Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.
50th SME North American Manufacturing Research Conference (NAMRC 50, 2022)

Production Loss Analysis in Mobile Multi-skilled Robot Operated Flexible Serial Production Systems

Kshitij Bhatta, Chen Li, Qing Chang*

Department of Mechanical and Aerospace Engineering, University of Virginia, Charlottesville, VA, 22904, USA

* Corresponding author. E-mail address: qe9nq@virginia.edu

Abstract

In a world where demand and supply can change instantaneously as demonstrated by the recent coronavirus pandemic, it is very important for our production lines to be able to handle abrupt changes very effectively. Flexible manufacturing systems (FMS) are designed to be able to ramp production up and down quickly to make this possible. In this paper, we propose a novel data-enabled method of modelling an FMS using mobile multi-skilled robots and evaluate its dynamic performance. We use permanent production loss as the performance metric and derive expressions for its evaluation and attribution and validate them using two case studies.

1. Introduction

Besides improving the ability of a manufacturing system to cope with internal and external disturbances [1], customizability, faster speed of delivery and future proofing are yet other needs of the current manufacturing industry especially with concepts such as Industry 4.0 and Industry 5.0 slowly becoming reality. Thus, manufacturing environments must be designed to cater to new challenges to survive and grow in the marketplace. This requirement of increased product variety, reduced manufacturing lead times, increased quality standards and competitive costs have been the key rationale for the development of flexible manufacturing systems (FMS) [2]. The first widely accepted definition of FMS was published by Browne et al. in his 1984 paper, ‘Classification of Flexible Manufacturing Systems’. He defined FMS as an integrated system that can process ‘medium-sized volumes of a variety of part types’, and such system possesses the flexibility in machine, process, and operation, etc [3]. As seen in the definition, FMS is not a very rigidly defined system since there is not a consensus on what ‘flexibility’ means. There could be several approaches on what flexibility means: the capability of producing different parts without major retooling; the ability to change a production schedule, to modify a part, or to handle multiple parts; or the ability to rapidly increase or decrease production levels or to shift capacity quickly from one product or service to another. Despite the flexibility of the term ‘flexible’, one thing that has been accepted as a key characteristic of FMS is its ability to adapt to changes and respond to various requirements with little penalty in time, effort, cost, or performance [4].

There have been many innovative designs of FMS [5]–[8], a key example being Toyota’s flexible assembly lines which have the ability to produce multiple vehicle models in any production sequence. Their FMS is able to respond to changes in market demand, shorten the lead time in production, and help
in production planning and production ordering [5]. [9] discusses the use of Autonomous Guided Vehicles (AGVs) for material handling to increase flexibility of a manufacturing system and the role of automation in FMSs has been discussed in [10]. Petri nets have been used to model and control a FMS in [8], [11]. Multi-skilled robots have also been used in flexible manufacturing systems. These are robots capable of performing multiple tasks through the use of multifunctional manipulators or quick-change end effectors. When mobile, such robots allow for flexibility to perform various tasks in different workstations spread throughout the plant floor [12]. Even though there has been quite a bit of research on flexibility of manufacturing systems [1], [2], [4]–[8], [12], there has been minimal performance evaluation of such systems. Traditional analytical and simulation methods focus mostly on steady state dynamics of manufacturing systems using simplistic models (e.g. Bernoulli process) [13]–[15] while the available literature on transient dynamics only consider fixed configuration production systems [16]–[20]. Thus, we propose a novel data-enabled approach to modelling an FMS using mobile multi-skilled robots and derive expressions for transient performance metrics: real-time permanent production loss (PPL) evaluation and attribution. The novelty of this approach comes from the fact that unlike other models for an FMS, this one uses a real time performance metric, i.e., PPL, to determine how well the system is doing at any given time and therefore makes control design easier.

The remainder of the paper is organized as follows. Section 2 introduces the notations and assumptions applied in the paper. In Section 3, a data-enabled mathematical model of the proposed FMS is developed. Opportunity window evaluation, permanent production loss evaluation and permanent production loss attribution for the production line model is discussed in section 4. Based on previous sections, a numerical case study is demonstrated in Section 5 and finally Section 6 summarizes the work in a brief conclusion and mentions future work.

