Bayesian Optimization for the Vehicle Dwelling Policy in a Semiconductor Wafer Fab

Bonggwon Kang, Chiwoo Park, *Senior Member, IEEE*, Haejoong Kim, and Soondo Hong

***Abstract*— Many semiconductor fabrication plants (fabs) prefer simulation-based decision making for vehicle dwelling policies because it can capture a fab's scalability and complexity. Vehicle dwelling policies assign idle vehicles to intra-bay and outer loops in automated material handling systems (AMHSs) to respond quickly to transportation demands. Fabs are motivated to control vehicle dwelling policies when fabs experience significant fluctuations, i.e., changes in product mix. Fab operators evaluate manually designed candidate solutions because it is time-intensive to run a large-scale simulation with numerous potential solutions. To determine a vehicle dwelling policy, we propose a simulation optimization approach based on Bayesian optimization (BO) with class-based clustering. BO adaptively traces efficient vehicle dwelling policies based on a surrogate model and an acquisition function. Class-based clustering alleviates the high dimensionality of the design space by grouping bays into a small number of classes. By striking a balance between the complexity of the design space and the quality of the solutions, our proposed policy significantly reduces the number of simulation runs required to determine efficient vehicle dwelling policies. We conclude that BO with class-based clustering is more advantageous than using a genetic algorithm (GA) and using heuristics.**

***Note to Practitioners*— This study is motivated by the difficulties in simulation-based decision making for optimal vehicle dwelling policies in a semiconductor wafer fab's AMHS. While existing research has demonstrated the effectiveness of simulation analysis on operational planning in a fab, simulation optimization for a vehicle dwelling policy is still problematical due to the heavy computation burden for simulation runs and its large design space with the increased number of control variables in a fab. Therefore, we develop a simulation optimization approach using BO with class-based clustering. The proposed approach significantly reduces the number of simulation runs to obtain efficient vehicle dwelling policies, resulting in decreased delivery times and vehicle utilization rates.**

***Index Terms*— Automated material handling system, semiconductor fab, vehicle dwelling policy, simulation, Bayesian optimization.**

I. Introduction

AUTOMATED material handling systems (AMHSs) automate and optimize material flows in manufacturing facilities, using process and material route information from a facility's manufacturing information system. The semiconductor wafer manufacturing industry uses AMHSs to transfer and store wafers between fabrication steps because they can achieve high delivery speeds and operational reliability [1], [2], [3]. The industry prefers a vehicle-based AMHS with an overhead hoist transport (OHT) that travels on guided paths and transfers front-opening unified pods (FOUPs) to temporary storages and stockers in bays [4]. **Fig. 1** illustrates a conceptual layout of an AMHS in a 300mm semiconductor fabrication plant (fab).

Typically, large wafer fabs are partitioned into multiple bays, and hundreds of OHTs are assigned to transfer the wafers between the bays via internal routes. Unassigned OHTs generally circulate through the internal route of a designated bay. To respond quickly to transportation demands, fabs try to maintain a minimum number of idle OHTs within each bay. Since too many OHTs in a bay could lead to a vehicle shortage from other bays, fab operators need to control the minimum and the maximum number of idle OHTs for each bay.

A black-box problem, also termed derivative-free optimization, is identified by a partial or total lack of closed-form equations with constraints and an objective. Since the AMHS performance under a vehicle dwelling policy represents a black-box function due to the unknown interactions among vehicles and machines, simulation analysis has been used to evaluate AMHS performance under high uncertainties and achieve optimal policies [5], [6]. Many fabs prefer simulation-based decision making for vehicle dwelling policies because it can capture a fab's scalability and complexity, and fabs can control vehicle dwelling policies when they experience significant fluctuations.

Manuscript received 27 March 2023; revised 4 July 2023; accepted 18 September 2023. This article was recommended for publication by Associate Editor H.-J. Kim and Editor L. Moench upon evaluation of the reviewers' comments. The work of Bonggwon Kang was supported by the 2022 BK21 FOUR Graduate School Innovation Support funded by Pusan National University (PNU-Fellowship Program). The work of Chiwoo Park was supported in part by the Brain Pool Fellowship of the National Research Foundation Korea under Grant 2019H1D3A2A01100649 and in part by the National Science Foundation under Grant 2152655. The work of Haejoong Kim was supported by the Technology Innovation Program (Development of AI-based manufacturing process logistics optimization technology that supports operation optimization for each process situation) funded by the Ministry of Trade, Industry & Energy (MOTIE, South Korea) under Grant 20019178. The work of Soondo Hong was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean Government (MSIT) under Grant NRF-2020R1A2C2004320. *(Corresponding author: Soondo Hong.)*

Bonggwon Kang and Soondo Hong are with the Industrial Data Science and Engineering, Department of Industrial Engineering, Pusan National University, Busan 46241, South Korea (e-mail: bonggwon.kang@gmail.com; soondo.hong@pusan.ac.kr).

Chiwoo Park is with the Department of Industrial and Manufacturing Engineering, Florida State University, Tallahassee, FL 32310 USA, and also with the Department of Industrial and Systems Engineering, University of Washington, Seattle, WA 98195 USA (e-mail: cpark5@fsu.edu).

Haejoong Kim is with the Department of Industrial and Management Engineering, Kyonggi University, Suwon, Gyeonggi 16227, South Korea (e-mail: haejoong.kim@kyonggi.ac.kr).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TASE.2023.3320189.

Digital Object Identifier 10.1109/TASE.2023.3320189

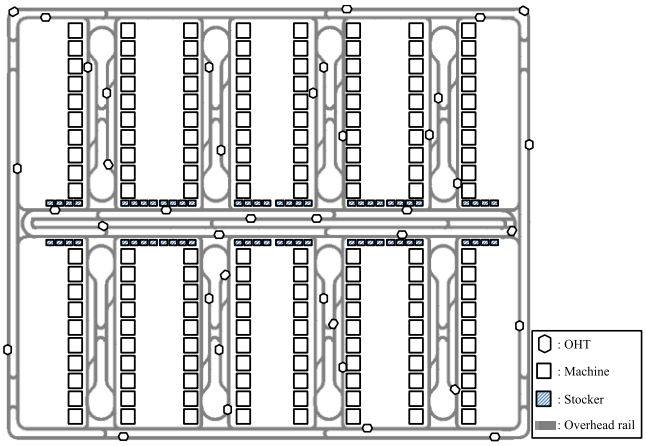

Fig. 1. Conceptual layout of an AMHS in a 300mm fab.

However, a high fidelity of simulation with the increased number of vehicles, machines, and intra-bays is computationally expensive and often requires several hours for a single run. Further, the increase in the number of control variables requires extensive simulation runs to obtain optimal solutions (e.g., in a d-dimensional design space, an n-point discretization of each dimension provides n^d candidate solutions to be explored). As a result, many fab operators evaluate manually designed candidates within a limited simulation budget that may not provide optimal solutions [7].

In this study, we propose a simulation optimization approach based on Bayesian optimization (BO) with class-based clustering to determine the vehicle dwelling policy within a limited number of simulation runs. Bayesian optimization (BO) has proven to be an efficient alternative to simulation optimization [8] because it can replace exhaustive simulation runs with the predictions from a surrogate model based on the outcomes from past simulation runs. We study an application-specific BO for optimizing a vehicle dwelling policy when the dimensionality of the design space increases proportionally to the number of bays. To reduce the dimensionality, we group the bays into a small number of classes by similarity and determine the same number of dwelling limits for the same class. Our class-based clustering approach strikes a balance between the complexity of the design space and the quality of the solutions.

The remainder of this study is organized as follows. Section II reviews the related literature and the associated challenges. Section III discusses the vehicle dwelling policy and the minimum and maximum number of vehicles required for each dwell point. Section IV summarizes our simulation optimization approach based on BO with class-based clustering to optimize the vehicle dwelling policy in an AMHS. Section V provides the details of the simulation experiments and discusses the results. Section VI concludes and provides suggestions for future research.

II. Literature Review

A vehicle dwelling policy assigns vehicles to specific dwell points so they can respond quickly to transfer requests [9]. The policy's objective is to reduce delivery time by minimizing lot waiting time, empty travel time, loaded travel time, and blocking time simultaneously. Lot waiting time corresponds to the time a lot must wait for an available vehicle that initiates the transfer request at an origin machine. Empty travel time corresponds to the interval between the time a vehicle is initiated and the time it starts to load a lot at the origin port. Loaded travel time corresponds to the interval between the time a vehicle finishes loading at an origin port and the time it starts to unload at a destination port. Blocking time corresponds to the time a vehicle waits for other vehicles to complete their loading/unloading operations.

