Simulation and Analytical Models of Flexible, Robotic Automotive Assembly Line

Yi-Shiuan Tung, Michael Kelessoglou, Matthew Gombolay, and Julie Shah

***Abstract*—Seeking to adapt to a rapidly changing market, the automotive industry is interested in flexible assembly lines that can handle disruptions resulting from machine failures, scheduling changes, or stochastic task times. In this paper, we propose a layout for transporting cars that incorporates mobile robotic platforms capable of moving off of the assembly line when disruptions occur. We use discrete event simulation to analyze the throughput of our flexible layout on a segment of an automotive assembly line, with the results indicating average speed improvements of 26% and 36% compared with a conventional layout for a single band and two bands with a finite buffer, respectively. In addition, we study the robustness of the flexible layout in the presence of additional inefficiencies inherent in the adoption of new technologies. Next, we present analytical models for throughput analyses of both layouts. We improve upon previous two-machine line analytical models by augmenting the state space to model every machine in a band, and report that the discrete models best approximate the throughput in most cases.**

Note to Practitioners: ***Abstract*—This paper was motivated by the problem of slowdowns that occur on automotive assembly lines due to machine breakdowns, logistical delays, or uneven task times. Currently, cars typically move linearly along conveyor belts or monorails that are fixed in position; we propose a new layout that transports cars using mobile platforms able to move to the side or onto a different track. We modeled a segment of an automotive assembly line in simulation, and show that our proposed layout yields significant improvements to throughput. Upon the inclusion of additional disruptions into the model in order to account for unforeseen problems that can occur when implementing a new technology, the proposed layout continued to outperform the conventional linear layout. We also present analytical approaches for throughput analysis so that future designs can be validated more quickly. Our methods are applicable to analysis of other assembly designs that handle a variety of products and can experience random disruptions during operations. In future work, we plan to improve the simulation's fidelity and expand the analysis to longer assembly lines.**

***Index Terms*—Flexible assembly line, Discrete event simulation, Throughput analysis**

I. Introduction

RECENT challenges in the automotive industry include frequent changes to customer demands, updates to assembly technologies, and the introduction of new models and materials. The traditional assembly line designed for mass production cannot efficiently support a market that demands a significant degree of product variety and customization. One ongoing issue is the need for an assembly line to respond and adapt to market changes, product changes, and system failures [1], all of which require a flexible production system [2]. Previous work has identified several potential areas of manufacturing flexibility with respect to the volume and variety of products, as well as variety among processes used for assembly and material handling [3]. In this work, we present a new assembly line layout that addresses process flexibility: the ability of the system to handle changes and disruptions to the manufacturing process, including machine failures, scheduling changes, or stochastic task times due to mixed-model assembly.

The assembly process is divided into four stages: stamping, body shop, painting, and final assembly (FA) [4]. Our proposed layout uses mobile platforms to transport cars along a band in the FA stage, where most variations in task times due to customization occur. (For example, optional features, such as navigation systems and moonroofs, are installed during the FA stage.) In the conventional layout (CL), cars are transported along a linear conveyor belt, and the band is divided into workstations at which specific tasks are performed. When a car requires more time at a particular station due to assembly disruptions, the entire band pauses until any issues are resolved, resulting in undue idle time for workers. The proposed layout, which we henceforth refer to as the flexible layout (FL), allows a system the flexibility to move a car to different tracks or off of the line altogether when disruptions occur. Figures 1 and 2 depict CL and FL, respectively.

The FL requires the following major changes from the CL: 1. equipment must be placed on mobile platforms, 2. logistics must either be placed on or moved along the platforms via autonomous systems such as automatic guided vehicles (AGVs) [4], and 3. workers are required to perform multiple different assembly tasks. While further research is required for the design of these mobile platforms, we study how such a layout would be able to mitigate disruptions along an assembly line. The main contribution of this paper is the analysis of the benefits of FL, assuming that such technology is available.

First, we use discrete event simulation to model both the CL and FL for performance analysis. Using data collected from an automotive plant, we simulate the manufacturing process of a single band and two bands with a single finite buffer. In order to address the potential for unforeseen inefficiencies following the adoption of FL, we also provide an analysis of FL performance when additional disruptions are added. Lastly, because simulation has relatively long implementation time and large computational costs, we present analytical models and evaluate their throughput prediction by comparing them

Y. Tung, M. Kelessoglou and J. Shah are with the Massachusetts Institute of Technology, Cambridge, MA, 02139 USA. (email: yishiuant@gmail.com; mkelesoglou@gmail.com; julie_a_shah@csail.mit.edu)

M. Gombolay is with the Georgia Institute of Technology, Atlanta GA, 30331 USA. (email: matthew.gombolay@cc.gatech.edu)

to the simulation's throughput.

In Section II, we review related work in the fields of assembly line technologies, simulation, and analytical models. In Section III, we formulate the problem and list our assumptions. In Section IV, we describe the simulation architecture and present our findings. In Section V, we present the analytical models for throughput prediction and evaluate the models by comparing their results with the simulation throughput. We summarize our findings and provide suggestions for future work in Section VI and VII.

II. Related Work

Early work assessing flexible manufacturing systems (FMS) studied the implementation of machines capable of multiple operations [5]; however, wide adoption of FMS is hindered by high equipment cost and low throughput [1]. In addition, FMS cannot scale quickly and thus do not respond well to market or product changes.

The concept of reconfigurable manufacturing systems (RMS) aims to solve scalability and responsiveness issues via changeable structures [1]. Hybrid reconfigurable systems combine RMS with human-centered design in order to better realize the advantages of human-robot collaboration [6].

Mobile robotics represents a major advancement with regard to system reconfigurability. For example, BMW's Mobi-Cell concept allows a robot cell (which includes 25-30 robots) to be loaded onto a truck and put into production at a new plant 3 days later [7]. The Peugeot 307 body shop uses extra robots that can automatically replace mainline robots that experience technical issues [8].

More recently, mobile robots with reconfigurable end effectors that can autonomously move along the factory floor have been studied [9], [10]. Mobile robots can perform a variety of functions and are able to move to different locations along an assembly line to resolve issues or help finish tasks. This approach is most similar to our proposed flexible layout, but in our case, car parts are carried and tasks are performed on mobile platforms. To the best of our knowledge, such a layout has not yet been proposed or evaluated, either in simulation or through analytical modeling.

Simulation has been widely used for assembly system design and performance analysis [11], and discrete event simulation (DES) is a particularly popular technique because of the discrete nature of assembly lines. DES has reportedly been used in over 40% of research papers in the fields of manufacturing and business and is appropriate for process analysis, resource utilization, queuing, and other forms of short-term analysis [12]. Smith provides a survey of DES in manufacturing systems [11].

Previous work has demonstrated the importance of analytical models in throughput analysis [13]. Papadopoulos *et al.* surveys Markov models of manufacturing systems including two machine lines and numerical solutions for larger systems [14]. This paper only addresses single- and two-band lines that incorporate unreliable machines. With regard to serial two-machine lines [15], existing work includes assessment of analytical models for both synchronous [16], [17], [18], [19] and asynchronous systems [16], [19], [20], [21]. (The uptimes and downtimes of machines are geometrically distributed in synchronized models and exponentially distributed in asynchronized models.) Other types of distributions, such as Erlang and phase-type, have also been analyzed in prior works [22], [23]; here, we consider only the geometric and exponential families of uptimes and downtimes. Given that existing methods incorporate two-machine lines, we augment the Markov chain state representation in order to model two-band lines; we also differentiate between methods for conventional and flexible layouts.

III. Problem Formulation

A. Assembly Layout

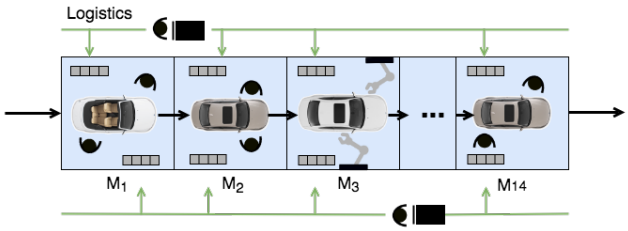

Fig. 1: The conventional layout consists of a linear line. If any station experiences slowdowns, the entire line pauses for the station to catch up.