2. System Description

We consider a flexible serial production line with \( w \) workstations and \( w - 1 \) intermediate buffers. Here, each workstation is operated entirely by mobile multi-skilled robots which can move from one workstation to another. Fig.1. illustrates the model where workstations are represented by rectangles and buffers by circles. Each workstation accommodates a certain number of robots which can freely move from one workstation to another. The following notation is adopted in the model:

1. \( S \) is the set of all workstations in the production line such that each element \( S_i \) denotes the \( i^{th} \) workstation, where \( i = 1, 2, ..., w; \)
2. \( R \) is the set of all robots in the production line such that, each element, \( R_j \) denotes the \( j^{th} \) robot in the production line, where \( j = 1, 2, ..., r; \)
3. Each workstation relies entirely on multi-skilled robots for production, so we define \( S_i(t) \subseteq R \) as the set of all robots assigned to workstation \( i \) at time \( t \) where \( i = 1, 2, 3, ..., w \). We use an \( r \times 1 \) indicator vector, \( \mathbf{u}_i(t) = [u_{i1}(t), ..., u_{ir}(t)]^T \) to denote robot assignment of each workstation \( i \) at time \( t \). Each element \( u_{ij}(t) = 1 \) if \( R_j \in S_i(t) \) and 0 otherwise, where \( i = 1, 2, ..., w \) and \( j = 1, 2, ..., r \). All the indicator vectors can be merged to form the assignment matrix, \( U(t) = [\mathbf{u}_1, ..., \mathbf{u}_w] \) which represents the assignment of all robots in the production line.
4. The total number of robots assigned to a workstation is defined by \( m_i(t) = |S_i(t)| \), where \( |S_i(t)| \) is the cardinality of the set. This number has an upper limit of \( m^{[i]} \) due to workstation size and user preference. Therefore, \( m_i(t) \leq m^{[i]} \forall t; \)
5. Each workstation has a specified base cycle time \( T_i \), where \( i = 1, 2, ..., w \). This is the time workstation \( i \) takes to produce one part with only a single robot working. However, since each workstation can have more than one robot working, the total cycle time for each workstation at any time \( t \) is \( \frac{T_i}{m_i(t)} \) where \( m_i(t) \) is the number of robots working in workstation \( i \) at time \( t \). This proportionality is a simplification for ease of mathematical formulation. A different relation between base cycle time and number of robots working does not change the model in any way. Note that \( \frac{T_i}{m_i(t)} \leq \frac{T_i}{m_i(t)} \forall t \) because it is not necessarily true that all assigned robots will be working.
6. The \( 1 \times w \) vector of \( m_i(t) \) and \( \overline{m}_i(t), \mathbf{M}(t) \) and \( \overline{\mathbf{M}}(t) \) respectively are referred to as the assignment configuration and working configuration of the production line at time \( t \).
7. The configuration of the production line in the absence of disruption events is called the ideal clean configuration, \( \mathbf{M}_c = [m_{c,1}, ..., m_{c,w}] \). Similarly, \( U_c \) is the ideal clean assignment matrix of the system.
8. Each buffer has a finite capacity denoted by \( B_l \) where \( l = 2, 3, ..., w; \)
9. \( \mathbf{b}(t) = [b_2(t), b_3(t), ..., b_w(t)]^T \) are the buffer levels at time \( t \);
10. \( \overline{e}_{ij} = (j, t_i, d_j) \) denotes the \( i^{th} \) disruption event where robot \( j \) is down at time \( t_i \) for \( d_j \) time period, where \( i = 1, 2, ..., n \) and \( j = 1, 2, ..., r \). \( n \) represents the total number of disruption events;
11. \( E = [\vec{e}_i, ..., \vec{e}_n] \) denotes a sequence of disruption events;
12. \( MTBF_i \) denotes the mean time between failure of robot \( R_j \) and \( MTTR_{ij} \) denotes the mean time to repair;
13. \( S_m(t) \) denotes the slowest workstation and \( T_m^t \) denotes the cycle time for the slowest workstation.

We make the following assumptions in the paper:
1. The total number of robots in the system remains constant, i.e., \( \sum_{i=1}^{m_i} m_i(t) = r \).
2. A robot is considered working in workstation \( i \) if it is assigned to the workstation and is operational.
3. A workstation is down if there are no robots working on it, i.e., \( m_i = 0 \) and slowed down if the number of robots working on it is less than the number of robots assigned to it, i.e., \( m_i \leq m_i \).
4. A workstation is partially blocked if it is operational, its downstream buffer is full and its downstream workstation is slowed down and fully blocked if it is operational, its downstream buffer is full, and its downstream workstation is down.
5. A workstation is partially starved if it is operational, and its upstream buffer is empty and its upstream workstation is slowed down and fully starved if it is operational, and its upstream buffer is empty, and its upstream workstation is down.
6. The first workstation, \( S_1 \) is never starved and the last workstation, \( S_n \) is never blocked since the performance of the isolated production line is to be studied.