Various analytical approaches have been considered to evaluate and improve the performance of vehicle-based systems. Maxwell and Muckstadt [10] proposed an analytical methodology to determine the number of required vehicles, their routes, and the total number of travels over routes. Egbelu [11] addressed the problem of selecting dwell points holding vehicles in a single loop-type automated guided network and proposed an analytical model. Hu and Egbelu [12] developed two models with different objectives for determining dwell points to minimize maximum and mean response times in a guided path system. Bruno et al. [13] introduced linear heuristics to dynamically determine the dwell points of automated guided vehicles in a material handling system. Nishida and Nishi [14] presented an integer programming model and a heuristic for a conflict-free routing problem based on a just-in-time delivery concept, and investigated the relationships between delivery completion times and tardiness/earliness penalties under dynamic task arrivals. Nazzal and McGinnis [15] developed an analytical model to estimate the blocking time in an AMHS under the assumption that the sum of the lot waiting time and empty travel time was independent of the lot's original location. Baykasoğlu et al. [16] presented a novel integer programming model for freight planning, fleet composition, fleet allocation, idle vehicle repositions, and fleet outsourcing problems. Nazzal et al. [17] investigated a queuing-based analytical model to estimate the performance of a conveyor-based AMHS in a fab; experiments under the Poisson assumption indicated that the analytical model performed well, with an average error of 4.2%, but showed relatively high errors under heavy traffic.

Simulation analysis provides important details for real-time operational control of system dynamics, i.e., vehicle downtime and product mix changes, in fabs [18], [19]. Ahn and Park [20] proposed a cooperative zone-based rebalancing algorithm based on multi-agent reinforcement learning to dynamically reallocate vehicles in a fab; experiments indicated that the real-time control algorithm significantly reduced the empty travel time and vehicle utilization rate. Hwang and Jang [21] investigated a reinforcement learning-based dynamic routing algorithm to avoid time-variant vehicle congestion in a fab's AMHS. Ahn et al. [22] suggested a dynamic routing algorithm using a sequence of graphs to consider historic traffic information in a fab's AMHS; experiments indicated that the proposed model dynamically rerouted vehicles to find efficient paths. Lee et al. [23], who addressed vehicle repositioning based on multi-task learning that considered traffic patterns in multiple areas, demonstrated that a robust high-accuracy performance could be obtained under dynamic variability.

Operational planning approaches for vehicle management in fabs have examined system reliability and efficiency [19]. The approaches have the merit of model validity to provide explicit relationships between input and output variables for strategic management. Lee et al. [24] developed a routing algorithm using novel congestion metrics to identify the systematic bottlenecks in an actual fab. The results of their simulation indicated that the proposed algorithm reduced the delivery time and saved the loss in production. Kim and Park [25] showed that vehicle dwelling policies significantly impacted AMHS productivity in various simulation scenarios. Kiba et al. [26] discussed a vehicle dwelling policy with the potential to improve AMHS performance, considering the minimum and maximum number of required idle vehicles for each dwell point. Lin et al. [27] investigated the minimum and maximum number of vehicles required for each dwell point by designing and evaluating several candidate solutions for their simulation study.

Simulation optimization approaches have been proposed for finding optimal input variables for complex and stochastic functions within limited computation budgets. Lin and Huang [28], who studied optimizing combinatorial dispatching rules due to the limited number of simulation runs, proposed a genetic algorithm-based simulation optimization to determine the combination dealing with system dynamics in a fab's AMHS. Chang et al. [29], who proposed a genetic algorithm-based simulation optimization to find the optimal combination of vehicle dispatching rules, analyzed the transfer delays with the increased number of simulation runs. Lee et al. [30] presented a space-filling design method with a classification model to accelerate the simulation optimization of combinatorial dispatching rules in an LCD manufacturing system. Huang et al. [31], who suggested a mathematically formulated model for determining the optimal number of vehicles for dwell points, employed a simulation optimization approach to iteratively trace optimal solutions via a discrete-event simulation. The experimental results confirmed that their approach increased throughput with the increased number of simulation runs. Chang et al. [32] proposed a sequential surrogate to determine the number of required vehicles for an AMHS. Their surrogate model, which was based on the quadratic relationship between the input and output variables of an objective function, employed a sampling scheme to enhance the solution quality with fewer evaluations. Kuo et al. [33] investigated a neural network (NN)-based surrogate model with a uniform design to expedite decision making in a fab's AMHS; the results indicated that the model estimated simulation outcomes within a relatively small number of training data. Can and Heavey [34] provided a comparative analysis of genetic programming- and NN-based surrogate models to predict the inventory cost of a fab's AMHS. Even though the genetic programming-based surrogate model required more computing resources, it was still competitive in model validity because it produced explicit functions for strategic advantage in operational management. Although the alternative approaches can handle a wide range of problems, they often require larger numbers of simulation runs and long computation times, both of which become troublesome bottlenecks in decision making.

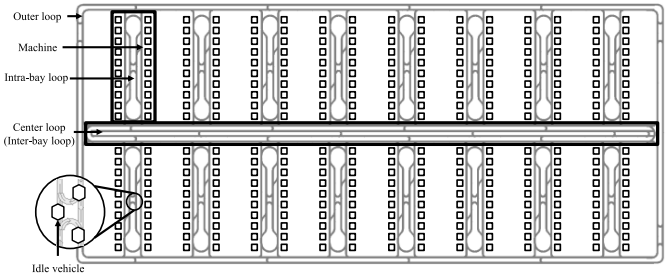

Fig. 2. AMHS layout of the wafer fab.

BO approaches with Gaussian process (GP) are suitable for solving black-box objective functions in cases with a limited number of evaluations. Since GP is a non-linear and flexible surrogate with uncertainty quantification, the BO adaptively and deliberately samples the next design point considering the exploration and exploitation of the design space [35], [36]. Candelieri et al. [37] solved the pump operation scheduling problem in water distribution systems based on BO with a limited number of evaluations. Hickish et al. [38] utilized BO to find an acceptable solution in rail network operations. Kang et al. [39] proposed BO to optimize collaborative operations between twin yard cranes in an automated container terminal; the results indicated that BO accounted for complex and uncertain factors in the terminal.

Most studies have applied BO to problems with moderate dimensions of up to 10 [40]. Although the BO approach is known as a powerful scheme for optimization with limited evaluations, it still suffers from high dimensionality [35], [41], [42], [43], [44]. To relax BO complexity and speed up convergence, Salem et al. [45] proposed a sequential dimension reduction method by identifying the influential input variables. Chen and Liao [46] proposed a dimension reduction method that decomposed high dimensionality into a union of low dimensionality, and built a surrogate model with a small amount of training data.

This study contributes to the published literature as follows: (1) We investigate the optimization of the minimum and the maximum numbers of vehicles for an entire AMHS, rather than using manually designed candidate solutions; (2) We expedite simulation-based decision making for vehicle operations by constructing a surrogate model and sampling schema using BO; and (3) We show how BO with class-based clustering is able to balance between the complexity of the design space and the quality of the solutions.

III. Problem Definition

A. AHMS Layout With Intra-Bay and Outer Loops

In this study, we investigate a hypothetical 300mm fab. The AMHS layout has 16 intra-bays, intra-bay loops, and inter-bay and outer loops as shown in **Fig. 2.** Each intra-bay consists of machines that perform one of the eight processes (such as photolithography, etching, and diffusion). The inter-bay loop

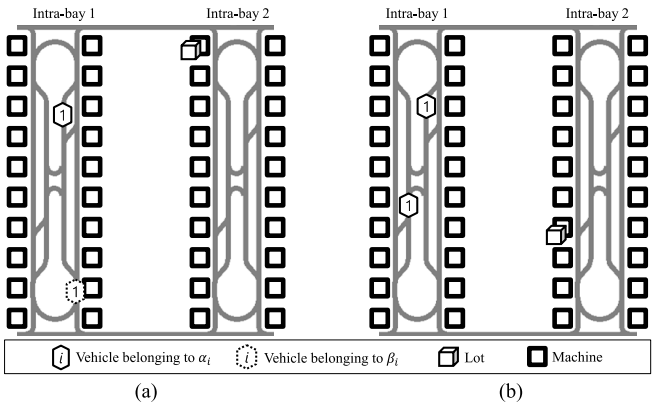

Fig. 3. Examples of vehicle initiation under different vehicle dwelling conditions: (a) vehicle initiating the incoming lot; (b) vehicle shortage for the incoming lot.