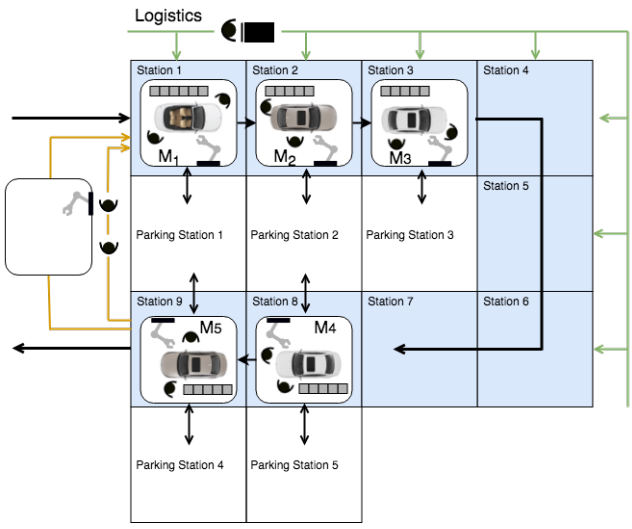

Fig. 2: The flexible layout uses mobile platforms to transport cars. If a platform experiences slowdowns, the platform moves to a parking station to avoid blocking other cars.

In a conventional layout (CL), a band consists of a series of n work stations $M_1, ..., M_n$, where each work station M_i is assigned a fixed set of agents for the entire duration of the production process. In a flexible layout (FL), a band consists of main stations and parking stations, with each station able to include at most one mobile platform. Each mobile platform $M_i, i \in \{1...n\}$ is assigned a set of agents when car $c \in C$ is loaded onto that platform (n represents the total number of mobile platforms in the system). If a platform experiences a

slowdown, the platform moves to an adjacent parking station if one is available. If there are multiple adjacent parking stations, it moves to one with the least adjacent working stations. For example, in Figure 2, M_5 moves to parking station 4 if both parking stations 1 and 4 are available. In this work, we consider $1,385$ car variants, which is the cardinality of C. We anonymize the agents in both layouts, and assume the agents can perform all assigned tasks. As shown in Figures 1 and 2, we consider a linear layout for CL and a sideways U-shaped band for FL; however, FL can take any shape, depending upon the designer's optimization criteria. In our case, FL benefits from a sideways U-shaped band, because mobile platforms that finish unloading a car have a shorter distance to travel to the beginning of the line, and parking stations are accessible via multiple stations along the line.

B. Assembly Task

The assembly process for a car c is divided into tasks W_c, where each task $w \in W_c$ is defined by the number of agents n_w required to perform it, and a fixed duration, d_w. In CL, each task w is also assigned to a specific workstation. We use simple temporal constraints (STC) to encode ordering constraints between tasks. A STC has the form $\alpha_{ij} \leq t_j - t_i \leq \beta_{ij}$ for two time points, t_j and t_i, and a lower and upper bound $[\alpha_{ij}, \beta_{ij}]$ [24]. Let w_1 and w_2 be any two tasks with start times st_{w_1} and st_{w_2} and end times et_{w_1} and et_{w_2}, respectively. In our case, w_2 can be constrained by w_1 as follows:

1) Delay constraint $\alpha_{w_1 w_2} \leq st_{w_2} - et_{w_1} \leq \infty$*:* w_2 can only begin execution after w_1 has been completed for at least some time $\alpha_{w_1 w_2}$. (There is no upper bound.)

2) Immediate constraint $0 \leq st_{w_2} - et_{w_1} \leq 0$*:* w_2 must be executed immediately after completion of w_1.

3) Simultaneous constraint $0 \leq st_{w_2} - st_{w_1} \leq 0$*:* Execution of w_2 begins at the same time as w_1.

C. Assembly Error

Any disruption along the line that affects the production process is considered an error. In CL, we do not differentiate between types of errors (such as machine breakdown vs. logistical delays), and errors have the same effect (stopping the band). In FL, we model five different types of errors defined by the following properties:

1) Requires Immediate Parking: : The platform must park immediately and wait at the parking station until the given error is resolved. If no parking station is available, the platform remains in the error state.

2) Requires Agent to Resolve: : One of the agents on the platform is assigned to resolve the given error (for example, tool wear or machine breakdown) and cannot perform tasks for that errors duration.

3) Increases Makespan: : The given error increases the makespan of the car due to any rework that must be performed.

4) Blocks the Current Task: : The current task cannot be worked on until the given error (for example, a logistical delay) is resolved. Any progress on the task is lost. Table I depicts the error types and their associated properties.

	Requires Immediate Parking	Requires Agent to Resolve	Increase Makespan	Blocks the Current Task
Type 1		•	•	
Type 2	•			
Type 3		•		•
Type 4			•	•
Type 5				•

TABLE I: Properties of the different error types modeled in our simulation.

D. Definitions

1) Throughput: We evaluate the layouts by the throughput (TP), or production rate, which is the number of cars produced per hour. The line efficiency, (E), refers to the proportion of time during which a line is operational. Our interest is in steady-state performance, or the average long-term throughput of the production line. Let λ_s be the number of units produced per hour on a line without errors, the throughput is determined by $TP = \lambda_s \times E$.

2) Failures and Repairs: We use machines as a general term to refer to stations in CL or platforms in FL. For a line with n machines, each machine M_i for $i \in n$ has a p_i probability of breaking down per cycle. Once that machine is down, the average number of cycles needed to repair machine M_i is represented as the mean time to repair $(MTTR_i)$, or mean downtime. The average time for machine M_i to fail is represented as the mean time to failure $(MTTF_i)$, or mean uptime. Let r_i be the probability that machine M_i is repaired during a cycle given that it is down at the beginning of that cycle:

$$r_i = \frac{1}{MTTR_i}, \qquad p_i = \frac{1}{MTTF_i} \tag{1}$$

Here, the uptime and downtime of machines follow geometric distributions. This is a discrete model because the time unit is discretized into cycles. For a continuous model, machines up- and downtimes follow exponential distributions. Let λ_i and μ_i represent the failure and repair rates of the i^{th} machine. Equation 1 is as follows:

$$\mu_i = \frac{1}{MTTR_i}, \qquad \lambda_i = \frac{1}{MTTF_i} \tag{2}$$

This is equivalent to modeling each machine as a M/M/1/k queue, where queue size, k, is one, and the errors occur according to a Poisson process (i.e., exponentially distributed inter-arrival times with parameter λ). The proportion of time that machine i spends resolving errors, ρ_i, is $\lambda_i/(\lambda_i + \mu_i)$. The efficiency of machine i, η_i, is given by the following:

$$\eta_i = 1 - \rho_i = 1 - \frac{\lambda_i}{\lambda_i + \mu_i} = \frac{\mu_i}{\lambda_i + \mu_i} \tag{3}$$

3) Failure Mode: Failures can be divided into two categories: time-dependent failures (TDF) or operation-dependent failures (ODF). TDF can occur at any time, while ODF only occurs while a machine is operating. Some examples of TDF include power line or transfer mechanism failures, while ODF are task-related and include equipment breakdown or logistical delays. We only consider single-station failures, and as a result,

the failure and repair of each machine is independent of other machines. For our analytical models, CL is modeled with ODF, but FL is modeled with TDF in order to account for mobile platform failures.

4) Two-Machine vs. Two-Band Lines: A two-machine line consists of machine M_1 connected to a buffer B_1, which is in turn connected to a second machine, M_2. M_1 is known as the "upstream machine, and M_2 the "downstream machine. A two-band line consists of a series of machines, $M_1, ..., M_n$, connected to a buffer B_1, which is in turn connected to a second series of machines, $M_{n+1}, ..., M_{n+m}$. In this case, $M_1, ..., M_n$ is known as the "upstream band and $M_{n+1}, ..., M_{n+m}$ the "downstream band. In our case, the two bands include the same number of machines ($n = m$). Figures 3 and 4 depict the two-machine and two-band lines, respectively.

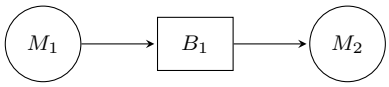

Fig. 3: Two-Machine Line

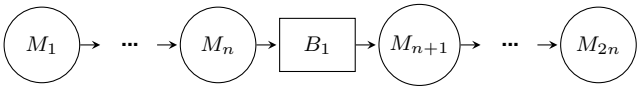

Fig. 4: Two-Band Line

The buffer serves as a storage queue and decouples the machines or bands. A machine is considered starved if the upstream buffer is empty, and blocked if the downstream buffer is full; a machine becomes idle when either starved or blocked. We do not incorporate deliberate idleness into our models, so a given machine is always processing a part if it is able to. We consider a saturated model, wherein the first machine is never starved and the last machine is never blocked; a non-saturated model is used when the input and output are modeled by stochastic processes. For both the simulation and analytical models, we assume that the buffer has no transition time (i.e., a part that enters the buffer can immediately be loaded onto the next machine).

E. Buffer Properties [25]

1) Monotonicity: The production rate monotonically increases as a function of the buffer capacities. More formally, consider two lines, L_1 and L_2, with buffer capacities z_1 and z_2. If $z_1 \leq z_2$, then $TP(L_1) \leq TP(L_2)$, where $TP(L_i)$ is the throughput of line L_i.

