Integrating Robot Assignment and Maintenance Management: A Multi-Agent Reinforcement Learning Approach for Holistic Control

Kshitij Bhatta, *Graduate Student Member, IEEE*, and Qing Chang, *Senior Member, IEEE*

***Abstract*—Modern manufacturing requires effective integration of production control and maintenance scheduling to improve productivity and quality. However, there have been few studies on this integrated control due to a lack of a comprehensive manufacturing system model. In response to this challenge, this letter presents a mathematical model framework for a mobile multi-skilled robot-operated manufacturing system that integrates three essential control aspects: robot assignment, maintenance scheduling, and product quality. Furthermore, an integrated control scheme is formulated in the Decentralized Partially Observable Markov Decision Process (Dec-POMDP) framework to showcase the proposed method's efficacy in control. Results show that the proposed integrated model outperforms models that consider only system-level parameters, as well as those that only address maintenance scheduling and quality-related parameters.**

***Index Terms*—AI and machine learning in manufacturing and logistics systems, collaborative robots in manufacturing, computer integrated manufacturing, intelligent and flexible manufacturing, reinforcement learning.**

I. Introduction

SMART manufacturing offers several benefits to the industrial sector, revolutionizing traditional manufacturing processes and enabling enhanced efficiency, productivity, and competitiveness. It leverages advanced technologies such as Internet of Things (IoT), Artificial Intelligence (AI), and data analytics to optimize production processes and improve operational efficiency. It enables real-time monitoring, predictive maintenance, and automation, resulting in reduced downtime and improved overall effectiveness [1]. By implementing smart manufacturing techniques, manufacturers can achieve higher levels of product quality and consistency. For example, AI-powered quality control systems, coupled with real-time monitoring and feedback mechanisms, enable early detection of defects or deviations from desired specifications, ensuring timely corrective actions and minimizing waste [2].

Next-generation manufacturing systems incorporate multiple networked work cells/stations, robots, distributed sensors and wireless devices throughout the facility to provide real-time information on machine and product states. In these systems, the production control subsystem alters production rates of individual work cells and maintenance subsystem schedules repairs and upkeep based on machine (and potentially product) state information. Appropriate production control increases the throughput of the system whereas timely maintenance increases the useful life of equipment and ensures quality compliance in products. Since the overall success and global competitiveness of any company depends on delivering high-quality products, the harmonious integration of production operations, quality control, and maintenance activities is crucial [3]. However, conventional studies have traditionally treated these fields as separate entities. This letter addresses this limitation by developing an integrated system model and control algorithm that holistically considers production control (through robot assignment) and maintenance activities (through Preventive Maintenance (PM) and tool change).

The literature on smart manufacturing is vast. Especially with the advent of industry 4.0 and the very recent industry 5.0, researchers have focused on various related problems such as modeling and automation of manufacturing systems, integration of robotics, and maintenance scheduling [4], [5], [6], [7], [8]. Ref. [4] has modeled a networked manufacturing system for enhanced routing flexibility, [9] has proposed a batch discrete manufacturing model with a control scheme that addresses the issue of perishability in products, and [5] has developed a smart control framework for mobile multi-skilled robots by integrating robot perception and process understanding.

Before, robots were only used for material handling or repetitive tasks, but with the introduction of social robots, new robotic systems are capable of collaborating with humans to increase efficiency. The number of studies on such systems have been increasing rapidly with examples such as [10] which develops a task scheduling framework to improve the make-span and ergonomics in a human robot collaboration environment and [11] which proposes a method to use mobile robot clustering in machining of large workpieces.

Maintenance scheduling has also been studied widely in literature with real-time maintenance scheduling gaining great attention [6], [7], [8]. Ref. [6] addresses a multi-objective preventive maintenance (PM) scheduling problem in a multiple production line setting. The objective of the study is to find an optimal PM schedule that balances the reliability of production lines, maintenance costs, failure rates, and system downtime. An approach for maintenance scheduling in a multi-component production

Manuscript received 28 February 2023; accepted 4 July 2023. Date of publication 12 July 2023; date of current version 19 July 2023. This letter was recommended for publication by Associate Editor F. Ju and Editor J. Yi upon evaluation of the reviewers' comments. This work was supported by the National Science Foundation under Grant 1853454. *(Corresponding author: Qing Chang.)*

The authors are with the Department of Mechanical and Aerospace Engineering, University of Virginia, Charlottesville, VA 22904 USA (e-mail: qpy8hh@virginia.edu; qc9nq@virginia.edu).

Digital Object Identifier 10.1109/LRA.2023.3294717

2377-3766 © 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: VIT University. Downloaded on January 22,2025 at 20:43:32 UTC from IEEE Xplore. Restrictions apply.

system incorporating real-time information from workstations, such as remaining equipment reliability and work-in-process inventories is developed in [7]. The authors use a factorial experimental design to monitor the affect of each possible maintenance schedule and emply a genetic algorithm to find the optimal one. Ref. [8] proposes a novel opportunistic preventive maintenance scheduling methodology for serial-parallel multistage manufacturing systems, optimizing both reliability and product quality. It models machine deterioration propagation, considers total maintenance cost, and achieves high system reliability and product quality with low maintenance costs. A maintenance scheduling algorithm based on a correction factor is introduced in [12] which dynamically updates the Preventive Maintenance (PM) interval to evaluate the risk of not performing PM, and [13] develops a Multi-Agent Reinforcement Learning based PM scheduler.