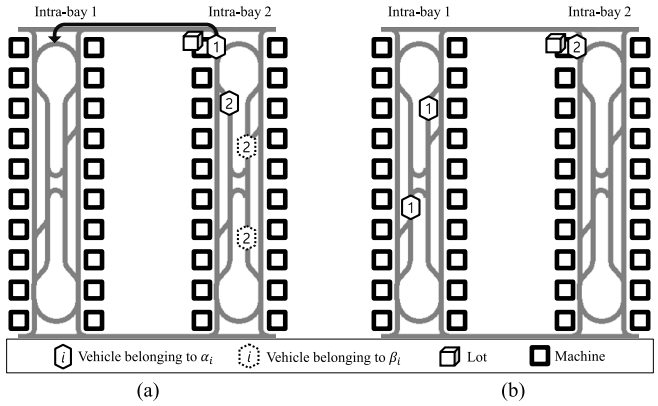

Fig. 4. Examples of vehicle movements under different vehicle dwelling conditions: (a) vehicle redistribution to the other intra-bay; (b) vehicle circulation in the intra-bay where the vehicle completed its transfer request.

connecting with the intra-bays allows direct delivery by a single vehicle. The outer loops unite the intra-bays and the inter-bay. All loops are unidirectional. An intra-bay has two dwelling loops for the vehicles. The wafer fab uses a vehicle dwelling policy to manage the number of vehicles staying in an intra-bay. The vehicle dwelling policy distributes idle vehicles over the inter-bay and the outer loops. A distributed vehicle travels on its designated dwelling point until a transfer request initiates the vehicle.

B. Intra-Bay Dwelling Policy With the Minimum and Maximum Number of Required Idle Vehicles

When a vehicle becomes idle at an intra-bay after a transfer, all intra-bays in order of nearest to the intra-bay where the vehicle completed its transfer, check their minimum and maximum bounds. If there is room for the vehicle below the minimum bound, the intra-bay keeps the vehicle ready in the intra-bay with an exclusive priority. Transfer requests from other intra-bays cannot initiate the vehicle belonging to the minimum number of required vehicles for the intra-bay. If all intra-bays have idle vehicles over the minimum bounds and there is still room for the vehicle under the maximum bound, the intra-bay keeps the vehicle ready in the intra-bay without an exclusive priority. If the maximum bounds of all of the intra-bay loops are filled, the vehicle moves to the outer loop. We use α_i and β_i to denote the minimum and maximum number of idle vehicles, respectively, required for intra-bay i. We define the following parameters, sets, and decision variables.

Parameters and Sets

I, i: Set of bays and its index $i \in I$.

Decision variables

α_i :Minimum number of required idle vehicles in intra-bay i.

β_i :Maximum number of required idle vehicles in intra-bay i.

The vehicle dwelling policy determines the vehicle's distribution over the intra-bays. **Fig. 3**(a) and (b) are examples of vehicle initiation under different vehicle dwelling conditions. In **Fig. 3**(a), the transfer request generated from intra-bay 2 cannot initiate the vehicle belonging to α_1, but it can initiate the one belonging to β_1. In **Fig. 3**(b), although intra-bay 1 keeps a sufficient number of idle vehicles, intra-bay 1 does not distribute the vehicles to intra-bay 2 to quickly respond to upcoming requests from intra-bay 1. The transfer request generated from intra-bay 2 cannot initiate the vehicle belonging to α_1, so the transfer request must wait until to be initiated by another vehicle belonging to α_2, β_2, or β_1.

Fig. 4(a) and (b) are examples of vehicle movements under different vehicle dwelling conditions. In **Fig. 4**(a), intra-bay 2 redistributes the vehicle as soon as the lot is unloaded to avoid vehicle shortage from an upcoming lot requested from intra-bay 1. In **Fig. 4**(b), as soon as the vehicle unloads its lot at intra-bay 2, it circulates in intra-bay 2 for an upcoming lot requested from intra-bay 2.

IV. Simulation Optimization for the Vehicle Dwelling Policy

Wafer fabs need to determine an optimal vehicle dwelling policy within a limited planning period. While a determination of the minimum/maximum bounds for each intra-bay could improve delivery time, the increased dimensionality of the solution space requires an exponentially increased number of simulations to guarantee the optimal vehicle dwelling policy. This section explains how to address the problem of high dimensionality when using BO to predict the simulation outcomes of the vehicle dwelling policy and determine the minimum and maximum number of vehicles in all intra-bays.

A. Class-Based Clustering for Relaxing High Dimensionality

We begin by grouping the intra-bays into classes according to similar transportation demands. Inspired by inventory and storage management studies, we call it class-based clustering [47], [48]. The practitioners can estimate upcoming transportation demands for the eight fabrication steps [2], [15], [49]. Next, we set the classes' minimum/maximum bounds based on the estimated transportation demands.

Both α_c and β_c denote the minimum and the maximum numbers of idle vehicles, respectively, required for the intra-bays in class c. The impact of α_c and β_c on intra-bay i in class c, i.e.,

Process	Stationary probability	Class	α_i	β_i
CMP	0.04	Class 1	1	1
METAL	0.05			
IMP	0.09			
CVD	0.10			
DIFF	0.11	Class 2	2	2
PHOTO	0.15			
ETCH	0.22			
CLEAN	0.24			

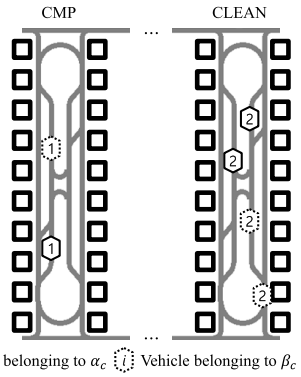

Fig. 5. Example of a dwelling condition with decision variables α_c and β_c based on the class-based clustering with $|C| = 2$.

$i \in I_c$, is highly sensitive to the demand patterns of the transfer requests. The demand patterns are determined by the stationary and transaction matrices generated from a production sequence [15], [49]. To manage α_c and β_c, we sort them in order of demand and assign them to different classes, e.g., given the number of classes $|C|$, where the number of bays $(B) \gg |C|$, we assign intra-bay i to class I_c. We set the upper bound of the number of required idle vehicles based on an expected vehicle utilization rate and the total number of vehicles. In this study, we set u as 0.85. We define the following parameters, sets, decision variables, and objectives.

Parameters and Sets

C, c: Set of classes and its index c, $\bigcup_{c \in C} I_c = I$, $\bigcap_{c \in C} I_c = \emptyset$.

P, p_i: Set of stationary probabilities and its element.

Q, q_c: Set of $|C|$-quantiles with boundaries and its element.

u: Expected vehicle utilization rate, $0 < u \leq 1$.

N: Number of vehicles.

Decision variables

α_c : Minimum number of required idle vehicles per intra-bay in class c.

β_c : Maximum number of required idle vehicles per intra-bay in class c.

Objectives

We aim to minimize the delivery time such that

$0 \leq \alpha_c \leq \lceil N(1-u) \rceil$, for all c.

$0 \leq \beta_c \leq \lceil N(1-u) \rceil$, for all c.

$I_c = \{i : q_{c-1} \leq p_i < q_c, i \in I\}$, for all c.

$q_0 = 0, q_{c+1} = 1$.

Fig. 5 shows a dwelling condition when decision variables (α_1,β_1) and (α_2,β_2) are (1, 1) and (2,2), respectively. The intra-bays in Class 2 with the higher demand have higher minimum and maximum number of required idle vehicles than the intra-bays in Class 1 with the lower demand. $|C|$, which is the number of classes, determines the number of feasible solutions as $\lceil N(1-u) \rceil^{2|C|}$. When the number of classes is two, the number of feasible solutions is $\lceil N(1-u) \rceil^4$ with decision variables (α_c,β_c) identified as (α_1,β_1) and (α_2,β_2).

B. Simulation Modelling for the Fab's AMHS

We build a simulation model with Plant Simulation software that provides basic objects, to build the relationship of a hierarchy and inheritance between the objects, and use the built-in programming language to implement the operational policies in a simulated wafer fab after Bartlett et al. [49] and Bartlett [50]. The simulation considers realistic vehicle traffic patterns, operation policies, and other input parameters. We consider an AMHS with 200 vehicles, 16 intra-bays, and 1,100 loading/unloading stations. The maximum velocity of vehicles on straight and curved rails is 3m/s and 1m/s, respectively. We account for vehicle sizes and acceleration at 50cm and 3m/s^2.

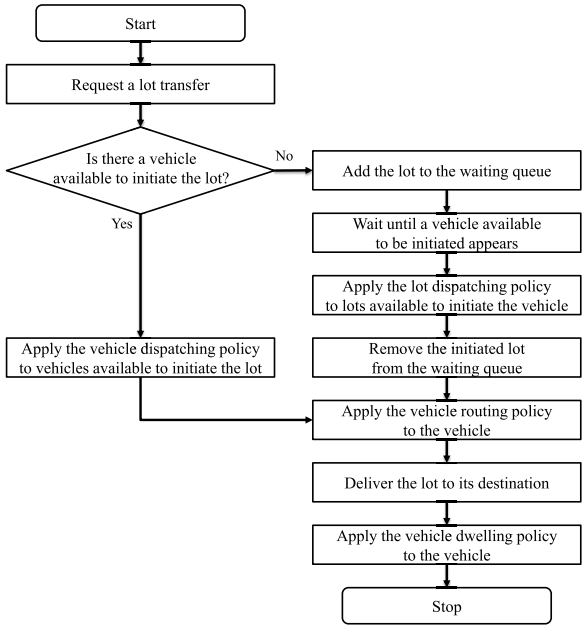

Fig. 6. Discrete-event modelling for our AMHS simulation.