2) Infinite buffer size: For a line with infinite buffer capacity, the throughput is bounded by the smallest throughput of an individual machine along that line. Consider a two-machine line, and let TP_1 and TP_2 be the isolated throughput of machines M_1 and M_2. The throughput (TP) of the line is given by $min(TP_1, TP_2)$; it follows that $TP \leq min(TP_1, TP_2)$ for any buffer size.

3) Reversibility: If a line with n machines is reversed, such that machine M_i and buffer B_i in the original line are the same as machine M_{n-i+1} and buffer B_{n-i+1} in the reversed line, then the two lines have the same throughput.

F. Assumptions

We summarize our main assumptions as follows:

- We consider homogeneous and interchangeable agents and therefore do not take their skill sets into account for task assignment.
- The service, failure, and repair processes of each machine are independent of other machines; all machines are identical and have the same service, failure, and repair probabilities and rates.
- Based on the data collected, CL is modeled with ODF, and FL is modeled with TDF.
- The buffer is finite with capacity Z and does not incur movement cost (i.e., the transition time is zero).
- The first machine is never starved and the last machine is never blocked. A machine is always processing a part if it is not in an error or idle state.

IV. Simulation

A. Single Band

The simulation is composed of the pre-processor, scheduler, and simulator. To simplify task scheduling, the pre-processor eliminates simultaneous and immediate constraints (SC and IC, respectively). A pair of tasks with SC, w_1 and w_2, is merged into a single task, $w_{1,2}$, where $d_{w_{1,2}} = max(d_{w_1}, d_{w_2})$ and $n_{w_{1,2}} = n_{w_1} + n_{w_2}$. The duration of this new task is the maximum of the two original tasks' duration, and the number of agents required for the new task is the sum of the number of agents of the original two tasks. With IC, the new task $w_{1,2}$ has a duration of $d_{w_{1,2}} = d_{w_1} + d_{w_2}$ and $n_{w_{1,2}} = max(n_{w_1}, n_{w_2})$ number of agents. Note that in CL, tasks with IC can only be merged if both tasks are assigned at the same station, whereas FL does not have this restriction. When chains of SC and IC exist for a given set of tasks, tasks with SC are merged prior to those with IC. Task merging results in suboptimal schedules when $n_{w_1} \neq n_{w_2}$ for tasks with IC and when $abs(d_{w_1} - d_{w_2}) > \epsilon$ for tasks with SC; however, such cases are rare in our data, justifying the simplification. The pre-processors output is a new set of tasks wherein the only type of temporal constraint that exists between tasks is the delay constraint.

In CL, the scheduler assigns tasks for all cars on the band in each cycle; in FL, the scheduler assigns all tasks for a single car upon that vehicles entry onto the band. Given a set of tasks, the scheduler uses a greedy strategy to output a satisficing schedule. The scheduler first assigns tasks without delay constraints to available agents, then assigns the remaining tasks when their constraints are satisfied. In FL, when a mobile platform either blocks the current task or encounters an error that an agent is required to resolve (i.e., error types 1, 3, 4, and 5), the tasks that have yet to be executed are rescheduled. The simulator emulates the production process by keeping track of the state of the band and the cars; it uses a priority queue to process events such as assembly errors, task completion, and the movement of cars along the line.

B. Sensitivity of Flexible Layout to Band Efficiency

When a new line is introduced, it is possible that the predicted times necessary to complete each job within the assembly process are over-optimistic. This optimism could arise, for example, from unforeseen periodic delays that workers may have to address. Further, the workers may simply not have enough experience on the new line to be able to act as quickly as initially anticipated.

Before investing in a new assembly line, it is important to understand the operational parameters required to make that investment profitable. In order to investigate these parameters, we ran the simulation with additional errors, varying the error rate λ and machine efficiency η. Rearranging Equation 2, given λ and η, the repair rate μ is given by $\mu = \frac{\eta\lambda}{1-\eta}$.

C. Two Band

To extend single-band simulation to systems involving two bands, we use a client-server model to pass messages between the upstream band, downstream band, and buffer. The bands and buffer are clients that send messages to the server, which then relays these messages to their intended recipients. The upstream and downstream bands only communicate with the buffer, and the buffer communicates with both bands.

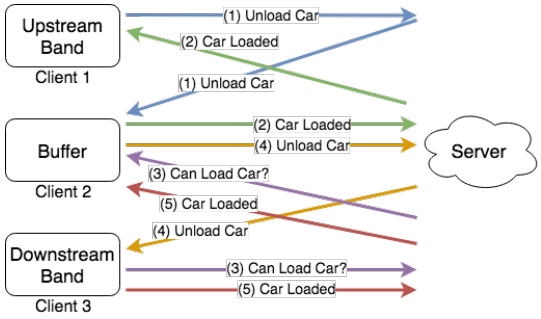

Fig. 5: A client-server model designed to relay messages during the process of loading and unloading cars along an assembly line.

When the upstream band is ready to unload a car (i.e., the car is ready to exit the band), the band sends a message to the buffer (1). If the buffer is not currently full, it replies to the band immediately and increases its inventory level by one (2). However, if the buffer is full, it waits until a spot is open before replying. In CL, the upstream band is blocked until the buffer replies; in FL, the upstream band may be able to load a new car by moving cars into parking stations, and the band only becomes blocked when all main and parking stations are occupied.

On the other hand, when the downstream band is ready to load a car, it sends a message to the buffer (3). The buffer immediately replies with a message indicating whether it's currently empty(4). If the buffer unloads a car, the downstream band replies with an acknowledgement (5) but can continue to the next cycle if the buffer is empty. The downstream band is only identified as starved when it is entirely empty.

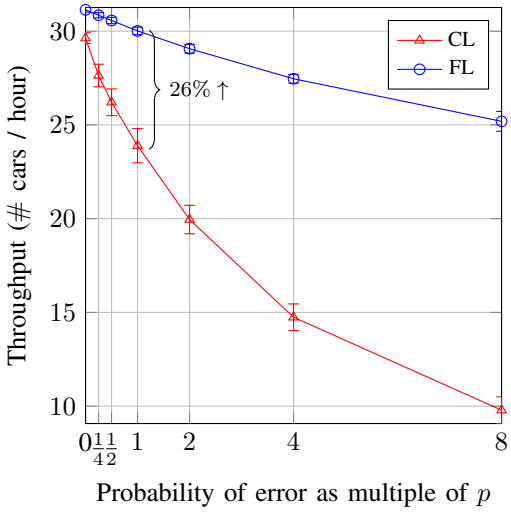

Fig. 6: A comparison of the CL and FL simulation results. For factory-observed error probability (i.e., an error multiple of 1), FL achieved an average speed improvement of 26% over CL.

D. Data Set

Task and error data used in this work were collected from an operational automotive assembly line in the FA stage. The cars produced along this line are highly customizable, which means that tasks often vary between cars. We used a black-box tool that generates sets of tasks and constraints while taking the frequencies of different customization into account. The tool takes 8 minutes to generate tasks for a single car in a virtual machine on a commercial 2.3GHz Intel Core i5 processor with 8GB of RAM. To speed up the simulation, we uniformly sampled from a pre-generated pool of $1,385$ car variants.

We also used a black-box tool from the automotive plant that samples the line's typical error types and duration for both CL and FL. The errors used in the simulation were uniformly sampled from a pre-generated pool of $1,194$ errors for each layout.

For each cycle, each station that has tasks in progress (in CL) or is occupied by a platform (in FL), has a p probability of experiencing an error. The parameter p is chosen such that the expected number of errors is equal to the number of errors observed along the operational assembly line on which our data set was based. We also evaluated layouts with different multiples of p, or error multiples, $\phi \in \{0, 0.25, 0.5, 1, 2, 4, 8\}$, in the simulation. Each simulation was run until 700 cars were produced, representing the daily production rate of the factory on which our model was based. We ran 30 simulations for each error multiple ϕ. In both layouts, we used a cycle time of 100 seconds. The simulation was implemented in the Java programming language.

E. Results

1) Single Band: Figure 6 depicts average throughput as a function of error multiple ϕ, or multiples of factory-observed error probability p. When the line had no errors, FL had a slightly higher throughput than CL due to the lack of spatial

constraints in FL. Since tasks in FL are performed on a single mobile platform, task execution can begin immediately after a given tasks temporal constraints are satisfied. In CL, tasks may be assigned to different stations, and task execution cannot begin until the car is at a particular station. The advantage of FL is more apparent as ϕ increases. Recall that the band stops when an error occurs in CL, while mobile platforms can move to a parking station in FL, allowing the other platforms to continue along the band. The increase in throughput improvement as a function of ϕ indicates that parking stations can help to alleviate error-related slowdown.