Despite such vast literature on individual manufacturing aspects, the integration of these has received surprisingly limited attention, particularly within the context of FMSs employing mobile robots. Therefore, the objective of this work is to address this research gap by developing a mathematical control framework that incorporates these aspects to achieve higher efficiency and production quality. This would benefit practitioners in designing control strategies that can effectively minimize losses, and academicians to more efficiently study the problem in a realistic setting. The system used in this letter is a mobile multi-skilled robot operated FMS, the original model for which was developed in [14] but without maintenance or quality considerations.

This letter has three main contributions: 1) Development of a rigorous mathematical model which incorporates robot assignment, maintenance scheduling and product quality 2) Formulation of a control problem in the Decentralized Partially Observable Markov Decision Process (Dec-POMDP) framework to demonstrate its use in control and 3) Comparison of the control strategy from the proposed integrated model with two benchmarks to highlight its importance. The rest of the letter is organized as follows: System introduction and modeling is done in Section II, the control problem is formulated in Section III, and a case study is presented in Section IV. Finally, conclusions are given in Section V.

II. System Description and Model

A mobile multi-skilled robot operated flexible manufacturing system consisting of w workstations, r robots and $w-1$ intermediate buffers is considered. Each robot is represented as $R_i(i=1,\ldots,r)$, each workstation as $W_i(i=1,\ldots,w)$ and each buffer as $B_i(i=2,\ldots,w)$. Quality control mechanisms are in place after every workstation such that all parts that are deemed defective are taken out of circulation. These mechanisms are denoted as $Q_i(i=1,\ldots,w)$. An illustration of the serial FMS is shown in Fig. 1. The following notation is used in the letter:

1) A $r\times 1$ vector $\mathbf{u}(t)$ is used to denote the assignment of all robots at time t. Each element of $\mathbf{u}(t)$, $u_j(t)=i$ if robot R_j is assigned to workstation W_i at time t;
2) The total number of robots assigned to workstation W_i at time t is denoted by $m_i(t)$, $i=1,2,\ldots,w$;

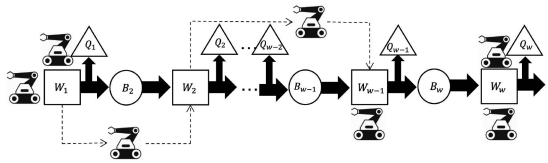

Fig. 1. Illustration of multi-skilled robot operated FMS.

3) $\boldsymbol{\theta}=[\theta_1(t),\theta_2(t),\ldots,\theta_r(t)]^T$ is the vector of working status of all robots. Each element, θ_j is 1 if R_j is working and 0 otherwise. A robot is considered *working* in workstation W_i if it has arrived to it's assigned workstation and is not currently undergoing maintenance;
4) The total number of robots working in a workstation W_i is denoted by $\hat{m}_i(t)$, $i=1,2,\ldots,w$;
5) Each workstation has a specified base cycle time T_i, where $i=1,2,\ldots,w$. This is the time it takes one robot in workstation W_i to produce one part. However, since each workstation can have more than one robot working, the total cycle time for each workstation at any time t is $\frac{T_i}{\hat{m}_i(t)}$. The production speed of the workstation at time t is the reciprocal of the total cycle time, i.e, $\frac{\hat{m}_i(t)}{T_i}$. This direct proportionality is a simplification for math purposes and a different relation does not change the mathematical model to be developed;
6) $\mathbf{b}(t)=[b_2(t),b_3(t),\ldots,b_w(t)]^T$ are the buffer levels at time t. B_i is the buffer capacity of the i^{th} buffer, where, $i=2,3,\ldots,w$;
7) Each robot has one tool for each workstation making w tools per robot. State of robot R_j's i^{th} tool at time t is denoted as $x_j^i(t)$, $i=1,2,\ldots,w\ \forall\ j=1,2,3\ldots,r$;
8) Age of each robot at time t is represented by $a_j(t)$ where $j=1,2,\ldots,r$. There is a prescribed maximum age for each robot, denoted by A_j, and reaching this age leads to its failure;
9) With abuse of notation, $Q_i(t)$ represents the number of parts that have been discarded from W_i, for $i=1,\ldots,w$;
10) $MTBF_j$ and $MTTR_j$ represents the mean time between failures and mean time to repair for robot R_j. This applies for random disruption events;

We make the following assumptions in the letter:

1) All parameters and quantities discussed in the letter are in discrete time. An argument of t, is adopted for ease of notation and corresponds to the value of the parameter at discrete time t. Time advances with a small timestep δt;
2) The total number of robots in the system remains constant, i.e, $\sum_{i=1}^{w} m_i(t)=r,\forall t$.
3) At any given time, the number of robots working in W_i cannot exceed an upper bound, $\hat{m_i}^{[U]}$, i.e, $\hat{m}_i(t)\leq \hat{m_i}^{[U]},\forall t$;
4) The system has a unique and constant ideal clean configuration, i.e, the number of robots in each workstation that results in the best performance is constant and unique, This number is represented by $\hat{m}_{i,c}$ for the workstation W_i [14].
5) A workstation is *slowed down* if the number of robots working on it at time t is less than the number of robots in the ideal clean configuration, i.e, $\hat{m}_i(t)<\hat{m}_{i,c}$ and it is *down* if there are no robots working, i.e, $\hat{m}_i(t)=0$ for $i=1,2,\ldots,w$;

Authorized licensed use limited to: VIT University. Downloaded on January 22,2025 at 20:43:32 UTC from IEEE Xplore. Restrictions apply.