The fab uses vehicle dispatching, lot dispatching, and vehicle routing. The vehicle dispatching determines which vehicle among those available initiates a requested lot. We dispatch the nearest idle vehicle to a requested lot. If no vehicle is available, the lot will be added to a waiting queue. The lot dispatching chooses which lot among those in the waiting queue initiates an available vehicle. We dispatch the closest lot to a vehicle available to be initiated. The vehicle routing determines the routes of vehicles from current to destination locations. We guide the shortest-travel-distance route for an initiated vehicle.

We consider the following assumptions: (1) The production schedule corresponds to the loading/unloading operations, which require 10 s deterministically; (2) The inter-arrival time between transfer requests follows an exponential distribution (mean 0.9s); and (3) The stationary and transaction matrices from the production sequence data of a prototype product determine the origin and destination of a transfer request. **Fig. 6** shows the flowchart of our discrete-event modelling.

To identify the major performance measurements, a vehicle utilization rate and delivery time, based on the sum of a lot waiting time, empty travel time, loaded travel time, and blocking time, we run a few trials to validate the simulation model and determine a warm-up period. Next, we run each simulation for 12 hours including the warm-up period of 6 hours. We note that the statistics vary slightly in our simulation model.

C. Bayesian Optimization With the Class-Based Clustering

BO consists of a surrogate function and an acquisition function. Applications of surrogate functions have proven to reduce computational costs [51]. As a surrogate function, we use GP suitable to consider flexible covariances with uncertainty quantification [52]. The GP approximates the full covariance over all possible solutions with a small set of data based on a kernel function. The GP, which is characterized by a Gaussian distribution with mean and covariance functions, draws a posterior distribution of the delivery time according to the minimum and maximum number of vehicles required for each dwell point.

With a limited number of trials, the acquisition function adaptively decides where to sample; we note that some studies have used the alternative approaches EI, knowledge-gradient, and entropy search [35], [53]. We employ the EI as an acquisition function for optimizing an underlying function. We took the negative of the objective value to represent a benefit required to be minimized. The EI evaluates the amount of expected improvement for the potential solutions based on a posterior distribution from the GP and recommends the next solution with the maximum expected improvement.

We let $x \in \mathbb{R}^{2|C|}$ represent a $2|C|$-dimensional vector as the simulation input variables $\{\alpha_1, \ldots, \alpha_c, \beta_1, \ldots, \beta_c\}$, and $f(x)$ represents an unknown response of the simulator described in Section IV-B for a given input x. We model $f(x)$ as GP with mean function m and covariance function k as

$$f(x) \sim GP\big(m(x), k\big(x, x'\big)\big). \tag{1}$$

The kernel function $k\big(x_i, x_j\big)$ uses the prior distribution to be inferred for posterior. We use the RBF kernel function k, the most widely used covariance function,

$$k\big(x_i, x_j\big) = \theta \exp\left(\frac{\left|\left|x_i - x_j\right|\right|^2}{2l^2}\right) + \sigma^2 I, \tag{2}$$

where amplitude parameter θ represents the overall variance, length scale parameter l controls the smoothness of the posterior probability distribution, and the noise term σ^2 represents the noise variance.

Given the data $D_n = \{x_{1:n}, f_{1:n}\}$, where $x_{1:n} = x_{1,\ldots,}x_n$ and $f_{1:n} = f(x_1)_{,\ldots,} f(x_n)$, the kernel matrix K represents a multivariate Gaussian distribution. We express the joint distribution of Gaussian processes, $f_{1:n}$ and $f_{n+1,}$ as

$$\begin{bmatrix} f_{1:n} \\ f_{n+1} \end{bmatrix} \sim N\left(0, \begin{bmatrix} K & k \\ k^T & k(x_{n+1}, x_{n+1}) \end{bmatrix}\right), \tag{3}$$

where

$$K = \begin{pmatrix} k(x_1, x_1) & \cdots & k(x_1, x_n) \\ \vdots & \ddots & \vdots \\ k(x_n, x_1) & \cdots & k(x_n, x_n) \end{pmatrix} + \sigma^2 I, \tag{4}$$

$$k = \big[k(x_{n+1}, x_1), k(x_{n+1}, x_2), \ldots, k(x_{n+1}, x_n)\big]. \tag{5}$$

We obtain the posterior probability distribution with the mean and standard deviation of the unobserved x_{n+1} using Bayesian statistics [52] as

$$P(f(x_{n+1}) \mid D_{1:n}) = N\big(\mu(x_{n+1}), \sigma^2(x_{n+1})\big), \tag{6}$$

$$u(x_{n+1}) = k^T K^{-1}\{y_1, \ldots, y_n\}, \tag{7}$$

$$\sigma^2(x_{n+1}) = k(x_{n+1}, x_{n+1}) - k^T K^{-1} k. \tag{8}$$

The EI evaluates the amount of expected improvement over the black-box objective function and chooses the next sample point x^*_{n+1}, considering the trade-off between exploitation and exploration of the search space [54] as

$$x^*_{n+1} = \text{argmax}_{x \in \chi} \text{EI}(x). \tag{9}$$

Exploitation samples the next point with a high objective value and exploration samples the next point with high uncertainty. We let γ be a hyperparameter controlling the trade-off between exploration and exploitation, y^+ be the current best solution, and ϕ and Φ be the PDF and CDF, respectively, of the standard normal distribution Z.Then,

$$\text{EI}(x) = \begin{cases} \big(u(x) - y^+ - \gamma\big)\Phi(Z) + \sigma(x)\phi(Z) & \text{if } \sigma(X) > 0, \\ 0 & \text{if } \sigma(X) = 0, \end{cases} \tag{10}$$

$$Z = \begin{cases} \dfrac{u(x) - y^+ - \gamma}{\sigma(x)} & \text{if } \sigma(X) > 0, \\ 0 & \text{if } \sigma(X) = 0. \end{cases} \tag{11}$$

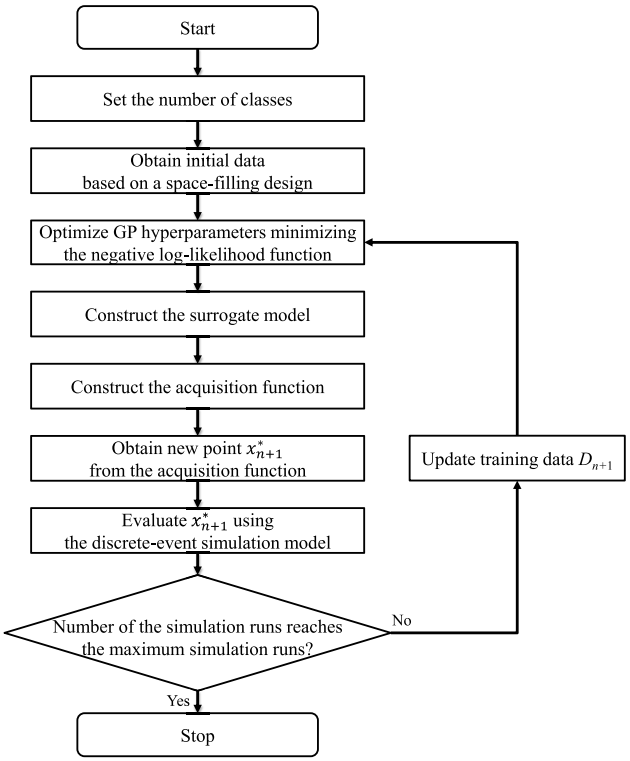

Fig. 7. Procedure of the proposed simulation optimization.

Fig. 7 shows the procedure of our proposed approach. The procedure optimizes the GP hyperparameters given data, D_n, and builds a surrogate model of the objective function using Bayesian statistics. The acquisition function recommends the next point, and the discrete-event simulation evaluates it and updates the observed data D_{n+1} as D_n. This process repeats

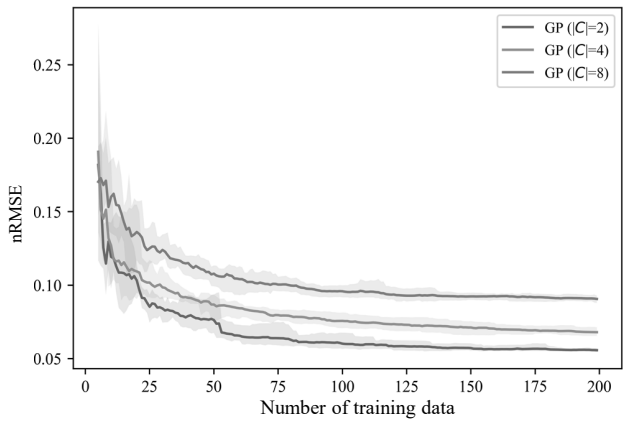

Fig. 8. nRMSE with the increased number of training data.

until the number of simulation evaluations reaches the maximum number of simulation runs.