3. Data-Driven Mathematical Model

A manufacturing system is a dynamic system operating under control inputs and random disturbances. Thus, it can be modelled using a state space equation. However, since we have two different dynamics involved: Robot motion and workstation dynamics in part production, we need to define two state space equations and one output equation. Here, we will be considering the assignment matrix \( U = [u_1, ..., u_n] \) as our control input and the production count of the final product as the output.

3.1. State Equation for Robot Motion

\[
\dot{X}_R(t) = \bar{F} \left( X_R(t), X_{Rf}(t), W(t), \tau_{move}(t) \right)
\] (1)

Each component of Eq. (1) is defined as:

- \( X_R(t) = [X^1_R(t), X^2_R(t), ..., X^n_R(t)]^T \), where \( X^j_R(t) \) is the distance of the \( j \)th robot from a reference which determines the workstation the robot is currently working on.
- \( \bar{F}(\cdot) \) is the function that defines each robot’s movement.
- \( W(t) = [W_1(t), W_2(t), ..., W_r(t)]^T \) is the robot status variable at time \( t \) which determines if robot \( j \) suffers a disruption event. \( W_j(t) = 1 \) if robot \( j \) is down at time \( t \) and \( W_j(t) = 0 \) if it is operational.

- \( X_{Rf}(t) = [X^1_{Rf}(t), X^2_{Rf}(t), ..., X^n_{Rf}(t)]^T \), where \( X^j_{Rf}(t) \) is the final assignment of robot \( j \) at time \( t \). It determines the final position of \( j \)th robot at time \( t \) and changes due to system control action.
- \( \tau_{move}(t) = \left[ \tau_{move,1}(t), \tau_{move,2}(t), ..., \tau_{move,r}(t) \right]^T \) is the time it takes for robot \( j \) to get to \( X^k_{Rf} \) from \( X^k_R \) at time \( t \).

Using Eq. (2) we can solve for the \( \tau_{move}(t) \) of all robots:

\[
\tau_{move}(t) = \left( \frac{X^1_{Rf}(t) - X^1_R(t)}{X^1_R(1 - W_1(t))} \right) \left( \frac{X^2_{Rf}(t) - X^2_R(t)}{X^2_R(1 - W_2(t))} \right) \cdots \left( \frac{X^n_{Rf}(t) - X^n_R(t)}{X^n_R(1 - W_n(t))} \right)
\] (3)

The robot velocity here is considered a system parameter and the term \( (1 - W_i(t)) \) is added to account for the fact that it takes an indeterminate amount of time for a robot to get to its final assignment if it is down.

3.2. State Equation for Part Production

\[
\dot{X}_W(t) = F \left( X_W(t), \bar{M}(t) \right)
\] (4)

Where each component of Eq. (4) is defined as:

- \( X_W(t) = [X^1_W(t), X^2_W(t), ..., X^n_W(t)]^T \), where \( X^j_W(t) \) is the production count of the \( i \)th workstation up until time \( t \).
- \( F(\cdot) \) here is a function that defines the production dynamics of each workstation.
- \( \bar{M}(t) = [\bar{m}_1(t), \bar{m}_2(t), ..., \bar{m}_n(t)]^T \), where \( \bar{m}_i(t) \) is the number of robots working in the \( i \)th workstation at time \( t \).
- To account for the time delay caused by the movements of robots in the assigned workstation, we define a vector \( \tau_{delay}(t) = [\delta_1(t), \delta_2(t), ..., \delta_n(t)]^T \). Here, each \( \delta_i(t) = \delta_k(\tau_{move,j}(t),0) \) where \( \delta_k \) is the Kronecker delta which is defined as \( \delta_k(u,v) = 1 \) if \( u = v \) and 0 otherwise.
We define a $r \times 1$ vector $\Theta(t) = u_i(t) \circ \Theta(t)$ called the working robot status where $\circ$ is the Hadamard product operator and $I_{r \times 1}$ is a $r \times 1$ vector of ones. $\Theta_i(t) = \Theta(t) \circ [I_{r \times 1} - W_i(t)] \circ \delta(t)$ is the working robot status where $\circ$ is the Hadamard product operator and $I_{r \times 1}$ is a $r \times 1$ vector of ones.