6) A workstation is blocked (partially blocked) if it is operational, its immediate downstream buffer is full and the subsequent downstream workstation is down (slowed down);
7) A workstation is starved (partially starved) if it is operational, it's immediate upstream buffer is empty and the subsequent upstream workstation is down (slowed down);
8) The first workstation W_1 is never starved and the last machine W_w is never blocked since the performance of this isolated production line is to be studied;

A manufacturing system is a dynamic stochastic system subject to disturbances. Thus, it can be modeled using a state space equation,

$$\dot{\mathbf{X}}(t) = \mathbf{F}\left(\mathbf{X}(t), \mathbf{Q}(t), \hat{\mathbf{u}}(t), \boldsymbol{\theta}(t)\right) \tag{1}$$

where each component of (1) is defined as:

- $\mathbf{F}(*)$ is a function that defines the rate of part production for all workstations at any time t.
- $\mathbf{X}(t) = [X_1(t), \ldots, X_w(t)]^T$ is the system state, such that $X_i(t)$ is the cumulative production count of workstation W_i up to time t.
- $\mathbf{Q}(t) = [Q_1(t), \ldots, Q_w(t)]^T$, such that $Q_i(t)$ is the cumulative defective product count from workstation W_i at time t.
- $\hat{\mathbf{u}}(t)$ is vector of control inputs of all robots at time t.
- $\boldsymbol{\theta}(t)$ is the vector of all robot status' at time t.

A. Control Input

The control input to the system is a vector of either assignment actions, tool change actions, maintenance actions or combined tool change and maintenance actions. Consider the set of assignment actions $\mathscr{U}$, the set of tool change actions $\mathscr{C}$, the set of maintenance actions $\mathscr{M}$ and the set of combined tool change and maintenance actions $\mathscr{N} = \mathscr{C} \times \mathscr{M}$. The control input is the vector $\hat{\mathbf{u}}$ such that each element $\hat{u}_i \in \mathscr{U} \cup \mathscr{C} \cup \mathscr{M} \cup \mathscr{N}$ for $i = 1, 2, \ldots r$.

Robots can be assigned to all workstations, i.e, $\mathscr{U} = \{1, 2, \ldots, w\}$. The time remaining for arrival of robot R_i at time t is represented by $\tau_i(t)$, and the constant time it takes a robot to move between concurrent workstations is represented by ψ. Assignment actions can only be taken if the robot is active and has already arrived to its assigned workstation.

Tool change actions are taken to replace the tool, the state of which affects the quality of the products.There can be multiple tool wear levels, the probability of transition among which is conditioned on the current tool state. Therefore, the state of the i^{th} tool of the robot R_j is represented by $x_j^i(t) \in \{1, 2, \ldots, n^t\}$, where n^t is the number of tool states. The transition model can be expressed using the probabilistic form,

$$p(x_j^i(t) | x_j^i(t - \delta t)) \tag{2}$$

This model can be obtained experimentally for the type of tool being used and the type of operation being performed. Tool change actions consist of only one option, "change the tool": $\mathscr{C} = \{1\}$, which when taken resets the new state of the tool to the best possible.

Maintenance actions are taken to restore the health of the robots. Maintenance effects are modeled using the Kijima III model [15]. A robot is functional until a prescribed "virtual age" after which Corrective Maintenance (CM) is automatically evoked unless PM is done to decrease the virtual age. The age is reduced using a multiplicative parameter called the recovery factor, $0 \leq r < 1$, the value of which depends on the level of PM performed. Lower value of r implies better age reduction and vice versa. Thus, the set of robot maintenance actions is $\mathscr{M} = \{r_k\}$ where $k = 1, 2, \ldots, l$ and l is the number of maintenance levels. Finally, the combined tool change and maintenance actions can be defined by the set $\mathscr{N} = \mathscr{C} \times \mathscr{M}$. Besides aging, each robot R_j is subjected to random disruption events, the occurrence of which is assumed to be an exponential random process with a mean of $MTBF_j$.

Performing tool change or maintenance actions lead to downtimes. The time taken for changing i^{th} tool of any robot is denoted by $Time_i^c$ for $i = 1, 2, \ldots, w$, the time taken for Preventive Maintenance (PM) of level k in robot R_j is denoted by $Time_j^{PM,k}$, $k = 1, 2, \ldots, l \; \forall \; j = 1, 2, \ldots, r$. l is the total number of PM levels, and the time taken for Corrective Maintenance (CM) in robot R_j is denoted by $Time_j^{CM} \; \forall \; j = 1, 2, \ldots, r$. The time taken for combined tool change and robot maintenance is equal to having just robot maintenance since it is assumed that the two are carried out in parallel. The time that is remaining for the repair to be completed on robot R_i at time t is denoted by $t_i^{rep}(t)$.

B. Model Derivation

For a small value of δt, forward Euler discretization can be applied to get a discrete version of (1):

$$\mathbf{X}(t + \delta t) - \mathbf{X}(t) = \mathbf{F}\left(\mathbf{X}(t), \mathbf{Q}(t), \hat{\mathbf{u}}(t), \boldsymbol{\theta}(t)\right) \delta t \tag{3}$$

The dynamics of the production system which is defined by the function, $\mathbf{F}(\mathbf{X}(t), \mathbf{Q}(t), \hat{\mathbf{u}}(t), \boldsymbol{\theta}(t))$ is derived in what follows.

