and computer science from North Carolina State University, USA, in 2020, and the M.Sc. degree in mechanical engineering from The Hong Kong University of Science and Technology (HKUST), Hong Kong, in 2022, where he is currently pursuing the Ph.D. degree in mechanical and aerospace engineering. His research focuses on artificial intelligence for mechanical systems, smart manufacturing, robotics, and automation.

Haozhe Cui received the B.Eng. degree in electronic and computer engineering and the M.Phil. degree in individualized interdisciplinary program from The Hong Kong University of Science and Technology (HKUST), in 2023 and 2025, respectively. His research interests include reinforcement learning, robotics, and adaptive control for intelligent systems in dynamic environments.

Chun Yin Yip received the B.Eng. and M.Phil. degrees in mechanical engineering from The Hong Kong University of Science and Technology (HKUST), Hong Kong, in 2023 and 2025, respectively. His research focuses on automation and robotics, including automated platforms for material synthesis, soft robotics, and bio-inspired robots.

Weibin Zhang received the B.Eng. degree from Zhengzhou University, China, in 2018, and the M.Eng. and Ph.D. degrees in mechanical engineering from The Hong Kong University of Science and Technology (HKUST), in 2019 and 2026, respectively. His research focuses on transparent solar radiation management coatings for windows.

Molong Duan (Member, IEEE) received the B.S. degree in theoretical and applied mechanics from Peking University, Beijing, China, in 2012, and the M.S. and Ph.D. degrees in mechanical engineering from the University of Michigan, Ann Arbor, MI, USA, in 2014 and 2018, respectively. He received the Rackham Centennial Fellowship from the University of Michigan in 2013. He is currently an Assistant Professor with the Department of Mechanical and Aerospace Engineering, The Hong Kong University of Science and Technology, Hong Kong, SAR, China. His research interests include physics-based and data-driven modeling and control of dynamic systems, robotic manufacturing vibration compensation, additive manufacturing control, continuous fiber additive manufacturing, flexible aircraft modeling and control, and vertical take-off and landing (VTOL) aircraft modeling and control.

Jinglei Yang received the B.S. degree from the University of Science and Technology of China in 1999, the Dr.-Ing. degree from the Technical University of Kaiserslautern Germany in 2006, and the M.S. degree from the University of Science and Technology of China in 2022. Currently, he is a Full Professor with the Department of Mechanical and Aerospace Engineering, The Hong Kong University of Science and Technology. His research interests include interfacial materials science, sustainable and green manufacturing, and laboratory intelligence with AI+robotics-enabled autonomous high throughput methods to discover novel materials. He is the Chinese Associate Editor of *Composites Part A: Applied Science and Manufacturing*.