V. Experiment and Results

In this section, we investigate the alternatives for determining the optimal vehicle dwelling policy. We run all simulations on a personal computer (PC) equipped with Windows 10, Intel Core i5, 2.7-GHz, 12 GB RAM, and 64-bit using Siemens Tecnomatix 12.0.

A. Accuracy of the GP Surrogate Model

An accurate prediction of a surrogate model expedites the convergence of optimization. We obtain 1000 data from Latin Hypercube Design (LHD) with three classes ($|C|$ = 2, 4, and 8), respectively. We use k-fold cross-validation (k= 5) to evaluate the accuracy. **Fig. 8** shows the ranges of the normalized root mean squared error (nRMSE) with the increased number of training data for the three classes. When the dimensionality of the input variables increases, the number of required simulation runs to converge also increases. The nRMSE converges to 0.05, 0.07, and 0.09 with the class-based clusterings $|C|$ = 2, 4, and 8, respectively, with 200 training data. We note that the GP with the high dimensionality of the input variables is data-hungry and shows high prediction errors.

B. Comparison With Simulation Optimization Alternatives

Since the performances of vehicle dwelling policies vary depending on vehicle utilization rate and vehicle congestion, we evaluate the robustness of our proposed approach with 190, 200, and 210 numbers of vehicles. We use the cumulative best delivery time to confirm which simulation optimization alternatives obtain an efficient vehicle dwelling policy with a limited number of simulation runs. We repeat each simulation optimization five times.

We design a GA-based simulation optimization with the class-based clustering $|C|$ = 8 modified from Pinedo [55]. Appendix A details the GA-based simulation optimization. We set the maximum number of individuals in a population as (P) = 10, the maximum number of generations as (G) = 45, and the probability of mutations as 0.05. We set the simulation budget for the GA-based simulation optimization as the total number of simulation runs including the initial population and all generations. Since each generation requires two simulation runs to evaluate offspring, the total number of simulation runs for the GA-based simulation optimization is 100.

Due to slight variations in prediction errors over 100 training data (See **Fig. 8**), we set the maximum number of simulation runs as 100 including initial data and all iterations for the BO-based simulation optimizations with the class-based clusterings $|C|$ = 2, 4, and 8. We use LHD to obtain the initial data with the number of input variables, and use the *scikit-learn* Python library to model the GP and optimize its hyperparameters of the kernel function using L-BFGS-B algorithm which is a widely used gradient-based optimization method.

Fig. 9 shows the cumulative minimum delivery time with the increased number of simulation runs. The BO $|C|$ = 8 improves the mean cumulative delivery time by 5.27%, 4.33%, and 4.32% with the different numbers of vehicles compared to the GA $|C|$ = 8, and decreases the standard deviations of the mean cumulative delivery time by 58.32%, 61.62%, and 22.2%. The BO $|C|$ = 8 shows a high mean and standard deviation compared to the BO $|C|$ = 2 and 4 because the high dimensionality of the input variables reduces the convergency of the optimization. When the AMHS uses 190 vehicles, the BO $|C|$ = 4 finds the balance between design space complexity and solution quality robustness. When the AMHS uses 200 and 210 vehicles, the BO $|C|$ = 2 is more advantageous compared to the BO $|C|$ = 8.

C. Operational Impacts of the Vehicle Dwelling Policies on the AMHS

Each simulation optimization alternative provides a different distribution of idle vehicles α_c and β_c by each repetition. Therefore, we use the average value of α_c and β_c to validate the performance of each optimization alternative. We use intuitive heuristics for the vehicle dwelling policy for comparative results, outer-loop-circulatory policy (OLCP) and inner-loop-circulatory-policy (ILCP) modified from the "move-when-necessary policy" and "continuous-move-policy" in [25]. Under the OLCP, an idle vehicle continues to circulate in the outer loop until it initiates another transfer request to alleviate congestion in the inner-bay loops. The OLCP is advantageous when a vehicle's speed is relatively fast, but the AMHS experiences heavy congestion. Under the ILCP, an idle vehicle circulates in the bay where the vehicle unloads its transfer request until the vehicle initiates another transfer request. The ILCP is advantageous when vehicle inflow and outflow balance for each intra-bay.

Fig. 10 shows the boxplots for the lot waiting time and the empty travel time with the different numbers of vehicles. The GA shows the longest delivery time because it keeps more minimum vehicles than the intra-bays require. When the AMHS uses 190 vehicles, BO $|C|$ = 4 and 8 increase the lot waiting time by 20.26% and 26.80%, but decreases the empty travel time by 7.60% and 8.93%, respectively, compared to the ILCP. Because the improvement in the empty travel time compensates for the loss of the lot waiting time, BO $|C|$ =

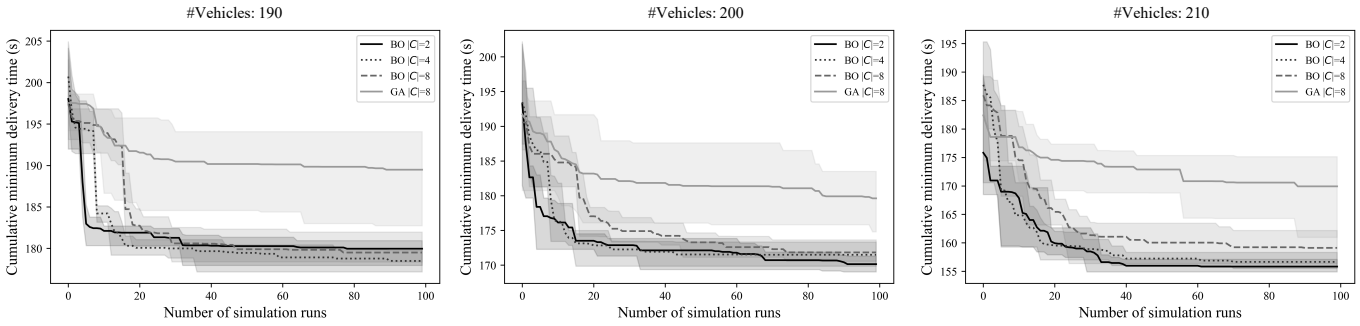

Fig. 9. Cumulative minimum delivery times of the optimization alternatives over the increased number of simulation runs.

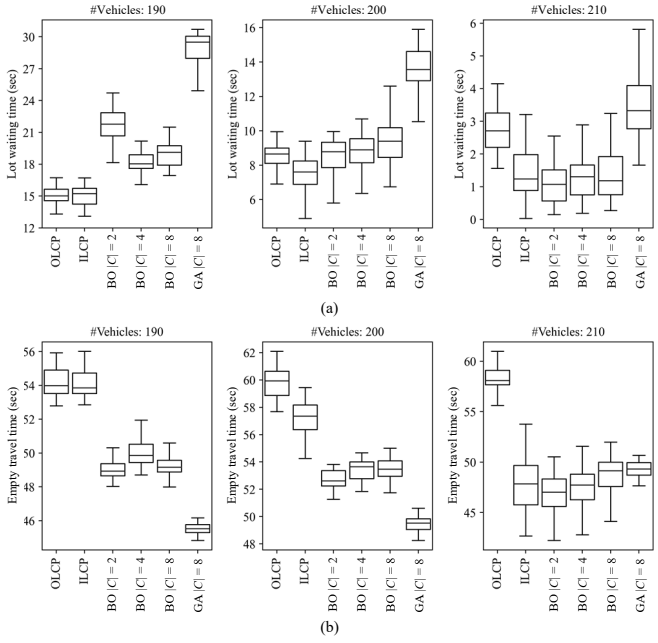

Fig. 10. Performance measurements of the vehicle dwelling policies with 190, 200, and 210 vehicles: (a) average lot waiting time; (b) average empty travel time.

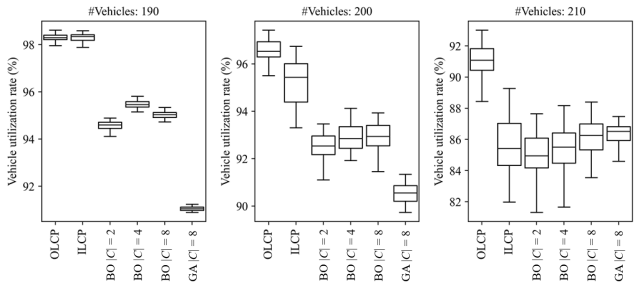

Fig. 11. Average vehicle utilization rates with the different numbers of vehicles.