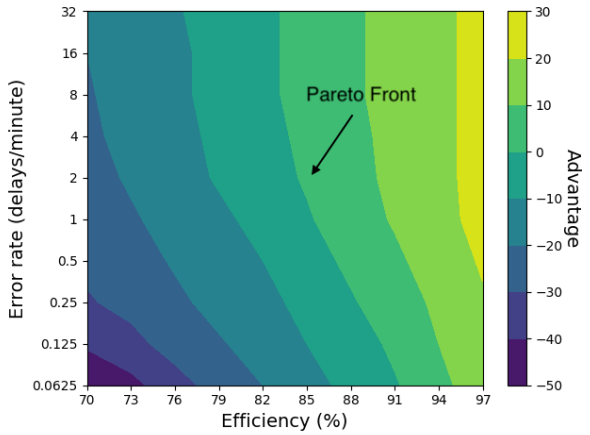

Fig. 7: The average performance gain of FL as a function of efficiency and the number of delays per minute. FL has an advantage over CL if efficiency is at least 94% or if delays resolve promptly.

In order to analyze the sensitivity of FL's performance gain to additional inefficiencies, we simulated the layouts with factory-observed error frequency (i.e., $\phi = 1$) and included additional errors to FL. (These additional errors represent unforeseen delays that could follow from the adoption of FL.) Figure 7 depicts the average performance gain of FL as a function of error rate λ and machine efficiency η. The repair rate μ is determined by η and λ, as shown in Equation 3. FL performs better than CL if machine efficiency is at least 94%; with lower efficiency, FL can still perform better if errors are frequent but of short duration. FL's advantage is given by the following:

$$Advantage = \frac{TP_{FL} - TP_{CL}}{TP_{CL}}$$

TP_L is the throughput of layout L.

2) Two Band: Figures 8 and 9 depict the throughput of the layouts and FL's advantage over CL, respectively, as a function of error multiple ϕ and buffer size. (Note that ϕ is the same for the upstream and downstream bands.) In Figure 8, the throughput of both layouts generally increases as a function of buffer size (property III-E1); it does not monotonically increase due to noise from the simulation of a stochastic environment. In all cases, the maximum throughput is bounded by the throughput of the single-band layout with the same error multiple (property III-E2).

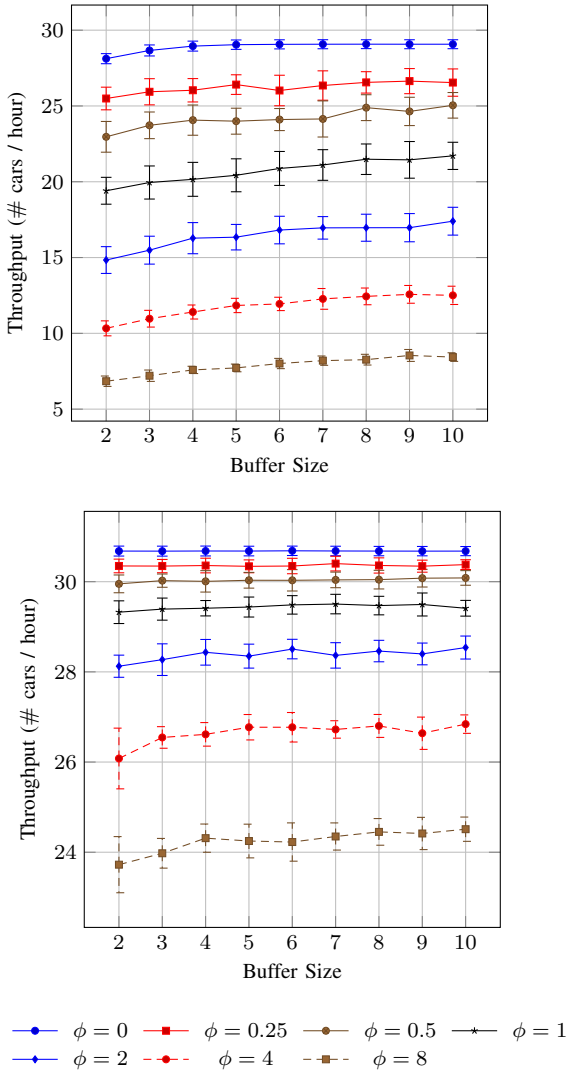

Fig. 8: Throughput of the two-band CL (top) and FL (bottom) with equal band failure probability. The buffer improves the throughput of both layouts, but this improvement is bounded and does not overcome the deficit resulting from greater error frequency.

Figure 9 indicates that FL's advantage is greatest when ϕ is high and the buffer size is small. This advantage decreases as buffer size increases because the buffer helps to alleviate error propagation and has a greater effect on a line that is itself more affected by errors. With regard to factory-observed error probability (i.e., $\phi = 1$), the two-band FL has a 36% advantage over CL with a buffer size of 10. On a line with no errors, FL's advantage remains constant as buffer size varies; in this case, the throughput converged to its upper bound, and increasing buffer size does not improve the throughput of either layout.

Figure 10 depicts the throughput of the line when error multiples differ between the upstream and downstream bands, with the buffer size fixed to five. One finding of note is the presence of the reversibility property (III-E3): a line where the upstream and downstream bands have failure probabilities of p_1 and p_2 has the same throughput as a line where the

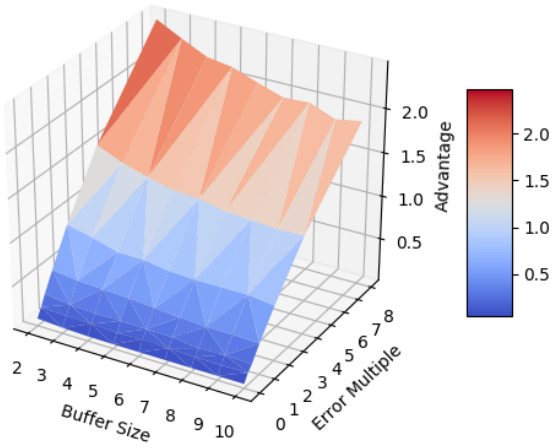

Fig. 9: The advantage of FL over the CL as a function of buffer size and error frequency. FL's advantage is greatest with higher error multiples and smaller buffer sizes.

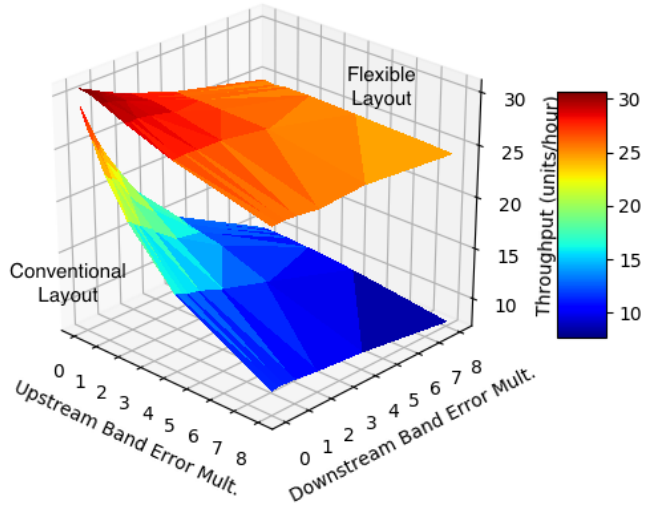

Fig. 10: The throughput of CL and FL with a buffer capacity of five. Both layouts exhibit the reversibility property.

upstream and downstream bands have failure probabilities of p_2 and p_1. We also observe that the throughput is higher for a line with error multiples of 2 for both bands than a line with no errors for one band and an error multiple of 4 for the other. This adheres to property III-E2, in that the throughput of the entire line is bounded by the machine with the worst throughput.

V. Analytical Models

Simulation is a useful tool for assembly line analysis but has several drawbacks, including a lengthy implementation time, difficult debugging due to the stochastic environment, and potentially high computational costs. In this section, we present analytical models that predict the throughput of CL and FL.

The main differences between the analytical models for the two layouts are that CL is modeled with operation-dependent failures (ODF) while FL is modeled with time-dependent failures (TDF), as well as the assumption that FL is operational when the number of machines in error is less than or equal to the number of parking stations. This assumption overestimates throughput because it ignores the loss in time resulting from cars transferring to parking stations. A more-accurate model would include the specific stations in error and the occupancy of the parking stations into the state space; however, this would also result in a significantly larger, intractable state space.