$\Theta_i(t) = [\theta_{1}(t), \ldots, \theta_{r}(t)]$ where $\theta_{j}(t) = 1$, if robot $j$ is working in workstation $i$ at time $t$ and 0 otherwise.

$\Theta(t) = [\Theta_1(t), \ldots, \Theta_w(t)]$ is the working robot status matrix which comprises of the working robot status of the production line.

The number of robots working in workstation $S_i$ can be calculated by $m_i(t) = \sum \Theta_i(t)$ and the number of robots assigned to workstation $S_i$ can be calculated by $m(t) = \sum U_j \Theta_j(t)$.

The accumulated production difference between two workstations $S_i$ and $S_j$ with $i < j$ is defined by $\tau_{ij}(t) = X_{i}^{W}(t) - X_{j}^{W}(t)$

This quantity is bounded by a condition that all buffers between $S_i$ and $S_j$ might get full $(i < j)$ or empty $(i > j)$. If we call this boundary $\beta_{ij}$,

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

Extending the same idea to all the workstations, we have

$X_{i}^{W}(t) = \min \left\{ \frac{\zeta \left( X_{i}^{W}(t) - X_{j}^{W}(t) \right) - \beta_{ij} \cdot \overline{m}_i(t)}{T_1}, \frac{\zeta \left( X_{i}^{W}(t) - X_{j}^{W}(t) \right) - \beta_{ij} \cdot \overline{m}_j(t)}{T_2} \right\}$

$\tau_{ij}(t) = X_{i}^{W}(t) - X_{j}^{W}(t) - \beta_{ij} \cdot \overline{m}_i(t)$

The buffer level in any buffer $B_{i+1}$ at time $t$ can be calculated as:

$\beta_{i+1}(t) = X_{i+1}^{W}(t) + X_{i+1}^{W}(t) + b_{i+1}(0)$

Since we focus on the modelling and performance evaluation of the FMS, we assume that the control $U(t)$ is known. So, we can calculate $X_{t_f}^{R}(t)$ by first calculating the matrix $\Delta U = U(t_k) - U(t_{k-1})$. Here $t_k$ represents the time when the $k$th control action is taken. Sum of all rows in $\Delta U$ sum to zero since the same robot cannot be in more than one workstation. For each row which represents a robot, the column with -1 is the workstation where it currently is and the column with +1 is the workstation it needs to go to. So, $X_{t_f}^{R}(t)$ of the robots with -1 values is updated to the distance of the workstation with the +1 value every time a control action is taken.

### 4. System Performance Evaluation

#### 4.1. Ideal Clean Case

The mathematical model in Eq. (1)-(12) represents a robot operated flexible production line subjected to disruption events. However, to accurately evaluate system performance, it is necessary to choose a reference to which the system in
This clean case model describes a virtual scenario that no random disruptions occur in a production line. In such a scenario, the system will reach a steady state where the production rate of each machine and buffer level of each buffer is constant, i.e., \( \bar{m}_c(t) = \bar{m}_c \). One thing to note is that for the ideal clean configuration is constant, i.e., \( \bar{M}_c(t) = \bar{M}_c \). One thing to note is that for the ideal clean case, \( \bar{M}(t) = \bar{M}(t) \) and \( \bar{U}(t) = \bar{U}(t) \) since all assigned robots are working. The Ideal clean case can be obtained by optimizing the control input, \( \bar{U} \) for maximum throughput, \( X_W(t) \) over the time horizon \( [0, T] \) which will not be discussed here in detail.

### 4.2. Permanent Production Loss (PPL) Evaluation

For serial production lines with finite buffer capacities, permanent production loss is an extremely valuable metric to gauge the transient performance of the system. This is because, due to variation of cycle time in each machine/workstation, there can be disruption events which do not cause permanent production loss and such transient events are left out by steady state analytic metrics like system throughput.