4 and 8 reduce the delivery time. The improvements in the sum of the lot waiting time and the empty travel time from BO $|C| = 4$ and 8 are 1.55% and 1.16%, respectively, compared to the ILCP. When the AMHS uses 200 vehicles, BO $|C| = 2$ outperforms the OLCP, ILCP, and BO $|C| = 4$ and 8 by 4.11%, 2.11%, 1.61%, and 1.30%, respectively. When the AMHS uses 210 vehicles, the vehicle dwelling policies have small impacts on the lot waiting time and the empty travel time. Further, the vehicle dwelling policies have a marginal impact on the loaded travel time compared to the waiting time and empty travel time (See Appendix B).

Fig. 11 shows the vehicle utilization rates over the vehicle dwelling policies with the different numbers of vehicles. As mentioned, the prompt reaction for the next transfer request and redistribution of vehicles saves the empty travel time and vehicle utilization rate. Since the heuristics cannot consider the minimum number of required vehicles for the intra-bays, they show high vehicle utilization rates. Contrary to the OLCP and ILCP, the GA $|C| = 8$ shows the lowest vehicle utilization rate with the longest delivery time among the alternatives because it assigns more vehicles than the intra-bays require. When the AMHS uses 190 vehicles, the BO $|C| = 4$ and 8 reduce the vehicle utilization rates by 2.84% and 3.32%, respectively, with improvements in the delivery time compared to the ILCP. When the AMHS uses 210 vehicles, the alternatives, except for the OLCP, show small differences between the vehicle utilization rates.

As residual analyses are satisfied at the 95% confidence level, we run statistical significance tests to obtain the p-values of the different vehicle dwelling policies on the delivery time. **Table I** shows that vehicle dwelling policies have significant impacts on the delivery time when the AMHS uses 190 vehicles. The ILCP and the BO-based dwelling policies ($|C| = 2$ and 4) have no statistical differences when the AMHS uses 210 vehicles. The results confirm the need to set the number of classes carefully because a larger number of classes cannot guarantee more improvement, depending on the vehicle utilization rate and vehicle congestion, even if the required number of simulation runs increases.

D. The GP Versus Three Surrogate Alternatives: Comparative Results

In this subsection, we conducted comparative experiments using three surrogate alternatives (SVR: Support Vector Regression, RF: Random Forest, and ANN: Artificial Neural Network). We used the scikit-learn library for the implementations and chose the hyperparameters with a grid search. **Fig. 12** and **Fig. 13** show that the GP outperforms the SVR, the RF, and the ANN in delivery time by 5.02%, 7.30%, and

TABLE I
THE P-VALUES OF THE DIFFERENT VEHICLE DWELLING POLICIES ON THE AVERAGE DELIVERY TIME

#Vehicles: 190					
#OHT 190	ILCP	BO $\|C\|=2$	BO $\|C\|=4$	BO $\|C\|=8$	GA $\|C\|=8$
OLCP	0.0003	<0.0001*	<0.0001*	<0.0001*	<0.0001*
ILCP	-	<0.0001*	<0.0001*	<0.0001*	<0.0001*
BO $\|C\|=2$	-	-	<0.0001*	<0.0001*	<0.0001*
BO $\|C\|=4$	-	-	-	0.0899	<0.0001*
BO $\|C\|=8$	-	-	-	-	<0.0001*
#Vehicles: 200					
#OHT 200	ILCP	BO $\|C\|=2$	BO $\|C\|=4$	BO $\|C\|=8$	GA $\|C\|=8$
OLCP	<0.0001*	<0.0001*	<0.0001*	<0.0001*	<0.0001*
ILCP	-	<0.0001*	<0.0001*	<0.0001*	<0.0001*
BO $\|C\|=2$	-	-	0.0024	<0.0001*	<0.0001*
BO $\|C\|=4$	-	-	-	0.0893	0.0008
BO $\|C\|=8$	-	-	-	-	0.1219
#Vehicles: 210					
	ILCP	BO $\|C\|=2$	BO $\|C\|=4$	BO $\|C\|=8$	GA $\|C\|=8$
OLCP	<0.0001*	<0.0001*	<0.0001*	<0.0001*	<0.0001*
ILCP	-	0.1084	0.3246	0.0168	<0.0001*
BO $\|C\|=2$	-	-	0.4593	<0.0001*	<0.0001*
BO $\|C\|=4$	-	-	-	<0.0001*	<0.0001*
BO $\|C\|=8$	-	-	-	-	<0.0001*

*: P-value lower than 0.0001

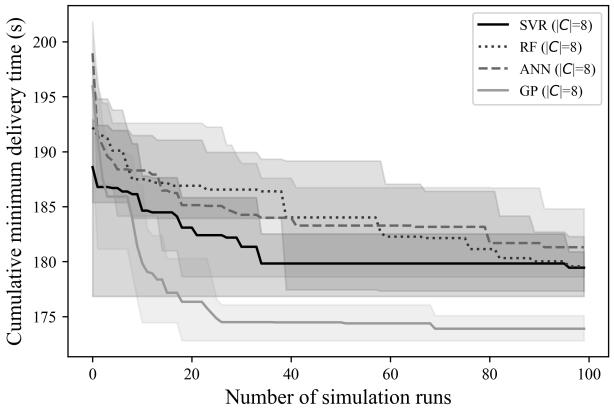

Fig. 12. Cumulative minimum delivery times of the GP and the three surrogate alternatives over the increased number of simulation runs.

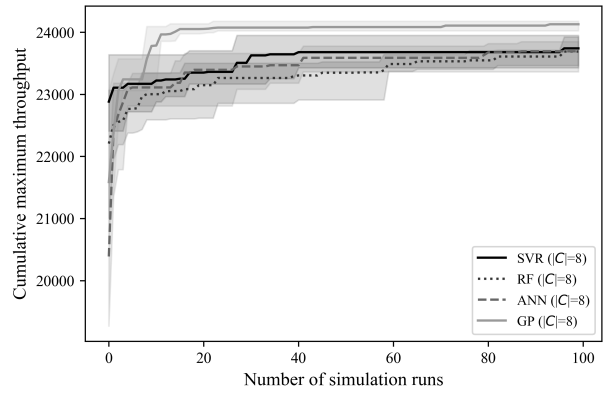

Fig. 13. Cumulative maximum throughputs of the GP and the three surrogate alternatives over the increased number of simulation runs.

6.23% within 50 simulation runs, and by 5.02%, 5.17%, and 6.23% within 100 simulation runs, respectively. The SVR, the RF, and the ANN underperform in processing the expected throughput (number of lots to be transferred) by 1.77%, 2.71%, and 3.83%, respectively, compared to the GP. In summary, the GP speeds up the convergence and achieves an efficient vehicle dwelling policy.

VI. CONCLUSION

This study proposed a simulation optimization approach based on the BO with the class-based clustering to determine the minimum and maximum number of idle vehicles required to respond to transfer requests in a fab's AMHS. In practice, industry simulation models require intensive searches over simulation scenarios with combinatorial decision variables. We suggest that wafer fabs operating over hundreds of OHTs could save millions of dollars in OHT investments because the BO with the class-based clustering uses fewer runs and shorter computation times to obtain efficient solutions.

Our interviews with industry experts confirmed the ongoing problem of inflow and outflow imbalances in wafer fabs with an AMHS. The problem arose when some stockers filled empty FOUPs using material handling devices such as manual carts, and the bays connected with the stockers requested more outflows than the inflows to the AMHS. Imbalances also occurred when vehicles could not access an area if a material handling device became inoperable.

We suggest the following topics for future research. One: A full investigation of intra-bay imbalance problems between inflows and outflows in semiconductor wafer fabs with an AMHS. Two: A surrogate model-based simulation optimization for layout problems with complex constraints in a fab's AMHS. Three: An optimization of the number of classes and clusters for future transportation demands based on historical vehicle operations and transportation demands. Four: An assessment and integrated analysis of the impacts of vehicle dwelling policies on various configurations of an AMHS and its operational policies, i.e., dynamic vehicle routing and initiating policies.

APPENDIX

A. The GA-Based Simulation Optimization

This section details the GA-based simulation optimization. The GA randomly generates solutions for an initial population and generates two offspring based on the best two solutions via crossover and mutation operators, iteratively. We use the two-point crossover operator, which randomly selects crossing points and then exchanges the genes from the parents (See **Fig. 14**). The mutation operator randomly determines all genes in an individual. **Fig. 15** shows the flowchart of the GA-based simulation optimization.