A. Single Band

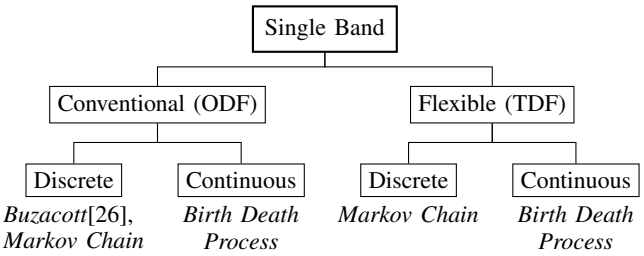

Fig. 11: Overview of Single-Band Analytical Models

Prior work in this area includes Buzacott's analysis of a line with ODF [26]. Buzacott's formula assumes that only one machine can be down at any time. For TDF, Buzacott considered the system analogous to a line of independent machines connected in series [27]. However, we do not evaluate Buzacott's TDF model because it is not straightforward how to adopt the formula to account for parking stations. In order to model failures involving multiple machines, we present the Markov chain and birth-death process models. These models handle multiple machine failures by using the number of machines in error as the state representation.

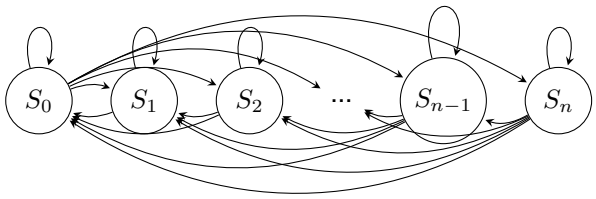

Fig. 12: Markov Chain for CL

1) Markov Chain: This is a discrete Markov chain model with geometric uptimes and downtimes. In the ODF case, the state of the Markov chain is an integer-valued random variable that represents the number of machines in error during a given cycle. We use S_i to indicate a state with i machines in error. The set of possible states is $\boldsymbol{S} = \{S_0, S_1, S_2, ..., S_n\}$ for a line with n machines. Figure 12 depicts the Markov chain with arrows indicating a non-zero probability of transition. At the start of a cycle (represented by S_0), every machine has a p probability of failing; the state transitions from S_0 to S_i if i machines fail. In the case of ODF, machines that did not fail become idle after completing tasks in the current cycle; therefore, S_i cannot transition to states S_j where $j > i$. For the set of machines in error, each machine has an independent

probability of repair, r, per cycle; thus, S_i is connected to every state S_j where $j < i$.

We use the binomial distribution to compute transition probabilities. Let T be the transition matrix and $T[i][j]$ be the transition probability from S_i to S_j. The following equations define all non-zero probabilities in T:

$$\begin{aligned} T[0][j] &= \binom{n}{j} p^j (1-p)^{n-j} \quad \forall\, j \in \{1 \dots n\} \\ T[i][j] &= \binom{i}{i-j} r^{i-j} (1-r)^j \quad \text{if } i > j \\ T[0][0] &= (1-p)^n \\ T[i][i] &= (1-r)^i \quad \forall\, i \in \{1 \dots n\} \end{aligned} \tag{4}$$

The Markov chain described above is ergodic, and there exists a unique positive steady-state vector π for ergodic finite-state Markov chains that is a left eigenvector of T corresponding to an eigenvalue $\lambda = 1$ [28].

$$T = U \Lambda U^{-1} \tag{5}$$

$$\pi T = \lambda \pi \quad \lambda = 1 \tag{6}$$

Equation 5 shows the eigendecomposition of T, where the columns of U are left eigenvectors of T and Λ is a diagonal matrix of eigenvalues. Equation 6 shows that π is the steady state vector. Let $\pi[0]$ be the first element of π corresponding to the steady state probability of S_0. The throughput of CL is calculated by multiplying $\pi[0]$ by λ_s, the service rate.

$$TP_{CL} = \lambda_s \times \pi[0] \tag{7}$$

In the TDF case, the Markov chain becomes a fully-connected graph. Given that idle machines can fail, non-zero probabilities exist for transitioning from state S_i to states with more than i machine failures. The transition matrix T for a n-machine line is given as follows:

$$T[0][0] = (1-p)^n \tag{8}$$

$$T[n][n] = (1-r)^n \tag{9}$$

$$T[0][j] = \binom{n}{j} p^j (1-p)^{n-j} \quad \text{if } 0 < j \leq n \tag{10}$$

$$T[n][j] = \binom{n}{n-j} r^{n-j} (1-r)^j \quad \text{if } 0 \leq j < n \tag{11}$$

$$\begin{aligned} T[i][j] = \sum_{n_r=0}^{N_r^{ij}} & \binom{N_r}{n_r} r^{n_r} (1-r)^{N_r - n_r} \binom{n-i}{j-i+n_r} \\ & p^{j-i+n_r} (1-p)^{n-j-n_r} \\ & \text{if } 0 < i < j, \quad N_r^{ij} = min(i, n-j) \end{aligned} \tag{12}$$

$$\begin{aligned} T[i][j] = \sum_{n_p=0}^{N_p^{ij}} & \binom{n-i}{n_p} p^{n_p} (1-p)^{n-i-n_p} \binom{i}{i-j+n_p} \\ & r^{i-j+n_p} (1-r)^{j-n_p} \\ & \text{if } n > i > j, \quad N_p^{ij} = min(j, n-i) \end{aligned} \tag{13}$$

$$\begin{aligned} T[i][j] = \sum_{n_e=0}^{N_e} & \binom{n-i}{n_e} p^{n_e} (1-p)^{n-i-n_e} \binom{i}{n_e} \\ & r^{n_e} (1-r)^{i-n_e} \\ & \text{if } i = j, \quad N_e^i = min(i, n-i) \end{aligned} \tag{14}$$

Equations 8 and 9 represent the probabilities that no machines failed or were repaired, respectively. Equation 10 shows transitions from zero machine errors to j machines with errors. Equation 11 depicts transitions during which $n - j$ machines were repaired. Equation 12 shows transitions from states with fewer machine failures to states with more machine failures. N_r^{ij} is the maximum number of machines that can be repaired while transitioning from S_i to S_j. Equation 12 sums all possible numbers of repaired machines that can still transition to a state where j machines are in error. Equations 13 and 14 are similar to Equation 12, but are for transitions from a state with more machine failures to a state with fewer machine failures and self-transitions, respectively.

The steady state vector π is calculated via Equations 5 and 6, and the throughput is given by the following:

$$TP_{FL} = \lambda_s \times \sum_{i=0}^{5} \pi[i] \tag{15}$$

$\sum_{i=0}^{5} \pi[i]$ is the proportion of time during which a line with five parking stations is operational.

2) Birth-Death Process: The birth-death process is a continuous model wherein uptimes and downtimes follow exponential distributions. It is a special type of Markov process wherein each state has two transitions: birth and death. A birth transition increases the state by one, while a death transition decreases the state by one. Figure 13 depicts the embedded Markov chain of our birth-death model.

Machine failures are treated as independent Poisson processes, each with rate λ [1]. When there are n machines, the combined Poisson process of all machines has a rate of $n\lambda$ [28]. Therefore, for any state S_i, the failure of $n-i$ machines is modeled by $n-i$ independent Poisson processes, which can be considered a single Poisson process with rate of $(n-i)\lambda$. Similarly, the repair of i machines in state S_i is modeled by a single Poisson process with rate $i\mu$.

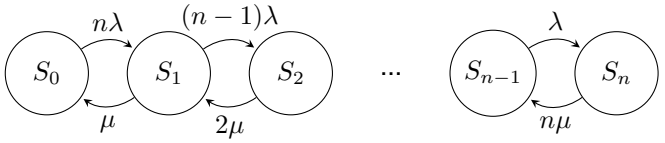

Fig. 13: Birth-Death Process

Let $\pi[i]$ be the time-average fraction of time spent in state S_i. Since the number of transitions from state S_i to state S_{i+1} is within 1 of the number of transitions from state S_{i+1} to state S_i, the following relationship exists: [28]

$$\pi[i](n-i)\lambda = \pi[i+1](i+1)\mu \tag{16}$$

Iteratively applying Equation 16 yields the following:

$$\pi[i] = \pi[0] \frac{n!}{(n-i)!i!} (\frac{\lambda}{\mu})^i \tag{17}$$

[1]A Poisson process models the times at which failures occur, and the intervals between failures are exponentially distributed with rate λ [28].

Since $\sum_{i=0}^{n} \pi[i] = 1$, we substitute 17 into the sum as follows:

$$\pi[0] + \pi[0] \sum_{i=1}^{n} \frac{n!}{(n-i)!i!} (\frac{\lambda}{\mu})^i = 1$$
$$\Rightarrow \pi[0](1 + \sum_{i=1}^{n} \frac{n!}{(n-i)!i!} (\frac{\lambda}{\mu})^i) = 1$$
$$\Rightarrow \pi[0] = \frac{1}{1 + \sum_{i=1}^{n} \frac{n!}{(n-i)!i!} (\frac{\lambda}{\mu})^i} \tag{18}$$

Given the steady state vector π, we use the same formulas as Markov chain to compute the throughput of CL and FL which is given by Equations 7 and 15 respectively.