The number of robots assigned to workstation W_i at time t, $m_i(t)$ can be defined as,

$$m_i(t) = \sum_{j=1}^{r} \mathbb{1}_{u_j(t)=i} \tag{4}$$

where $\mathbb{1}_{u_j(t)=i}$ is a binary variable which indicates if robot j is assigned to workstation i or not.

From among all the assigned robots, the robots that are down or have not arrived can then be subtracted to find the number of working robots, i.e, $\hat{m}_i$,

$$\hat{m}_i(t) = m_i(t) - \sum_{j=1}^{r} \left(1 - \theta_j(t)\right) \mathbb{1}_{u_j(t)=i} - \sum_{j=1}^{r} \left(1 - \delta_k(\tau_i(t), 0)\right) \mathbb{1}_{u_j(t)=i} \tag{5}$$

where, $\delta_k(x, y)$ is the Kronecker delta function, i.e., $\delta_k(x, y) = 1$ if $x = y$ and 0 otherwise.

The difference in accumulated *defective* production counts between workstation W_i and W_j within a time period $[0, t] \; \forall . i = 1, 2, \ldots, w, j = 1, 2, \ldots, w$, and $i \neq j$ is represented by,

$$Q_{ij}(t) = \begin{cases} \sum_{k=j}^{i} Q_k(0) - \sum_{k=j}^{i} Q_k(t) & i > j \\ \sum_{k=i}^{j} Q_k(t) - \sum_{k=i}^{j} Q_k(0) & i < j \end{cases} \tag{6}$$

Authorized licensed use limited to: VIT University. Downloaded on January 22,2025 at 20:43:32 UTC from IEEE Xplore. Restrictions apply.

The difference in accumulated production counts between two workstations W_i and W_j in a time period of [0,1] is represented by $\mu_{ij}(t)$. Thus, using the conservation of flow and (6), the difference in accumulated production of *non-defective* parts between two workstations, W_i and $W_j, \forall i, j \in 1, 2, \ldots w, i \neq j$ within a time period $[0, t]$ can be defined as:

$$\mu_{ij}(t) - Q_{ij}(t) = \begin{cases} \sum_{k=j+1}^{i} b_k(0) - \sum_{k=j+1}^{i} b_k(t) & i > j \\ \sum_{k=i+1}^{j} b_k(t) - \sum_{k=i+1}^{j} b_k(0) & i < j \end{cases} \tag{7}$$

$\mu_{ij}(t) - Q_{ij}(t)$ is bounded by an upper limit which we represent as β_{ij}. $\mu_{ij}(t) - Q_{ij}(t) = \beta_{ij}$ implies that W_i is either blocked or starved depending on its relative location from W_j. If W_j is downstream of W_i, then W_i is blocked and if W_j is upstream of W_i, W_i is starved.

$$\beta_{ij} = \begin{cases} \sum_{k=j+1}^{i} b_k(0) & i > j \\ \sum_{k=i+1}^{j} B_k - \sum_{k=i+1}^{j} b_k(0) & i < j \end{cases} \tag{8}$$

One important implication of this boundary is that if $\mu_{ij}(t) - Q_{ij}(t) = \beta_{i,j}$ and $\hat{m}_i/T_i(t) > \hat{m}_j/T_j(t)$, W_j will constrain the speed of W_i. Here, $\frac{\hat{m}_i}{T_i}(t)$ and $\frac{\hat{m}_j}{T_j}(t)$ are the production speeds of workstation W_i and W_j respectively. Therefore, an operational workstation either works at it's own speed or at the speed of its constraining workstation. This can be mathematically represented as:

$$X_i(t+\delta t) - X_i(t) = \min \left\{ \zeta \left((X_i(t) - X_j(t) - Q_{ij}(t)) - \beta_{ij}(t), \frac{\hat{m}_j(t)}{T_j} \right), \frac{\hat{m}_i(t)}{T_i} \right\} \delta t \tag{9}$$

where,

$$\zeta(u, v) = \begin{cases} +\infty \text{ if } u < 0 \\ v \text{ if } u = 0 \end{cases}$$

Comparing to all workstations in the system,we have,

$$X_i(t+\delta t) - X_i(t) = \min \begin{cases} \zeta \left((X_i(t) - X_1(t) - Q_{i1}(t)) - \beta_{i1}(t), \frac{\hat{m}_1(t)}{T_1} \right) \\ \zeta \left((X_i(t) - X_2(t) - Q_{i2}(t)) - \beta_{i2}(t), \frac{m_2(t)/T_2}{T_2} \right), \\ \vdots \\ \frac{\hat{m}_i(t)}{T_i} \\ \vdots \\ \zeta \left((X_i(t) - X_w(t) - Q_{ij}(t)) - \beta_{ij}(t), \frac{\hat{m}_w(t)}{T_w} \right) \end{cases} * \delta t \tag{10}$$

$$= f_i(X_i(t), Q_i(t), \hat{m}_i(t)) * \delta t$$

$\hat{m}_i(t)$ is a function of $u_i(t), \theta_i(t)$ and $\tau_i(t)$ from (5). Since $u_i(t)$ and $\tau_i(t)$ are dependent on $\hat{\mathbf{u}}(t)$, we can write,

$$f_i(X_i(t), Q_i(t), \hat{m}_i(t)) = f_i(X_i(t), Q_i(t), \hat{u}_i(t), \theta_i(t)) \tag{11}$$