B. Impacts of the Vehicle Dwelling Policies on the Loaded Travel Time

Fig. 16 shows the loaded travel time over the vehicle dwelling policies with the different numbers of vehicles. The results indicated that the vehicle dwelling policies have marginal impacts on the loaded travel time compared to the waiting and empty travel times (See **Fig. 10**). The loaded travel time has no statistical difference between the alternatives under 210 vehicles. The GA $|C|=8$ slightly outperforms

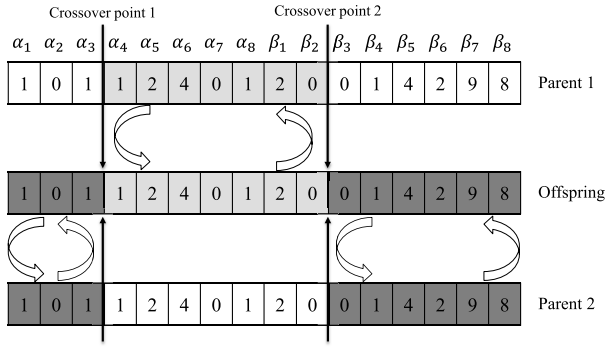

Fig. 14. The two-point crossover operator with $|C| = 8$.

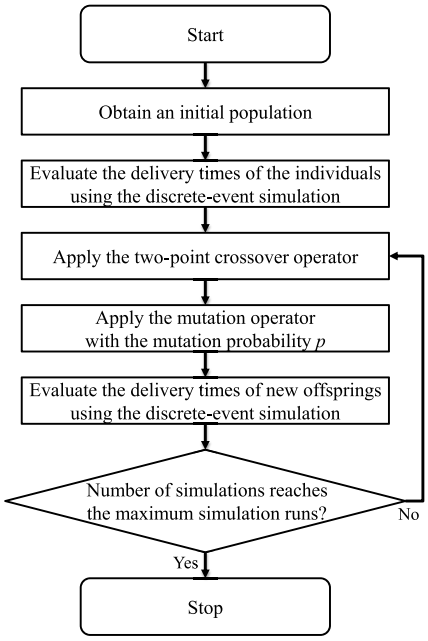

Fig. 15. Procedure of the GA-based simulation optimization.

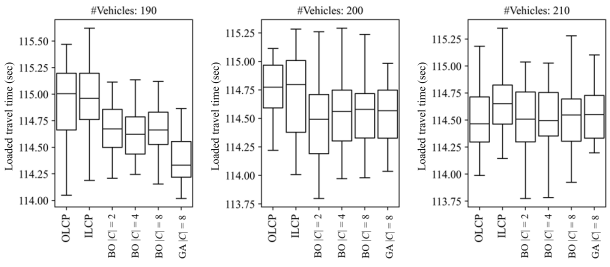

Fig. 16. Average loaded travel times with the different numbers of vehicles.

the alternatives in the loaded travel time under 190 vehicles. Because the GA $|C| = 8$ shows the lowest vehicle utilization rate (See **Fig. 11**), it mitigates vehicle congestion.

REFERENCES

[1] R. Schmaler et al., "Strategies to empower existing automated material handling systems to rising requirements," *IEEE Trans. Semicond. Manuf.*, vol. 30, no. 4, pp. 440–447, Nov. 2017.

[2] S. Hong, A. L. Johnson, H. J. Carlo, D. Nazzal, and J. A. Jimenez, "Optimising the location of crossovers in conveyor-based automated material handling systems in semiconductor wafer fabs," *Int. J. Prod. Res.*, vol. 49, no. 20, pp. 6199–6226, Oct. 2011.

[3] C.-F. Chien, C.-W. Chou, and H.-C. Yu, "A novel route selection and resource allocation approach to improve the efficiency of manual material handling system in 200-mm wafer fabs for Industry 3.5," *IEEE Trans. Autom. Sci. Eng.*, vol. 13, no. 4, pp. 1567–1580, Oct. 2016.

[4] J. C. Chen, R.-D. Dai, and C.-W. Chen, "A practical fab design procedure for wafer fabrication plants," *Int. J. Prod. Res.*, vol. 46, no. 10, pp. 2565–2588, May 2008.

[5] J. T. Lin, F.-K. Wang, and P.-Y. Yen, "Simulation analysis of dispatching rules for an automated interbay material handling system in wafer fab," *Int. J. Prod. Res.*, vol. 39, no. 6, pp. 1221–1238, Jan. 2001.

[6] H.-S. Min and Y. Yih, "Selection of dispatching rules on multiple dispatching decision points in real-time scheduling of a semiconductor wafer fabrication system," *Int. J. Prod. Res.*, vol. 41, no. 16, pp. 3921–3941, Jan. 2003.

[7] S. Amaran, N. V. Sahinidis, B. Sharda, and S. J. Bury, "Simulation optimization: A review of algorithms and applications," *Ann. Oper. Res.*, vol. 240, no. 1, pp. 351–380, 2016.

[8] J. Snoek, H. Larochelle, and R. P. Adams, "Practical Bayesian optimization of machine learning algorithms," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1–9.

[9] I. F. A. Vis, "Survey of research in the design and control of automated guided vehicle systems," *Eur. J. Oper. Res.*, vol. 170, no. 3, pp. 677–709, May 2006.

[10] W. L. Maxwell and J. A. Muckstadt, "Design of automatic guided vehicle systems," *IIE Trans.*, vol. 14, no. 2, pp. 114–124, Mar. 1982.

[11] P. J. Egbelu, "Positioning of automated guided vehicles in a loop layout to improve response time," *Eur. J. Oper. Res.*, vol. 71, no. 1, pp. 32–44, Nov. 1993.

[12] C.-H. Hu and P. J. Egbelu, "A framework for the selection of idle vehicle home locations in an automated guided vehicle system," *Int. J. Prod. Res.*, vol. 38, no. 3, pp. 543–562, Feb. 2000.

[13] G. Bruno, G. Ghiani, and G. Improta, "Dynamic positioning of idle automated guided vehicles," *J. Intell. Manuf.*, vol. 11, no. 2, pp. 209–215, 2000.

[14] K. Nishida and T. Nishi, "Dynamic optimization of conflict-free routing of automated guided vehicles for just-in-time delivery," *IEEE Trans. Autom. Sci. Eng.*, vol. 20, no. 3, pp. 2099–2114, Jul. 2022.

[15] D. Nazzal and L. F. McGinnis, "Analytical approach to estimating AMHS performance in 300 mm fabs," *Int. J. Prod. Res.*, vol. 45, no. 3, pp. 571–590, Feb. 2007.

[16] A. Baykasoğlu, N. Dudaklı, K. Subulan, and A. S. Taşan, "An integrated fleet planning model with empty vehicle repositioning for an intermodal transportation system," *Oper. Res.*, vol. 22, no. 3, pp. 2063–2098, Jul. 2022.

[17] D. Nazzal, J. A. Jimenez, H. J. Carlo, A. L. Johnson, and V. Lasrado, "An analytical model for conveyor-based material handling system with crossovers in semiconductor wafer fabs," *IEEE Trans. Semicond. Manuf.*, vol. 23, no. 3, pp. 468–476, Aug. 2010.

[18] J. R. Morrison, "Multiclass flow line models of semiconductor manufacturing equipment for fab-level simulation," *IEEE Trans. Autom. Sci. Eng.*, vol. 8, no. 1, pp. 81–94, Jan. 2011.

[19] J. N. D. Gupta, R. Ruiz, J. W. Fowler, and S. J. Mason, "Operational planning and control of semiconductor wafer production," *Prod. Planning Control*, vol. 17, no. 7, pp. 639–647, Oct. 2006.

[20] K. Ahn and J. Park, "Cooperative zone-based rebalancing of idle overhead hoist transportations using multi-agent reinforcement learning with graph representation learning," *IISE Trans.*, vol. 53, no. 10, pp. 1140–1156, 2021.

[21] I. Hwang and Y. J. Jang, "Q(λ) learning-based dynamic route guidance algorithm for overhead hoist transport systems in semiconductor fabs," *Int. J. Prod. Res.*, vol. 58, no. 4, pp. 1199–1221, Feb. 2020.

[22] K. Ahn, K. Lee, J. Yeon, and J. Park, "Congestion-aware dynamic routing for an overhead hoist transporter system using a graph convolutional gated recurrent unit," *IISE Trans.*, vol. 54, no. 8, pp. 803–816, 2022.

[23] S. Lee, D.-E. Lim, Y. Kang, and H. J. Kim, "Clustered multi-task sequence-to-sequence learning for autonomous vehicle repositioning," *IEEE Access*, vol. 9, pp. 14504–14515, 2021.

[24] S. Lee, J. Lee, and B. Na, "Practical routing algorithm using a congestion monitoring system in semiconductor manufacturing," *IEEE Trans. Semicond. Manuf.*, vol. 31, no. 4, pp. 475–485, Nov. 2018.

[25] B.-I. Kim and J. Park, "Idle vehicle circulation policies in a semiconductor FAB," *J. Intell. Manuf.*, vol. 20, no. 6, pp. 709–717, Dec. 2009.