B. Two Band

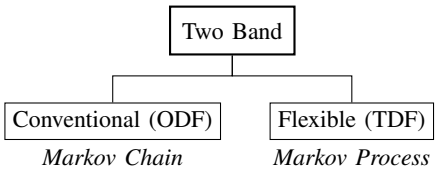

Fig. 14: An overview of two-band analytical models

The two-band Markov chain and Markov process are extensions of the single-band models. Instead of using the state representation S_x, where x represents the number of machines in error, the state is represented by the tuple (S_u, S_d, z), where S_u and S_d represent the states of the upstream and downstream bands and z is the buffer inventory level. In prior work involving two-machine lines, S_u and S_d have been binary (up or down) [17]; in order to extend the representation to two-band lines, we augment the representation such that S_u and S_d represent the number of machines in error and take on values from 0 to n, where n is the number of machines in each band. In this section, we refer to the upstream band as M_u and the downstream band as M_d.

1) Markov Chain: Similar to the single-band Markov chain, we first derive the transition matrix $\boldsymbol{T}$ to compute the steady state vector π. $\boldsymbol{T}$ is based on the single-band transition matrices T_u and T_d for the upstream and downstream bands, as given by Equation 4. We assume that the buffer level only changes if either M_u cannot output a finished part but M_d can process a part or M_u can output a finished part but M_d cannot process a part.

For buffer inventory level $i \in \{1 \dots Z-1\}$, where M_u is not blocked and M_d is not starved, the transitions from (S_{u_1}, S_{d_1}, z) to (S_{u_2}, S_{d_2}, z) are the products of individual band transitions since the two bands are independent. Z is the buffer's maximum capacity.

$$\boldsymbol{T}[S_{u_1}, S_{d_1}, i][S_{u_2}, S_{d_2}, j] = \begin{cases} T_u[S_{u_1}][S_{u_2}] \times T_d[S_{d_1}][S_{d_2}] \\ \quad \text{if } i = j \\ 0 \quad otherwise \end{cases} \tag{19}$$

If the buffer is empty and M_u is down, M_d is starved. Let Ω represent the operational states, and Φ the failure states. For CL, $\Omega = \{S_0\},\ \Phi = \{S_1, ..., S_n\}$; for FL with five parking stations, $\Omega = \{S_0, ..., S_5\},\ \Phi = \{S_6, ..., S_n\}$. CL assumes ODF, so M_d cannot fail when idle; FL assumes TDF, so M_d can still transition to any other state when idle.

$$\boldsymbol{T}[\omega_1, \omega_2, 1][\phi, \omega_3, 0] = T_u[\omega_1][\phi] \times T_d[\omega_2][\omega_3]$$
$$\forall\ \omega_1, \omega_2, \omega_3 \in \Omega,\ \phi \in \Phi \tag{20}$$

Equation 20 shows the transitions where M_d becomes idle due to M_u being down and the buffer being empty.

$$\boldsymbol{T}[\phi_1, \omega, 0][\phi_2, \omega, 0] = T_u[\phi_1][\phi_2]$$
$$\forall\ \phi_1, \phi_2 \in \Phi,\ \omega \in \Omega \tag{21}$$
$$\boldsymbol{T}[\phi, \omega, 0][\omega, \omega, 0] = T_u[\phi][\omega]$$
$$\forall\ \phi \in \Phi,\ \omega \in \Omega \tag{22}$$

Equation 21 depicts the transitions for ODF where M_d stays idle because M_u remains in failure states. The transition probability only depends upon T_u because M_d is idle. Equation 22 shows the transition of M_u from a failure state to an operational state. At the instant M_u becomes operational, M_d also becomes operational. The buffer level stays the same because the part M_u put into the buffer was taken by M_d. The TDF counterpart is shown below:

$$\boldsymbol{T}[\phi_1, S_{d_1}, 0][\phi_2, S_{d_2}, 0] = T_u[\phi_1][\phi_2] \times T_d[S_{d_1}][S_{d_2}]$$
$$\forall\ \phi_1, \phi_2 \in \Phi,\ S_{d_1},\ S_{d_2} \in \Phi \cup \Omega \tag{23}$$
$$\boldsymbol{T}[\phi, \omega_1, 0][\omega_2, \omega_3, 0] = T_u[\phi][\omega_2] \times T_d[\omega_1][\omega_3]$$
$$\forall\ \phi \in \Phi,\ \omega_1, \omega_2, \omega_3 \in \Omega \tag{24}$$

Given that M_d can fail when idle, all transitions must take T_d into account. Equation 23 is equivalent to Equation 21, and Equation 24 is equivalent to Equation 22 in the case of TDF.

The other edge case occurs when M_u is blocked because the buffer is full and M_d is down, as shown in Equation 25.

$$\boldsymbol{T}[\omega_1, \omega_2, Z-1][\omega_3, \phi, Z] = T_u[\omega_1][\omega_3] \times T_d[\omega_2][\phi]$$
$$\forall\ \omega_1, \omega_2, \omega_3 \in \Omega,\ \phi \in \Phi \tag{25}$$

Equations 26 and 28 depict transitions wherein M_u stays idle because M_d remains in failure states for ODF and TDF, respectively. Equations 27 and 29 show the transitions wherein M_d moves to an operational state and M_u is no longer blocked for ODF and TDF, respectively.

$$\boldsymbol{T}[\omega, \phi_1, Z][\omega, \phi_2, Z] = T_d[\phi_1][\phi_2]$$
$$\forall\ \omega \in \Omega,\ \phi_1, \phi_2 \in \Phi \tag{26}$$
$$\boldsymbol{T}[\omega, \phi, Z][\omega, \omega, Z] = T_d[\phi][\omega]$$
$$\forall\ \omega \in \Omega,\ \phi \in \Phi \tag{27}$$
$$\boldsymbol{T}[S_{u_1}, \phi_1, Z][S_{u_2}, \phi_2, Z] = T_u[S_{u_1}][S_{u_2}] \times T_d[\phi_1][\phi_2]$$
$$\forall\ S_{u_1}, S_{u_2} \in \Phi \cup \Omega,\ \phi_1, \phi_2 \in \Phi \tag{28}$$
$$\boldsymbol{T}[\omega_1, \phi, Z][\omega_2, \omega_3, Z] = T_u[\omega_1][\omega_2] \times T_d[\phi][\omega_3]$$
$$\forall\ \omega_1, \omega_2, \omega_3 \in \Omega,\ \phi \in \Phi \tag{29}$$

Given $\boldsymbol{T}$, the steady state vector is determined by Equation 6. The throughput of both layouts is calculated as follows:

$$TP = \lambda_s \Bigg(\sum_{\substack{\omega \in \Omega \\ S_d \in \boldsymbol{S}}} \sum_{z=0}^{Z-1} \pi[\omega, S_d, z] + \sum_{\substack{\omega_1, \omega_2 \\ \in \Omega}} \pi[\omega_1, \omega_2, Z] \Bigg) \quad (30)$$

$$= \lambda_s \Bigg(\sum_{\substack{S_u \in \boldsymbol{S} \\ \omega \in \Omega}} \sum_{z=1}^{Z} \pi[S_u, \omega, z] + \sum_{\substack{\omega_1, \omega_2 \\ \in \Omega}} \pi[\omega_1, \omega_2, 0] \Bigg) \quad (31)$$

This is similar to the line efficiency equation developed by Buzacott [17]. The sums in Equation 30 represent the proportion of time during which the first band is up and not blocked. This equates to the sums in Equation 31, which represent the proportion of time the second band is up and not starved.

2) Markov Process: The two-band continuous model is no longer a birth-death chain because there are more than just birth and death transitions. In addition to transitions for car failures and repairs, there are also transitions for buffer level changes. We continue to model the uptimes and downtimes with the exponential distribution. The buffer inventory increases or decreases if the upstream band M_u or downstream band M_d produces a part; the time to produce a part is also modeled by exponential random variables. We construct a transition rate matrix $\boldsymbol{Q}$ such that $\boldsymbol{Q}[x][y]$ denotes the rate of departing from state x and entering state y. Note that state x or y is a tuple of the form (S_u, S_d, z). One important property of $\boldsymbol{Q}$ is that diagonal elements are defined such that the rows sum to one. $\boldsymbol{Q}[x][x] = -\Sigma_{y \neq x} \boldsymbol{Q}[x][y]$.