Extending the idea to all workstations, we have,

$$\mathbf{X}(t+\delta t) - \mathbf{X}(t) = \begin{cases} f_1(X_1(t), Q_1(t), \hat{u}_1(t), \theta_1(t)) \\ f_2(X_2(t), Q_2(t), \hat{u}_2(t), \theta_2(t)) \\ \vdots \\ f_w(X_w(t), Q_w(t), \hat{u}_w(t), \theta_w(t)) \end{cases} * \delta t \tag{12}$$

$$= \mathbf{F}\left(\mathbf{X}(t), \mathbf{Q}(t), \hat{\mathbf{u}}(t), \boldsymbol{\theta}(t)\right) \delta t$$

At every workstation, the defective parts are taken out of circulation. The quality of each product is modeled as two discrete states, compliant(1) or defective(0), with a probability of transition conditioned on the state of the tool processing it. The quality of any product produced in workstation W_i by robot R_j at time t is denoted by $q_j^i(t) \in \{0, 1\}$, and is given in the following probabilistic form:

$$p(q_j^i(t) | x_j^i(t)). \tag{13}$$

The worst tool state among all robots working in a workstation is used to determine quality. Consider $\hat{x}_i(t)$ such that,

$$\hat{x}_i(t) = \max\left(x_j^i(t)\right) \; i = 1, 2, \ldots, w, \; j = 1, 2, \ldots, r \quad \text{s.t} \; u_j(t) = i, \tag{14}$$

$\hat{x}_i(t)$ represents the worst tool state among all robots working in workstation W_i at time t. Therefore, quality of products from W_i can be modeled using the following probability distribution:

$$p(q_i(t) | \hat{x}_i(t)). \tag{15}$$

In order to find the total accumulated defective parts at the next time step, the number of defective parts resulted after production needs to be calculated. First, the total number of completed parts is calculated as $\lfloor X_i(t+\delta t) - X_i(t) + rem_i(t) \rfloor$ where $\lfloor * \rfloor$ is the floor operator which disregards numbers beyond the decimal and rem_i is decimal value that is excluded due to the floor operation. This remainder value is taken into account in the next time step. Then, for each of these completed parts, quality is accessed by sampling from (15) which results in either a value of 1 or 0 for each part. All the values that are 0's are counted since they are defective parts and all the 1's are ignored. Thus, the total number of defective parts at time t is given by,

$$Q_i(t+\delta t) = Q_i(t) + \sum_{k=1}^{\lfloor X_i(t+\delta t) - X_i(t) + rem_i(t) \rfloor} \left(1 - \left[k^0 q_i \sim p\left(q_i(t) | \hat{x}_i(t)\right)\right]\right) \tag{16}$$

where $\sim$ implies individual sampling from the distribution.

$$rem_i(t+\delta t) = X_i(t+\delta t) - X_i(t) + rem_i(t) - \left(\lfloor X_i(t+\delta t) - X_i(t) + rem_i(t) \rfloor\right) \tag{17}$$

The accumulated non-defective parts produced at each workstation until time t represented as $\tilde{PC}_i(t)$ can then be calculated as,

$$\tilde{PC}_i(t) = X_i(t) - Q_i(t) \tag{18}$$

The output of the line is the number of good parts at the end-of-the-line workstation,

$$\mathbf{Y}(t) = \tilde{PC}_w(t) \tag{19}$$

Authorized licensed use limited to: VIT University. Downloaded on January 22,2025 at 20:43:32 UTC from IEEE Xplore. Restrictions apply.

and the buffer levels can be updated,

$$b_{i+1}(t+\delta t) = \tilde{PC}_i(t) - X_{i+1}(t) + b_{i+1}(0) \quad (20)$$

Given consistent initial conditions, the model can be used to compute production counts of the system at any time recursively.

III. Control Problem

To improve the performance of a production system that is subject to robot failure, tool wear, and other downtimes, a real-time control strategy can be developed using the mathematical model proposed. However, the lack of closed-form equations makes conventional control techniques impractical. To overcome this issue, the problem is formulated as a Dec-POMDP and tackled with a Multi-Agent Reinforcement Learning (MARL) algorithm. This subsection provides details on the Dec-POMDP formulation and the specific MARL algorithm employed

A. Problem Formulation

In the Dec-POMDP framework, each robot agent's state changes as it interacts with the environment by performing actions chosen from an action space. The agent receives a reward based on the system's performance due to the action. The agent learns to choose actions conditioned on its observations, called the policy, which is optimized to maximize the discounted sum of rewards over an episode. The observations, actions, and reward definitions for the Dec-POMDP are presented below.