[26] J.-E. Kiba, G. Lamiable, S. Dauzere-Peres, and C. Yugma, "Simulation of a full 300 mm semiconductor manufacturing plant with material handling constraints," in *Proc. Winter Simulation Conf. (WSC)*, Austin, TX, USA, Dec. 2009, pp. 1601–1609.

[27] J. T. Lin, F. K. Wang, and C. J. Yang, "The performance of the number of vehicles in a dynamic connecting transport AMHS," *Int. J. Prod. Res.*, vol. 43, no. 11, pp. 2263–2276, Jun. 2005.

[28] J. T. Lin and C.-J. Huang, "A simulation-based optimization approach for a semiconductor photobay with automated material handling system," *Simul. Model. Pract. Theory*, vol. 46, pp. 76–100, Aug. 2014.

[29] X. Chang, M. Dong, and D. Yang, "Multi-objective real-time dispatching for integrated delivery in a fab using GA based simulation optimization," *J. Manuf. Syst.*, vol. 32, no. 4, pp. 741–751, Oct. 2013.

[30] J.-H. Lee et al., "A sequential search method of dispatching rules for scheduling of LCD manufacturing systems," *IEEE Trans. Semicond. Manuf.*, vol. 33, no. 4, pp. 496–503, Nov. 2020.

[31] C.-J. Huang, K.-H. Chang, and J. T. Lin, "Optimal vehicle allocation for an automated materials handling system using simulation optimisation," *Int. J. Prod. Res.*, vol. 50, no. 20, pp. 5734–5746, Oct. 2012.

[32] K.-H. Chang, Y.-H. Huang, and S.-P. Yang, "Vehicle fleet sizing for automated material handling systems to minimize cost subject to time constraints," *IIE Trans.*, vol. 46, no. 3, pp. 301–312, Mar. 2014.

[33] Y. Kuo, T. Yang, B. A. Peters, and I. Chang, "Simulation metamodel development using uniform design and neural networks for automated material handling systems in semiconductor wafer fabrication," *Simul. Model. Pract. Theory*, vol. 15, no. 8, pp. 1002–1015, Sep. 2007.

[34] B. Can and C. Heavey, "A comparison of genetic programming and artificial neural networks in metamodeling of discrete-event simulation models," *Comput. Oper. Res.*, vol. 39, no. 2, pp. 424–436, Feb. 2012.

[35] P. I. Frazier, "Bayesian optimization," in *Recent Advances in Optimization and Modeling of Contemporary Problems*, D. Shier, Ed. Catonsville, MD, USA: Informs, 2018, pp. 255–278.

[36] B. Kang, J. Park, S. Hong, and P. V. E. Joatiko, "Yard template planning in a transshipment hub: Gaussian process regression," in *Proc. Winter Simulation Conf. (WSC)*, Singapore, Dec. 2022, pp. 1979–1989.

[37] A. Candelieri, R. Perego, and F. Archetti, "Bayesian optimization of pump operations in water distribution systems," *J. Global Optim.*, vol. 71, no. 1, pp. 213–235, May 2018.

[38] B. Hickish, D. I. Fletcher, and R. F. Harrison, "Investigating Bayesian optimization for rail network optimization," *Int. J. Rail Transp.*, vol. 8, no. 4, pp. 307–323, Oct. 2020.

[39] B. Kang, B. Kim, and S. Hong, "Sequential optimization of a temporary storage location for cooperative twin overhead shuttles in a rail-based automated container terminal," in *Proc. IFIP Int. Conf. Adv. Prod. Manage. Syst.* Cham, Switzerland: Springer, 2022, pp. 285–292.

[40] Z. Wang, F. Hutter, M. Zoghi, D. Matheson, and N. De Feitas, "Bayesian optimization in a billion dimensions via random embeddings," *J. Artif. Intell. Res.*, vol. 55, pp. 361–387, Feb. 2016.

[41] J. Snoek et al., "Scalable Bayesian optimization using deep neural networks," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Lille, France, 2015, pp. 2171–2180.

[42] Z. Wang, C. Li, S. Jegelka, and P. Kohli, "Batched high-dimensional Bayesian optimization via structural kernel learning," in *Proc. Int. Conf. Mach. Learn.*, Sydney, NSW, Australia, 2017, pp. 3656–3664.

[43] S. Rana, C. Li, S. Gupta, V. Nguyen, and S. Venkatesh, "High dimensional Bayesian optimization with elastic Gaussian process," in *Proc. Int. Conf. Mach. Learn.*, Sydney, NSW, Australia, 2017, pp. 2883–2891.

[44] C. Park, D. J. Borth, N. S. Wilson, and C. N. Hunter, "Variable selection for Gaussian process regression through a sparse projection," *IISE Trans.*, vol. 54, no. 7, pp. 699–712, 2022.

[45] M. B. Salem, F. Bachoc, O. Roustant, F. Gamboa, and L. Tomaso, "Gaussian process-based dimension reduction for goal-oriented sequential design," *SIAM/ASA J. Uncertainty Quantification*, vol. 7, no. 4, pp. 1369–1397, Jan. 2019.

[46] C. Chen and Q. Liao, "ANOVA Gaussian process modeling for high-dimensional stochastic computational models," *J. Comput. Phys.*, vol. 416, Sep. 2020, Art. no. 109519.

[47] X. Guo, Y. Yu, and R. B. M. De Koster, "Impact of required storage space on storage policy performance in a unit-load warehouse," *Int. J. Prod. Res.*, vol. 54, no. 8, pp. 2405–2418, Apr. 2016.

[48] M. A. Millstein, L. Yang, and H. Li, "Optimizing ABC inventory grouping decisions," *Int. J. Prod. Econ.*, vol. 148, pp. 71–80, Feb. 2014.

[49] K. Bartlett, J. Lee, S. Ahmed, G. Nemhauser, J. Sokol, and B. Na, "Congestion-aware dynamic routing in automated material handling systems," *Comput. Ind. Eng.*, vol. 70, pp. 176–182, Apr. 2014.

[50] K. Bartlett, "Congestion-aware dynamic routing in automated material handling systems," Ph.D. dissertation, School Ind. Syst. Eng., Georgia Inst. Technol., Atlanta, GA, USA, 2014.

[51] A. Bhosekar and M. Ierapetritou, "Advances in surrogate based modeling, feasibility analysis, and optimization: A review," *Comput. Chem. Eng.*, vol. 108, pp. 250–267, Jan. 2018.

[52] C. E. Rasmussen, "Gaussian processes in machine learning," in *Proc. Summer School Mach. Learn.* Berlin, Germany: Springer, 2003, pp. 63–71.

[53] M. C. Shewry and H. P. Wynn, "Maximum entropy sampling," *J. Appl. Stat.*, vol. 14, no. 2, pp. 165–170, 1987.

[54] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *J. Global Optim.*, vol. 13, no. 4, pp. 455–492, 1998.

[55] M. L. Pinedo, *Scheduling*. New York, NY, USA: Springer-Verlag, 2012.

Bonggwon Kang received the B.S. degree in industrial engineering from Pusan National University, South Korea, where he is currently pursuing the Ph.D. degree. His research interests include simulation modeling, analysis, and optimization of material handling operations in container terminals and semiconductor fabs.

Chiwoo Park (Senior Member, IEEE) received the B.S. degree in industrial engineering from Seoul National University and the Ph.D. degree in industrial engineering from Texas A&M University in 2011. He is currently a Professor with the Department of Industrial and Manufacturing Engineering, Florida State University. He is also an Affiliate Professor with the Department of Industrial and Systems Engineering, University of Washington. His main research interests include machine learning and data science with a focus on modeling and analysis of object data such as image, shape, motion, function, and directional data, AI-driven scientific discovery with a focus on modeling of physical and computer experiments and active learning for effective exploration of large dimensional experimental spaces for important scientific findings. He is a member of INFORMS and MSA. He is a Senior Member of IISE. He is an Associate Editor of *IISE Transactions* and IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING.

Haejoong Kim received the B.S., M.S., and Ph.D. degrees in industrial engineering from Seoul National University in 2001, 2003, and 2008, respectively. He was a Software Engineer and a Data Analyst with Samsung Electronics from 2008 to 2021. He is currently an Assistant Professor with the Department of Industrial and Management Engineering, Kyonggi University. His research interests include mathematical models and machine learning approaches to manufacturing and service systems.

Soondo Hong received the B.S. and M.S. degrees in industrial engineering from POSTECH, South Korea, and the Ph.D. degree from the Department of Industrial and Systems Engineering, Texas A&M University. He has worked on scheduling, material handling, and simulation at semiconductor and display companies. He is currently a Professor with the Department of Industrial Engineering, Pusan National University, South Korea. His research interests include logistics operations, material handling, and simulation in manufacturing, warehousing, display, semiconductor, and container terminal industries. He is a member of INFORMS and the Korean Institute of Industrial Engineers.