Algorithm 1: Create Transition Rate Matrix

$N_u \leftarrow$ number of stations in upstream band
$N_d \leftarrow$ number of stations in downstream band
$Z \leftarrow$ buffer capacity
$\lambda_s, \lambda, \mu \leftarrow$ service rate, failure rate, repair rate
$\boldsymbol{Q} \leftarrow$ initialize matrix
for (i, j, z) *where* $i \in \{0..N_u\}$, $j \in \{0..N_d\}$, $z \in \{0..Z\}$ **do**
  // machine repaired in M_u
  if $i > 0$ **then** $\boldsymbol{Q}[i, j, z][i-1, j, z] = i \times \mu$
  // machine fails in M_u
  if $i < N_u$ **then** $\boldsymbol{Q}[i, j, z][i+1, j, z] = (N_u - i) \times \lambda$
  // machine repaired in M_d
  if $j > 0$ **then** $\boldsymbol{Q}[i, j, z][i, j-1, z] = j \times \mu$
  // machine fails in M_d
  if $j < N_d$ **then** $\boldsymbol{Q}[i, j, z][i, j+1, z] = (N_d - j) \times \lambda$
  // M_u produces a part
  if $z < Z$ *and* $i \in$ *operational states* **then**
    $\boldsymbol{Q}[i, j, z][i, j, z+1] = \lambda_s$
  // M_d produces a part
  if $z > 0$ *and* $j \in$ *operational states* **then**
    $\boldsymbol{Q}[i, j, z][i, j, z-1] = \lambda_s$

To define the transition rates, we use idea that n independent Poisson processes each with rate λ is equivalent to a single Poisson process with rate $n \times \lambda$ [28]. For example, state $(2, S_d, z)$, which has two machines in failure in M_u, has a transition rate of 2μ into state $(1, S_d, z)$ where μ is the repair rate of a single machine. Algorithm 1 defines the transition rates for $\boldsymbol{Q}$. Let $\pi_t[S_u, S_d, z]$ be the probability of being in state (S_u, S_d, z) at time t. As t approaches infinity, π approaches steady state and doesn't change anymore; as a result $lim_{t \to \infty} \pi(t) \boldsymbol{Q} = 0$. We compute π by calculating the left nullspace of the matrix $\boldsymbol{Q}$ [29]. The throughput is then computed by Equations 30 and 31.

C. Parameter Estimation

The required parameters for the analytical models include the service rate, λ_s; failure probability, p; failure rate, λ; repair probability, r; and repair rate, μ for both layouts. We use the hat symbol $\wedge$ to represent the estimated values.

	Service	Two Band Line Failure		Repair	
	$\hat{\lambda_s}$	$\hat{p}$	$\hat{\lambda}$	$\hat{r}$	$\hat{\mu}$
	unit/hour	prob./cycle	unit/min.	prob./cycle	unit/min.
CL	29.63	9.43E-03	1.57E-02	0.404	0.243
FL	31.14	1.47E-02	2.44E-02	0.383	0.230

TABLE II: Estimated parameters

1) Service: We assume that the service times follow an exponential distribution with parameter λ_s, which is estimated by considering the throughput of the line when no errors are present. The units are measured in the number of cars produced per hour.

2) Failure: Based on collected data, a car has a 0.132 probability of experiencing an error while on the band. In order to determine the probability of failure per cycle $\hat{p}$, 0.132 is divided by the number of stations[2]. To calculate $\hat{\lambda}$, the number of failures per minute, $\hat{p}$ is multiplied by the cycle time (1.67 minutes).

3) Repair: The repair time is modeled by either a geometric or exponential random variable. The parameters to estimate for the geometric and exponential distributions are $\hat{r}$ and $\hat{\mu}$ respectively. The maximum likelihood estimator (MLE) for both distributions is computed by $n / \sum_i^n x_i$ where n is the total number of samples and x_i is the i'th repair time. To compute $\hat{r}$, the repair times are first converted to units of cars repaired per cycle. For $\hat{\mu}$, the repair times are measured in units of cars repaired per minute.

D. Results

We use the standard error σ_{pred} and Pearson's correlation P-ρ to evaluate the analytical models. The standard error measures how far off the predicted throughput is compared to the simulation and is computed by

$$\sigma_{pred} = \sqrt{\frac{\sum_i^N (pred_i - sim_i)^2}{N}} \quad (32)$$

where $pred_i$ and sim_i are the i'th sample of predicted and simulation throughput, and N is the total number of samples. Pearson's correlation measures the linear relationship between

[2]In our simulation, CL has 14 stations, while FL has 9 main stations and 5 parking stations.

	Single Band			Two Band	
	Markov Chain	Birth-Death	Buzacott	Markov Chain	Markov Process
σ_{pred}	**0.971**	3.213	1.373	**0.958**	4.125
	1.579	1.581	-	**2.861**	3.590
P-ρ	0.991	**0.994**	0.993	**0.987**‡	0.983
	0.840	0.840	-	**0.867**‡	0.852

TABLE III: Performance of analytical models in terms of the standard error and correlation of the predicted and simulation throughput. For each band type and model, the results for CL are reported on top and FL on bottom. The best model is highlighted in bold, and the superscripts † / ‡ denote statistically significant improvement of the best model compared to other models of the same category at 95% / 99% confidence intervals.

the predicted and simulation throughput. The performance of the analytical models are reported in Table III.

The Markov chain models perform the best when using standard error as the evaluation metric. An interesting observation is that the single-band birth-death model for the flexible layout (FL) has similar predictions as the Markov chain model, with the standard errors differing by only 0.02. But the birth-death model has the highest error on the conventional layout (CL). Figure 15 plots the predicted throughput of single-band analytical models as a function of error multiples ϕ. From Figure 15, the analytical models seem to underestimate the throughput for CL and overestimate for FL. Recall that we assume the FL is functional when the number of cars in error is less than the number of parking stations, and we do not account for the cost of transferring to parking stations. Therefore, the FL models' predictions are upper bounds of the simulation throughput. Our second observation is that the standard error is much higher for two-band models with the exception of the Markov chain. Since the two-band models are extensions of the single-band versions, the inaccuracies from single-band are also observed and usually worsened in the two-band case. However, the Markov chain model seems to be robust for CL, achieving similar standard errors for both single-band and two-band.

The Markov chain models have the highest Pearson's correlation except for the single-band CL. The birth-death model has the highest correlation followed by Buzacott. However, the correlation coefficients for the single-band models are not statistically significant. On the other hand, the two-band models have statistically significant differences for both 95% and 99% confidence intervals. The results here align with the standard error metric where Markov chain outperforms Markov process. An interesting observation is that the FL models are less correlated than the CL models, possibly due to the difficulty in accounting for parking stations.

VI. Discussion

There are two main limitations to our analyses: assembly line abstractions and limited data collection. We used specific tasks and temporal constraints in our simulation, but abstracted away other details, such as equipment usage and

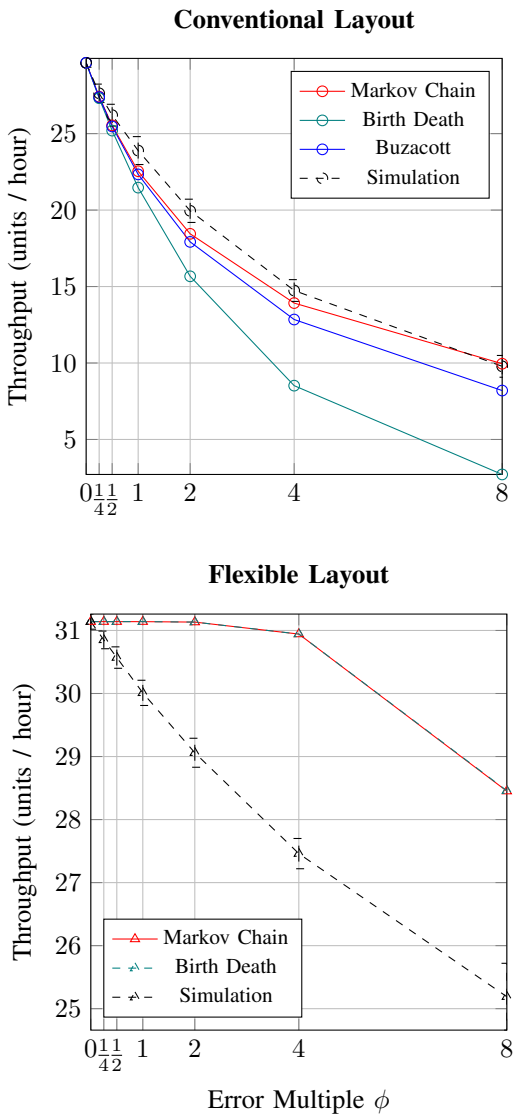

Fig. 15: Throughput predictions of single-band analytical models (CL top, FL bottom). For CL, Markov chain has the closest predictions to the simulation throughput. For FL, Markov chain and birth-death have similar predictions.

worker specialization. We assumed that each station had the equipment necessary for completing the tasks assigned to it; we also anonymized workers and assumed that they all were able to complete their assigned tasks. Our other limitation was that the simulation did not account for all possible disruptions along the band; for example, we only considered single-station failures rather than those that could involve multiple stations simultaneously.