- *Observation and State:* The environment and robot behavior is derived from the developed model. Thus observation elements are chosen from model variables,

$$\mathbf{o}_i = [u_i, \theta_i, \tau_i, a_i, t_i^{rep}, \hat{\mathbf{x}}_\mathbf{j}^\mathbf{i}, b_j, b_{j+1}, \hat{m}_1, \hat{m}_2, \ldots, \hat{m}_w] \quad (21)$$

  where, j is the index of the workstation that the robot R_i is working in, i.e, $u_j(t) = i$. The state is represented by a concatenation of all $\mathbf{o}_i$s for $i = 1, 2, \ldots, r$,

$$\mathbf{S} = [\mathbf{o}_1, \mathbf{o}_2, \ldots, \mathbf{o}_r] \quad (22)$$

- *Action:*
  Each robot can choose from three types of actions as explained in Section II and a new action can only be taken after the previous one is completed. The action space can be represented as,

$$\mathscr{A} = \{1, 2, \ldots, w, (0,1), (0,2), \ldots, (0,l), (1,0), (1,1), (1,2), \ldots, (1,l)\}, \quad (23)$$

  where the first w elements are assignment actions and the remainder are maintenance/tool change actions in the form: (tool change, robot maintenance) with a value of 0 implying no tool change/no robot maintenance.
- *Reward:*
  One global reward determines how good the joint actions of all the agents are. A good reward setting takes domain knowledge into account to ensure the system behaves the way we want it to. The reward setting for a mobile multi-robot operated FMS was discussed in our previous work [16] where it was concluded that a reward setting based on the Permanent Production Loss (PPL) was most effective to control the system.

$$\mathscr{R}(t) = -PPL(t) * w_{PPL} + (\tilde{PC}_w(t) - Q_w(t))(1 - w_{PPL}). \quad (24)$$

  where $PPL(t)$ is the Permanent Production Loss of the system which is defined as the production that is lost due to effective disruption events and can never be recovered. w_{PPL} is a term which weighs the importance of PPL over part quality. This parameter is tuned to obtain the best performance from the system. The expression for PPL is as follows:

$$PPL(t) = D(t)\left(\frac{\hat{m}_{w^*,c}}{T_{w^*}} - \frac{\hat{m}_{w^*}(t)}{T_{w^*}}\right), \forall T > T^* \quad (25)$$

  where $D(t)$ is the accumulated amount of time the slowest workstation W_{w^*} is down/slowed down until time t.

B. Value Decomposition Actor Critic (VDAC)

The problem-solving algorithm employed is VDAC, an on-policy multi-agent actor-critic approach that involves a dedicated actor network for each agent. The actor network determines the optimal action from the perspective of the agent, while the critic network assesses the actual effectiveness of the actors' actions. The actor network is made up of one Recursive Neural Network (RNN) and three fully connected neural networks which generates a local value function, a local q-value function and a policy for each agent. The local value function and q-value is used by the critic network to obtain the gradient for optimization [13]. The algorithm uses the idea of centralized training and decentralized execution since the critic has access to all parameters during training but is removed during execution. The reason for the choice of this algorithm is twofold: 1) It works well with large state and action space which is the case for this problem. and 2) It allows for running parallel episodes which helps reduce training time and effectively utilize available computing resources.

IV. Case Study

A case study is conducted to demonstrate the effectiveness of the control formulation based on the developed model. Additionally, the performance of the integrated model based control is compared with two other control policies: one that conditions only on system parameters and another that conditions only on maintenance and quality parameters. These policies are analogous to controlling a manufacturing system without considering the integration of production control, maintenance scheduling, and product quality. The comparison is done using four metrics: Number of completed compliant parts(measured from the last workstation), summed compliant parts from all workstations, summed defective parts from all workstations and Permanent Production Loss. The observation definitions for the two benchmark policies are presented below.

- *System Model:* System model only includes features that pertain to the system but not maintenance scheduling or product quality. The observations of each robots for this model is:

$$\mathbf{o}_i = [u_i, \theta_i, b_j, b_{j+1}, \hat{m}_1, \hat{m}_2, \ldots, \hat{m}_w] \quad (26)$$

Authorized licensed use limited to: VIT University. Downloaded on January 22,2025 at 20:43:32 UTC from IEEE Xplore. Restrictions apply.

TABLE I
PARAMETERS FOR MOBILE MULTI-SKILLED ROBOT OPERATED FMS

	R_1	R_2	R_3	R_4	R_5	R_6
$Time_j^{PM,1}$	9	9	9	9	9	9
$Time_j^{PM,2}$	5	5	5	5	5	5
$Time_j^{PM,3}$	4	4	4	4	4	4
$Time_j^{CM}$	15	18	16	13	14	13
A_j	1000	800	900	1000	950	875
$MTBF_j$	200	300	400	250	200	300
$MTTR_j$	10	9	11	12	11	13

	W_1	W_2	W_3
T_i	1	0.9	1.2
$Time_i^c$	2	3	1

	B_2	B_3
B_i	6	8

All times shown are in minutes.

where j is the index of the workstation the robot R_j is working in.

- *Maintenance and Quality Model:* This model only includes parameters that pertain to quality and maintenance but not system level parameters. The observations of each robots for this model is:

$$\mathbf{o}_i = [\tau_i, a_i, t_i^{rep}, \hat{\mathbf{x}}_\mathbf{j}^\mathbf{i}, m_j] \tag{27}$$

To showcase the efficacy of the proposed integrated method, and to make a comparative analysis with the two non-integrated models, we conduct simulation experiments. These experiments employ multi-skilled robot operated FMS using diverse configurations; a total of 20 configurations which are generated by randomly and evenly selecting from the following sets:

$w \in \{3, 4, 5\},\ r \in \{6, 8, 10\}$

$B_i \in [4\ \ 10],\ T_i \in [0.8\ \ 1.5]$

$A \in \{500, 600, 700, 800, 900, 1000\}$

$Time^{PM,1} \in [8\ \ 10],\ Time^{PM,2} \in [5\ \ 7],\ Time^{PM,3} \in [1\ \ 4]$

$Time_j^{CM} \in [13\ \ 20],\ Time^c \in [1\ \ 2]$

$MTBF_j \in \{200, 250, 300, 350, 400\},\ MTTR_j \in [8\ \ 15]$

where $j = 1, 2, \ldots, r$ and $i = 2, 3, \ldots w$. All intervals presented increase with an integer value of 1 except for T_i which increases with a value of 0.1. The time for PM ($Time^{PM,k}$), maximum age (A_j), and time for tool change ($Time^c$) are kept constant across all robots. Random disruptions are generated assuming an exponential distribution using $MTBF_j$ and $MTTR_j$, and the time taken to move between consecutive workstations is set to $\psi = 1$.