**Definition 1.** The permanent production loss of a manufacturing system can be defined as the difference between the ideal clean case output, \( Y_c(T) \) and the real output, \( Y(T) \) that cannot be recovered.

\[
PPL_i = Y_c(T) - Y_i(T) = PPL_{i, \text{cont}} + PPL_{i, \bar{e}}
\]

Permanent production loss can be further divided into two parts, permanent production loss due to control, \( PPL_{i, \text{cont}} \) and permanent production loss due to disruption events, \( PPL_{i, \bar{e}} \). The permanent production loss due to control is caused by the inefficiency of the control algorithm used and will not be discussed in this paper. \( PPL_{i, \bar{e}} \) is caused due to the presence of disruption events which cause downtime in robots. In order to focus on \( PPL_{i, \bar{e}} \), we will define a pre-determined control scheme which prescribes an \( U(t) \) for every \( t \) in \([0, T]\). This control scheme is randomly generated within constraints of constant robot number (assumption 1) and maximum robots within each workstation (notation 4). Henceforth, unless otherwise specified, \( PPL \) refers to the permanent production loss due to disruption events.

Evaluation of production loss can be done based on the status of the last slowest machine as explained in [17]–[19]. If the last slowest machine is partially or fully blocked/starved, permanent production loss will start to accrue. Before we define \( PPL \) mathematically, it is important to understand the concept of opportunity window. Opportunity window of a workstation \( S_j \), denoted as \( OW_j(T_d) \), is the longest possible downtime/slow-down at time \( T_d \) that would not result in permanent production loss at the end of the line machine. It is defined as:

\[
\begin{align*}
OW_j(T_d) &= \sup\{d \geq 0 : s.t. \exists T^*(d) \} \\
\int_0^T S_j(t) dt' &= \int_0^T S_j(t', \bar{e}) dt', \forall T' \geq T^*(d)
\end{align*}
\]

Previous studies [17]–[20] have shown that for a serial production line with finite buffer capacity, the opportunity window of workstation \( S_j \) is the time it takes for the buffers between \( S_j \) and the last slowest workstation \( S_{w^*} \) to become full/empty when \( S_j \) is down.

Further, the same studies have also shown that only disruption events longer than the opportunity window cause permanent production loss. If the duration of a disruption event \( \bar{e} = (i, t, d) \) is greater than the \( OW_j(t) \) of workstation \( R_i \in S_j(t) \), then for any workstation \( S_j \) in the line, there exists \( T^* \geq t + d \), which may depend on the relative location of workstation \( S_j \) with respect to the slowest workstation \( S_{w^*} \), such that:

\[
\begin{align*}
\int_0^T S_j(t') dt' - \int_0^T S_j(t', \bar{e}) dt' &= \left( \frac{d - OW_j(t)}{T_{w^*}} \right) m_{w^*} \\
\forall T > T^*
\end{align*}
\]

However, this is only true if the disruption event causes a downtime. In our case, disruption events can cause slowdowns as well. To take this into account, opportunity window can alternatively be defined as the time it takes the slowest workstation \( S_{w^*} \) to slow down to the disrupted workstation’s \( S_j \) speed.

\[
\begin{align*}
\int_0^T S_j(t') dt' - \int_0^T S_j(t', \bar{e}) dt' &= \left( \frac{d - OW_j(t)}{T_{w^*}} \right) \left( m_{w^*} - \bar{m}_j \right) \\
\forall T > T^*
\end{align*}
\]

Since our control input is the assignment matrix \( U(t) \), every time the system undergoes a control action, the slowest machine changes (due to the change in \( \bar{m}_j \)). To take such a condition into consideration, [18] has developed the notion of slowest machine switching interval. The assumption is that the slowest machine switches \( K \) times during a time interval \([0, T]\)
with the time point at which this change happens being \( t_k, k = 1, 2, \ldots, K \). This allows for the time period to be segmented into time intervals \([t_k, t_{k+1}],[t_{k+1}, t_{k+2}], \ldots\). This notion can be extended to our model as well. In our model each slowest machine switch is associated with the change in assignment where the slowest workstation switches, and the time of switch \( t_k \) is the time at which the \( k^{th} \) control action is taken.