VII. Conclusion & Future Work

The automotive assembly industry's transition from mass production to mass customization is generating interest in flexible assembly lines. In this work, we proposed a flexible assembly line layout (FL) that addresses a system's ability to handle disruptions. FL incorporates mobile platforms in order to transport cars along the band, and allows these platforms

to move off of the line when disruptions occur. Using data on assembly tasks and disruptions collected from an automotive assembly line, we used DES to simulate a segment of the final stage of automotive assembly; the results indicate 26% and 36% increases in throughput by adopting FL in single-band and two-band systems, respectively. Furthermore, we analyzed the performance of FL when not operating at full capacity due to inefficiencies arising when deploying new technology; our findings show that FL has benefits over CL if FL's efficiency is at least 94%. Lastly, we presented analytical models and showed that discrete models best approximate the simulation throughput in most cases.

One direction for future work would be to address the limitations discussed in Section VI. To incorporate equipment data into the simulation, we would have to identify the equipment used and determine whether it could be placed on mobile platforms. Particularly large or expensive equipment could be allocated to specific stations; however, similar to CL, this would then create spacial constraints when use of that equipment is required in order to complete assembly tasks. It is thus important to expand the simulation with equipment data to better compare the two layouts. We could also improve the simulation by considering more types of errors. FL handles single-station failures by moving the mobile platform with an error into a parking station, but we did not consider multi-station failures that would require the movement of mobile platforms. Lastly, it is important to evaluate the performance of both layouts on longer lines, as we only considered single- and two-band models with a single buffer. For analytical models, previous work used approximation methods such as decomposition and aggregation [30], [31]. The simulation used multiple processes in the two-band case, which is computationally expensive for longer lines; single-process frameworks and approaches must also be investigated.

REFERENCES

[1] Y. Koren and M. Shpitalni, "Design of reconfigurable manufacturing systems," *Journal of manufacturing systems*, vol. 29, no. 4, pp. 130–141, 2010.

[2] K. Georgoulias, G. Michalos, S. Makris, and G. Chryssolouris, "The effect of flexibility on market adaptation," in *2nd CIRP Conference on Assembly Technologies and Systems (CATS 2008), Toronto, Canada,(September 2008)*, 2008, p. 280.

[3] D. E. D'Souza and F. P. Williams, "Toward a taxonomy of manufacturing flexibility dimensions," *Journal of operations management*, vol. 18, no. 5, pp. 577–593, 2000.

[4] G. Michalos, S. Makris, N. Papakostas, D. Mourtzis, and G. Chryssolouris, "Automotive assembly technologies review: challenges and outlook for a flexible and adaptive approach," *CIRP Journal of Manufacturing Science and Technology*, vol. 2, no. 2, pp. 81–91, 2010.

[5] K. E. Stecke and J. J. Solberg, "Loading and control policies for a flexible manufacturing system," *The International Journal of Production Research*, vol. 19, no. 5, pp. 481–490, 1981.

[6] A. O. Andrisano, F. Leali, M. Pellicciari, F. Pini, and A. Vergnano, "Hybrid reconfigurable system design and optimization through virtual prototyping and digital manufacturing tools," *International Journal on Interactive Design and Manufacturing (IJIDeM)*, vol. 6, no. 1, pp. 17–27, 2012.

[7] A. Kochan, "Bmw uses even more robots for both flexibility and quality," *Industrial Robot: An International Journal*, vol. 32, no. 4, pp. 318–320, 2005.

[8] ——, "Robots help peugeot meet targets," *Industrial Robot: An International Journal*, vol. 29, no. 6, pp. 500–502, 2002.

[9] G. Michalos, P. Sipsas, S. Makris, and G. Chryssolouris, "Decision making logic for flexible assembly lines reconfiguration," *Robotics and Computer-Integrated Manufacturing*, vol. 37, pp. 233–250, 2016.

[10] G. Michalos, K. Kaltsoukalas, P. Aivaliotis, P. Sipsas, A. Sardelis, and G. Chryssolouris, "Design and simulation of assembly systems with mobile robots," *CIRP Annals-Manufacturing Technology*, vol. 63, no. 1, pp. 181–184, 2014.

[11] J. S. Smith, "Survey on the use of simulation for manufacturing system design and operation," *Journal of manufacturing systems*, vol. 22, no. 2, pp. 157–171, 2003.

[12] M. Jahangirian, T. Eldabi, A. Naseer, L. K. Stergioulas, and T. Young, "Simulation in manufacturing and business: A review," *European Journal of Operational Research*, vol. 203, no. 1, pp. 1–13, 2010.

[13] J. Li, D. E. Blumenfeld, N. Huang, and J. M. Alden, "Throughput analysis of production systems: recent advances and future topics," *International Journal of Production Research*, vol. 47, no. 14, pp. 3823–3851, 2009.

[14] C. T. Papadopoulos, J. Li, and M. E. O'Kelly, "A classification and review of timed markov models of manufacturing systems," *Computers & Industrial Engineering*, vol. 128, pp. 219–244, 2019.

[15] J. Li, D. E. Blumenfeld, and J. M. Alden, "Comparisons of two-machine line models in throughput analysis," *International journal of production research*, vol. 44, no. 7, pp. 1375–1398, 2006.

[16] D. Jacobs and S. M. MEERKOV, "A system-theoretic property of serial production lines: improvability," *International Journal of Systems Science*, vol. 26, no. 4, pp. 755–785, 1995.

[17] J. A. Buzacott and J. Shanthikumar, *Stochastic Models of Manufacturing Systems*, ser. Prentice-Hall international series in industrial and systems engineering. Prentice Hall, 1993. [Online]. Available: https://books.google.com/books?id=A9tTAAAAMAAJ

[18] J. Li and S. M. Meerkov, "Due-time performance of production systems with markovian machines," in *Analysis and modeling of manufacturing systems*. Springer, 2003, pp. 221–253.

[19] S. Gershwin, *Manufacturing Systems Engineering*, ser. American Studies. PTR Prentice Hall, 1994. [Online]. Available: https://books.google.com/books?id=M_ZTAAAAMAAJ

[20] J. Alden, "Estimating performance of two workstations in series with downtime and unequal speeds," *General Motors Research & Development Center, Report R&D-9434, Warren, MI*, 2002.

[21] B. Xia, J. Chen, and Z. Zhang, "An exact method for the analysis of a two-machine manufacturing system with a finite buffer subject to time-dependent failure," *Mathematical Problems in Engineering*, vol. 2015, 2015.

[22] C. Heavey, H. Papadopoulos, and J. Browne, "The throughput rate of multistation unreliable production lines," *European Journal of Operational Research*, vol. 68, no. 1, pp. 69–89, 1993.

[23] T. Altiok, "Approximate analysis of queues in series with phase-type service times and blocking," *Operations Research*, vol. 37, no. 4, pp. 601–610, 1989.

[24] R. Dechter, I. Meiri, and J. Pearl, "Temporal constraint networks," *Artificial intelligence*, vol. 49, no. 1-3, pp. 61–95, 1991.

[25] Y. Dallery and S. B. Gershwin, "Manufacturing flow line systems: a review of models and analytical results," *Queueing systems*, vol. 12, no. 1-2, pp. 3–94, 1992.

[26] J. A. Buzacott, "Automatic transfer lines with buffer stocks," *The International Journal of Production Research*, vol. 5, no. 3, pp. 183–200, 1967.

[27] J. Buzacott and L. E. Hanifin, "Models of automatic transfer lines with inventory banks a review and comparison," *AIIE transactions*, vol. 10, no. 2, pp. 197–207, 1978.

[28] R. Gallager, *Stochastic Processes: Theory for Applications*, ser. Stochastic Processes: Theory for Applications. Cambridge University Press, 2013. [Online]. Available: https://books.google.com/books?id=ERLrAQAAQBAJ

[29] G. Strang, *Introduction to linear algebra*. Wellesley-Cambridge Press Wellesley, MA, 1993, vol. 3.

[30] S. B. Gershwin, "An efficient decomposition method for the approximate evaluation of tandem queues with finite storage space and blocking," *Operations research*, vol. 35, no. 2, pp. 291–305, 1987.

[31] J. Li and S. M. Meerkov, *Production systems engineering*. Springer Science & Business Media, 2008.