The integrated model consistently exhibited superior performance and achieved more stable training outcomes compared to the non-integrated models across all tested configurations. To illustrate this, one representative configuration is provided below.

Consider a multi-skilled robot operated FMS with $w = 3$ workstations, $r = 6$ robots and 2 intermediate buffers. The system parameters for the manufacturing system are shown in Table I. To ensure robustness, the initial buffer level $b_i (i = 2, 3, \ldots w)$ is randomly selected between 0 and B_i, and 4 tool states are taken into account, i.e., $n^t = 4$. The transition model considered for the tool states is as follows:

$$p(x_j^i(t)|x_j^i(t-\delta t)) = \begin{bmatrix} 0.85 & 0.12 & 0.02 & 0.01 \\ 0.00 & 0.76 & 0.16 & 0.08 \\ 0.00 & 0.00 & 0.70 & 0.3 \\ 0.00 & 0.00 & 0.00 & 1.00 \end{bmatrix}$$

The product quality model considered in the experiment is as follows:

$$p(q_j^i(t) = 0|x_j^i(t)) = \begin{bmatrix} 0.01 & 0.38 & 0.67 & 0.99 \\ 0.01 & 0.50 & 0.73 & 0.99 \\ 0.01 & 0.40 & 0.59 & 0.99 \end{bmatrix}$$

The initial tool state for each episode is selected randomly, and three maintenance levels are considered: level 1 $(k = 1, r = 0)$, level 2 $(k = 2, r = 0.3)$, and level 3 $(k = 3, r = 0.6)$. The downtime caused by maintenance action of varying levels is presented in Table I. The initial age of each robot $a_j (j = 1, 2, \ldots, r)$ is randomly selected from 0 to A_j. To simulate random disruptions, an exponential distribution is used with parameters $MTBF_j$ and $MTTR_j$. The time required to move between consecutive workstations is assumed to be $\psi = 1$.

The model is trained by running several episodes until the policy is considered to have converged. To speed up the training, 20 episodes are run in parallel. Each episode represents a 10-hour workday with a time step (δt) of 1 minute. After every 10,000 steps (16.6 episodes), the policy is run for 96 test episodes. The test results include the average and standard deviation of several metrics. The actor network has a learning rate of 0.001 and the critic network has a learning rate of 0.0005. The batch size used for training is 20.

As shown in Fig. 2, the integrated model and system model converge at around 5×10^7 timesteps with a steady return, while the maintenance and quality model converge at 3×10^7

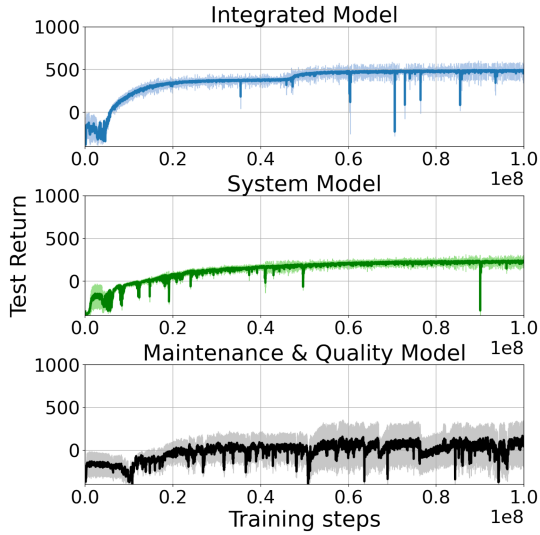

Fig. 2. Average test return vs. training steps for all three models. Shaded area represents standard deviation over test episodes.

Authorized licensed use limited to: VIT University. Downloaded on January 22,2025 at 20:43:32 UTC from IEEE Xplore. Restrictions apply.

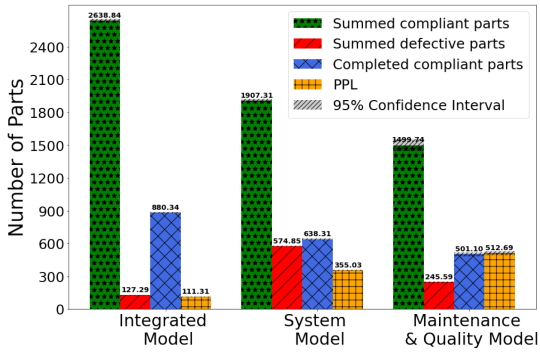

Fig. 3. Comparison of performance metrics among different models.

timesteps but exhibit a high standard deviation in test return due to the exclusion of important parameters.

Furthermore, as shown in Fig. 3, the integrated model outperforms the other two models, achieving a compliant part production of 880.34, a summed compliant part production of 2638.84, a summed defective part production of 127.29, and a PPL of 111.31. This is an increase of 242.03 and 379.24 in complete compliant parts, and 731.53 and 1139.1 in summed compliant parts from the system model, and maintenance and quality model respectively. The decrease in defective part production and PPL from the system model and maintenance and quality model are 447.56, 118.3 and 243.72, 401.38 respectively. Statistical analysis reveals that the performance increase of the integrated model is significant, as the 95% confidence intervals of the different models do not overlap. This success can be attributed to the model's inclusion of all important features, resulting in a highly effective control strategy.