Using this concept, we can derive the expression for \( \text{PPL} \) of workstation \( S_j \) in the production line subject to assignment changes during \([0, T]\) as the sum of production losses over all slowest machine switching intervals:

\[
\int_0^T S_j(t') dt' - \int_0^T S_j(t', \tilde{e}) dt' = \sum_{k=1}^K (d_k - OW_p(t_k)) \left( \frac{m_{w,k}}{T_{w'}} - \frac{m_{l,k}}{T_f} \right) \quad \forall T > T^*
\]  

### 4.3. Permanent Production Loss (PPL) Attribution

In a deterministic scenario where we know when the disruption events occur and which robots they happen in, we can attribute the permanent production loss to each event, robot and each workstation. Attribution to each event helps understand which event has the biggest impact in production and the attribution to each robot and workstation elucidates the idea that there are weak links in the production line, the disruption of which cause the most permanent production loss. Therefore, using this attribution we can rank each robot and workstation based on their capability to cause permanent production loss.

Given a sequence of disruption events \( E = [\tilde{e}_1, \ldots, \tilde{e}_n] \) in an interval \([t_a, t_b]\), the PPL attributed to each disruption event \( \tilde{e} = (p, t_p, d_i) \), \( 1 \leq i \leq n \), \( p \in S_j(t) \), is the total sum of PPL caused by that event in each slowest machine switching interval. That can be summarized as:

\[
P_{L_{\tilde{e}i}}[t_a, t_b] = \sum_{k=1}^K P_{L_{\tilde{e}i}}[tt_k, tt_{k+1}]
\]

Now, within each machine switching interval \([tt_k, tt_{k+1}]\), PPL is attributed to the event whose corresponding workstation has the smaller opportunity window and divided equally if more than one disruption event corresponding to the same smallest opportunity window [18]. Further, if more than one robot from the same workstation is down, PPL is attributed to each robot equally since both are identical when working at the same workstation.

\[
P_{L_{\tilde{e}i}}[tt_k, tt_{k+1}] = \begin{cases} 
\max\{0, (tt_{k+1} - tt_k) \left( \frac{m_{w,k}}{T_{w'}} - \frac{m_{l,k}}{T_f} \right) - \psi \} & \text{Cond. A} \\
0, & \text{Cond. B}
\end{cases}
\]

Where \( \psi = X_{ji}^W(tt_k) - X_{ji}^W(tt_k) + \beta_{ji,w}^* \), condition A is

\[
\min_1 \left\{ \begin{array}{l}
X_{ji}^W(tt_k) + \beta_{ji,w}^* \\
X_{ji}^W(tt_k) + \beta_{ji,w}^* \\
\vdots \\
X_{ji}^W(tt_k) + \beta_{ji,w}^*
\end{array} \right\} = X_{ji}^W(tt_k) + \beta_{ji,w}^* 
\]

\[ \rho \] is the total number of workstations satisfying the condition during \([tt_k, tt_{k+1}]\), and condition B is

\[
\min_1 \left\{ X^W_{ji}(tt_k) + \beta_{ji,w}^* \right\} < X^W_{ji}(tt_k) + \beta_{ji,w}^* 
\]

Given a sequence of disruption events that occur in robot \( R_j \) denoted as \( \tilde{e}_{j1}, \ldots, \tilde{e}_{jn} \) during time period \([0, T]\), attribution to each robot can be done by summing up the permanent production loss of all events corresponding to the robot.

\[
P_{L_{j}}[0, T] = \sum_{q=1}^n P_{L_{\tilde{e}j,q}} \text{ for } j = 1, 2, \ldots, r
\]

Attribution to each workstation is complex since robots move from one workstation to another after every control action. This means the set of robots in workstation \( j \) is dependent on the machine switching interval, \( k \). We call \( S_{j,k} \), the set of all robots in \( j^{th} \) workstation at the \( k^{th} \) machine switching interval. The PPL attributed to each workstation can be calculated as the sum of permanent production loss of all the disrupted robots belonging to that workstation in the time horizon \([0, T]\).

\[
P_{L_{j}}[0, T] = \sum_{q=1}^n \sum_{k=1}^K \sum_{R_{j,k}} P_{L_{\tilde{e}j,q}}[tt_k, tt_{k+1}]
\]
5. Case Study

To demonstrate the effectiveness of the proposed PPL evaluation and attribution methods, two studies are performed in generally reliable 50 different production lines, parameters for which are chosen from practical industries, such as automotive assembly lines, semi-conductor production lines, battery lines etc. The first study is done to verify PPL evaluation using Eq. (20) and the second study is done to validate the proposed PPL attribution method. For the first study we find $PPL_{I,E}$ using a recursive simulation based on the conservation of flow and Eq. 20 from the mathematical model. The values must be consistent for validation. For the second study, we first record the output of each workstation/robot with the presence of disruption events and then one by one remove disruption events on individual workstations/robots. The idea here is that if the PPL is attributed correctly, the overall performance improvement due to the removal of disruption events in simulation will be equal or very close to the PPL calculated using Eq. (23) and Eq. (24) in the mathematical model.

To demonstrate the validation process, a segment of a robot assembly line consisting of 5 workstations, 4 buffers and 10 robots is considered. The data used in the process are mocked up for confidential consideration. Workstation parameters, buffer parameters and robot parameters are presented in table 1, 2 and 3 respectively.

| Workstation Number | $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ |
|--------------------|-------|-------|-------|-------|-------|
| Base Cycle Time    | 4     | 5     | 6     | 7     | 5     |

| Buffer Number | $B_1$ | $B_2$ | $B_3$ | $B_4$ |
|---------------|-------|-------|-------|-------|
| Initial Buffer Level | 2     | 3     | 3     | 12    |
| Buffer Capacity | 12    | 48    | 32    | 15    |

A simulation is run for 5000 minutes to accurately replicate real-world manufacturing plant schedule. The ideal clean case is first found using an exhaustive search due to the small size of the line. Then, a pre-determined control scheme is used with no disruption events, and the final output is recorded. This case, we will call the “Real clean case”. Finally, randomly generated disruption events are added along with the control scheme. The disruption events are generated using an exponential distribution of MTTR and MTBF of individual robots. This is the real output. $PPL_{I,E}$ is calculated using Eq. (16) from simulation and this result is compared to the PPL calculated using Eq. (20) from the mathematical model. The results shown in Fig. 2. clearly show $PPL_{I,E} = 54.07$ and using Eq. (20) we get a value of 53.06 which is very close. Therefore our PPL evaluation method is validated. Another important conclusion drawn from this study is that the control has a very big impact on the total PPL of the system. This conclusion will lead into our future work using this model.

$PPL$ attribution is also validated as seen in Fig. 3. and Fig. 4. The performance improvement (in orange) is very close to the $PPL$ attributed to individual robot/workstation (in blue) which validates our method.

6. Conclusion and Future Work

In this paper, a novel flexible manufacturing system model using multi-skilled robots is proposed and expressions for Permanent production loss evaluation and attribution are derived. These expressions are then validated using two case studies. An important conclusion drawn from the case study is that control has a big impact on the $PPL$ of the system. Thus, our future work will include Permanent production loss prediction using the model and a real-time control strategy utilizing sensor data and the expressions derived here. Control strategies such as Model Predictive Control (MPC) and distributed multi agent control as well as reinforcement
learning methods will be considered for establishing an effective control framework.

Acknowledgement

This work is supported by the National Science Foundation Grants 1853454

References

[1] A. Jain, P. K. Jain, F. T. S. Chan, and S. Singh, “A review on manufacturing flexibility,” *Int. J. Prod. Res.*, vol. 51, no. 19, pp. 5946–5970, Oct. 2013, doi: 10.1080/00207543.2013.824627.

[2] A. Singholi, “Towards improving the performance of flexible manufacturing system: a case study,” *M. Ali*, p. 30, 2010.

[3] J. Browne, D. Dubois, S. Sethi, K. Rathmill, and K. Stecke, “Classification of Flexible Manufacturing Systems,” Social Science Research Network, Rochester, NY, SSRN Scholarly Paper ID 3599105, 1984. Accessed: Nov. 15, 2021. [Online]. Available: https://papers.ssrn.com/abstract=3599105.

[4] A. De Toni and S. Tonchia, “Manufacturing flexibility: A literature review,” *Int. J. Prod. Res.*, vol. 36, no. 6, pp. 1587–1617, Jun. 1998, doi: 10.1080/002075489193183.

[5] A. Masuyama, “Idea and practice of flexible manufacturing systems of toyota,” in *Manufacturing Research and Technology*, vol. 23, A. Raouf and M. Ben-Daya, Eds. Elsevier, 1995, pp. 305–316. doi: 10.1016/S1572-4417(06)80015-X.

[6] S. Huang, K. Shinya, N. Bergström, Y. Yamakawa, T. Yamazaki, and M. Ishikawa, “Dynamic compensation robot with a new high-speed vision system for flexible manufacturing,” *Int. J. Adv. Manuf. Technol.*, vol. 95, no. 9–12, pp. 4523–4533, Apr. 2018, doi: 10.1007/s00170-017-1491-7.