V. Conclusion

In conclusion, this letter presents a mathematical control framework that integrates robot assignment, maintenance scheduling, and product quality, for a mobile multi-skilled robot operated manufacturing system. Results from several simulation studies show that the proposed integrated model-based control achieves greater training stability and higher performance compared to the other two control policies; one that is conditioned on just system parameters and one that is conditioned on only maintenance and quality parameters. Overall, the study highlights the importance of integrating production control, maintenance of equipment and quality control for achieving higher productivity in modern manufacturing systems.

Furthermore, the study identifies two highly intriguing avenues for future exploration. Firstly, the model can be enhanced by integrating additional processes, such as material handling, parallel lines, and product variety, to create a more comprehensive framework. Secondly, there is a need to incorporate a more precise depiction of how robot dynamics impact the total cycle time and, consequently, the overall system dynamics. This aspect, particularly relevant for systems employing collaborative robots (cobots), has received limited attention thus far, highlighting its significant importance for future research.

References

[1] Y. Liao, F. Deschamps, E. de F. R. Loures, and L. F. P. Ramos, "Past, present and future of industry 4.0: A systematic literature review and research agenda proposal," *Int. J. Prod. Res.*, vol. 55, no. 12, pp. 3609–3629, 2017.

[2] B.-H. Li, B.-C. Hou, W.-T. Yu, X.-B. Lu, and C.-W. Yang, "Applications of artificial intelligence in intelligent manufacturing: A review," *Front. Inf. Technol. Electron. Eng.*, vol. 18, no. 1, pp. 86–96, 2017.

[3] M. Colledani et al., "Design and management of manufacturing systems for production quality," *CIRP Ann.*, vol. 63, no. 2, pp. 773–796, 2014.

[4] J. Zou, Q. Chang, Y. Lei, and J. Arinez, "Event-based modeling and analysis of sensor enabled networked manufacturing systems," *IEEE Trans. Automat. Sci. Eng.*, vol. 15, no. 4, pp. 1930–1945, Oct. 2018.

[5] A. C. Bavelos et al., "Enabling flexibility in manufacturing by integrating shopfloor and process perception for mobile robot workers," *Appl. Sci.*, vol. 11, no. 9, 2021, Art. no. 3985.

[6] V. Ebrahimipour, A. Najjarbashi, and M. Sheikhalishahi, "Multi-objective modeling for preventive maintenance scheduling in a multiple production line," *J. Intell. Manuf.*, vol. 26, no. 1, pp. 111–122, Feb. 2015.

[7] A. Arab, N. Ismail, and L. S. Lee, "Maintenance scheduling incorporating dynamics of production system and real-time information from workstations," *J. Intell. Manuf.*, vol. 24, no. 4, pp. 695–705, Aug. 2013.

[8] B. Lu and X. Zhou, "Opportunistic preventive maintenance scheduling for serial-parallel multistage manufacturing systems with multiple streams of deterioration," *Rel. Eng. Syst. Saf.*, vol. 168, pp. 116–127, 2017.

[9] F. Ju, J. Li, and J. A. Horst, "Transient analysis of Bernoulli serial line with perishable products," *IFAC-PapersOnLine*, vol. 48, no. 3, pp. 1670–1675, 2015.

[10] M. Pearce, B. Mutlu, J. Shah, and R. Radwin, "Optimizing makespan and ergonomics in integrating collaborative robots into manufacturing processes," *IEEE Trans. Automat. Sci. Eng.*, vol. 15, no. 4, pp. 1772–1784, Oct. 2018.

[11] X. Zhao, B. Tao, and H. Ding, "Multimobile robot cluster system for robot machining of large-scale workpieces," *IEEE/ASME Trans. Mechatron.*, vol. 27, no. 1, pp. 561–571, Feb. 2022.

[12] H. Ye, X. Wang, and K. Liu, "Adaptive preventive maintenance for flow shop scheduling with resumable processing," *IEEE Trans. Automat. Sci. Eng.*, vol. 18, no. 1, pp. 106–113, Jan. 2021.

[13] J. Su, J. Huang, S. Adams, Q. Chang, and P. A. Beling, "Deep multi-agent reinforcement learning for multi-level preventive maintenance in manufacturing systems," *Expert Syst. Appl.*, vol. 192, 2022, Art. no. 116323.

[14] K. Bhatta, J. Huang, and Q. Chang, "Dynamic robot assignment for flexible serial production systems," *IEEE Robot. Automat. Lett.*, vol. 7, no. 3, pp. 7303–7310, Jul. 2022.

[15] M. Kijima, "Some results for repairable systems with general repair," *J. Appl. Probability*, vol. 26, no. 1, pp. 89–102, Mar. 1989.

[16] K. Bhatta and Q. Chang, "An integrated control strategy for simultaneous robot assignment, tool change and preventive maintenance scheduling using heterogeneous graph neural network," *Robot. Comput.- Integr. Manuf.*, vol. 84, 2023, Art. no. 102594.

Authorized licensed use limited to: VIT University. Downloaded on January 22,2025 at 20:43:32 UTC from IEEE Xplore. Restrictions apply.