[7] S. C. Park, “A methodology for creating a virtual model for a flexible manufacturing system,” *Comput Ind.*, 2005, doi: 10.1016/j.compind.2005.04.002.

[8] L. Hu, Z. Liu, W. Hu, Y. Wang, J. Tan, and F. Wu, “Petri-net-based dynamic scheduling of flexible manufacturing system via deep reinforcement learning with graph convolutional network,” *J. Manuf. Syst.*, vol. 55, pp. 1–14, Apr. 2020, doi: 10.1016/j.jmsy.2020.02.004.

[9] F. Yao, B. Alkan, B. Ahmad, and R. Harrison, “Improving Just-in-Time Delivery Performance of IoT-Enabled Flexible Manufacturing Systems with AGV Based Material Transportation,” *Sensors*, vol. 20, no. 21, Art. no. 21, Jan. 2020, doi: 10.3390/s20216333.

[10] B. Johansson et al., “Enabling flexible manufacturing systems by using level of automation as design parameter,” in *Proceedings of the 2009 Winter Simulation Conference (WSC)*, Dec. 2009, pp. 2176–2184. doi: 10.1109/WSC.2009.5429728.

[11] K.-Y. Xing, B.-S. Hu, and H.-X. Chen, “Deadlock avoidance policy for Petri-net modeling of flexible manufacturing systems with shared resources,” *IEEE Trans. Autom. Control*, vol. 41, no. 2, pp. 289–295, Feb. 1996, doi: 10.1109/9.481550.

[12] G. Michalos, S. Makris, P. Aivaliotis, S. Matthaiakis, A. Sardelis, and G. Chryssoulouris, “Autonomous Production Systems Using Open Architectures and Mobile Robotic Structures,” *Procedia CIRP*, vol. 28, Dec. 2015, doi: 10.1016/j.procir.2015.04.020.

[13] J. Li, “Overlapping decomposition: a system-theoretic method for modeling and analysis of complex manufacturing systems,” *IEEE Trans. Autom. Sci. Eng.*, 2005, doi: 10.1109/TASE.2004.835576.

[14] S. B. Gershwin, *Manufacturing systems engineering*, Englewood Cliffs, N.J.: PTR Prentice Hall, 1994.

[15] Y. Bai, J. Tu, M. Yang, L. Zhang, and P. Denno, “A new aggregation algorithm for performance metric calculation in serial production lines with exponential machines: design, accuracy and robustness,” *Int. J. Prod. Res.*, vol. 59, no. 13, pp. 4072–4089, Jul. 2021, doi: 10.1080/00207543.2020.1757777.

[16] J. Zou, Q. Chang, J. Arinez, and G. Xiao, “Analysis and Evaluation of Control Action End-State Impact on Manufacturing System Future Production,” *IEEE Access*, vol. 6, pp. 60187–60197, 2018, doi: 10.1109/ACCESS.2018.2875331.

[17] J. Zou, Q. Chang, J. Arinez, and G. Xiao, “Data-driven modeling and real-time distributed control for energy efficient manufacturing systems,” *Energy*, vol. 127, pp. 247–257, May 2017, doi: 10.1016/j.energy.2017.03.123.

[18] C. Li, J. Huang, and Q. Chang, “Data-Enabled Permanent Production Loss Analysis for Serial Production Systems With Variable Cycle Time Machines,” *IEEE Robot. Autom. Let.*, vol. 6, no. 4, pp. 6418–6425, Oct. 2021, doi: 10.1109/LRA.2021.3093012.

[19] J. Zou, Q. Chang, J. Arinez, G. Xiao, and Y. Lei, “Dynamic production system diagnosis and prognosis using model-based data-driven method,” *Expert Syst. Appl.*, vol. 80, pp. 200–209, Sep. 2017, doi: 10.1016/j.eswa.2017.03.025.

[20] J. Zou, Q. Chang, Y. Lei, and J. Arinez, “Event-Based Modeling and Analysis of Sensor Enabled Networked Manufacturing Systems,” *IEEE Trans. Autom. Sci. Eng.*, vol. 15, no. 4, pp. 1930–1945, Oct. 2018, doi: 10.1109/TASE.2018.2861837.