# Parameter Estimation for Two-machine Non-exponential Production Line Models using Parts Flow Data

Yuting Sun and Liang Zhang

2023 IEEE 19th International Conference on Automation Science and Engineering (CASE) | 979-8-3503-2069-5/23/$31.00 ©2023 IEEE | DOI: 10.1109/CASE56687.2023.10260426

***Abstract*— The foundation of any quantitative analysis and optimization for a production system is a high-fidelity mathematical model of the physical process. To construct a stochastic process model more efficiently, a new approach to production system modeling has recently been proposed that uses system performance metrics (e.g., throughput, work-in-process) derived from the parts flow data to reversely compute the model parameters. This paper extends this modeling approach to two-machine non-exponential production lines. Since no analytical expressions of performance metrics are available for non-exponential systems, we use neural network-based surrogate models as a means to calculate those performance metrics as functions in terms of the system parameters. Then, based on the constructed surrogate models and given performance metrics, the machine parameters are estimated by solving a constrained optimization problem using a multi-start particle swarm optimization algorithm. The results indicate that different combinations of machine parameters can lead to practically the same system performance metrics and a linear relationship of the reliability parameters from these obtained estimations is observed. Numerical experiments are implemented to justify the robustness of the different combinations of machine parameters.**

## I. INTRODUCTION

Production systems research studies the behavior and properties of job flow in manufacturing processes and investigates methods for its improvement, control, and optimization. To conduct such studies, a production system is typically modeled as a stochastic process, where the operations of the workstations (machines) are characterized by randomly distributed uptimes, downtimes, and cycle times [1]–[3]. Based on these theoretical models, researchers can develop algorithms and numerical tools for performance analysis, bottleneck identification, resource allocation, and control of production systems described by different types of stochastic models (for instance, [4]–[7]). On the other hand, the issues of how to construct such mathematical models of a manufacturing process are not trivial [3]. Indeed, without a valid mathematical model of the production system, none of the above-mentioned model-based analyses can be implemented.

In the conventional approach of creating a mathematical model of a production system, one usually collects the operating status data (i.e., up- and downtimes) from each individual workstation and designs customized procedures to calculate the parametric model for the overall system. Examples of this modeling process are illustrated in the case studies of [8]–[11]. The main challenges and limitations of this approach lie in the complexity, quality, and even unavailability of equipment status data, as discussed in [12]–[17]. To overcome these drawbacks, a new modeling approach for production systems is proposed in [12]–[15] and applied to an industrial case study in [16] and [17]. For the new approach, instead of collecting equipment operation status data, we measure the *parts flow*, i.e., the entrances/exits of parts to/from the buffer and the number of parts in the buffers in each time slot, which are all based on part counting. Then, we reversely compute the parameters of a production system model based on the system performance metrics derived from the parts flow data. In the existing studies of the new modeling approach, the machine parameter identification problem has been solved for Bernoulli (in [12]) and exponential (in [15]–[17]) production line models via an analytical expression-based method. Specifically, based on the close-formed expression of the performance metrics as functions of system parameters, the parameter identification problems are as formulated as optimization problems to minimize the sum of squared errors of the resulting performance metrics under the estimated parameters compared with the true ones.

It should be noted that both Bernoulli and exponential reliability models imply that the resulting production system models are characterized by Markov chains, which makes it possible to derive analytical expressions for calculating system performance metrics (throughput, work-in-process, etc.). Unfortunately, in practice, the machines often have up- and downtime distributed non-exponentially, and few results about the analytical calculation of performance metrics for production systems with non-exponential machines are available at this point. Among a limited number of studies, it is observed in [3] that the steady-state throughput of non-exponential serial lines is approximately a linear function of the average coefficients of variation (CV) of the machine up- and downtimes (given that all other system parameters remain fixed). Similar results are reported in [18] for assembly systems with non-exponential machines. However, other performance metrics and the joint effects of the mean and CV's of machine up- and downtimes are not discussed. Another study on non-exponential production lines is conducted in [19], which investigates the transient behavior of such production systems with machines having gamma reliability models. However, it only discusses the cases that all the machines have identical parameters.

The goal of this paper is to extend the parts flow data-based production system modeling approach to non-exponential serial line models and investigate the properties of the machine parameters and performance metrics. Due to

This work was supported in part by the U.S. National Science Foundation (NSF), under Grant Number FM-2134367.

Yuting Sun and Liang Zhang are with Department of Electrical and Computer Engineering, University of Connecticut, Storrs, CT, 06269, USA. Email: yuting.2.sun@uconn.edu, liang.zhang@uconn.edu

the complexity of non-exponential serial line models, only the case of two-machine lines is discussed in this paper. The study of multiple-machine non-exponential serial lines will be carried out in future work.

The rest of the paper is organized as follows: The system model and problem addressed are described in Section II. The method of model parameter estimation is developed in Section III. Section IV provides the numerical experiment results and illustrations. Finally, the conclusions and future work are summarized in Section V.

## II. SYSTEM MODEL AND PROBLEM DESCRIPTION

### A. System Model

In this paper, we consider the synchronous two-machine production line model as shown in Fig.1. This production line model operates according to the following assumptions:

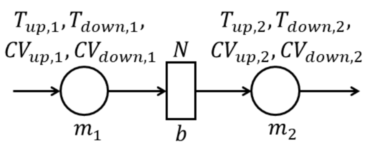


Fig. 1. Synchronous serial line with two non-exponential machines

1) Both machines have identical and constant cycle times. The model operates in continuous time and the time is measured in units of cycle time.
2) Machine $m_i$, $i = 1, 2$, are unreliable and subject to random failures, i.e., the uptime and downtime of machine $m_i$ are random variables following distributions $f_{up,i}$ and $f_{down,i}$, respectively. The means and CVs of the up- and downtime of $m_i$ are denoted as $T_{up,i}$, $CV_{up,i}$ and $T_{down,i}$, $CV_{down,i}$, respectively. The states (up and down) of the machines are independent. Using machine parameters $T_{up,i}$ and $T_{down,i}$, the efficiency of machine $m_i$, $e_i$, can be defined as
$$e_i = \frac{T_{up,i}}{T_{up,i} + T_{down,i}}, \quad i = 1, 2. \tag{1}$$
3) Buffer $b$ has capacity $N$, i.e., it can store up to $N$ parts.
4) Machine $m_2$ is starved if it is up, buffer $b$ is empty and machine $m_1$ is down. Machine $m_1$ is never starved.
5) Machine $m_1$ is blocked if it is up, buffer $b$ is full, and machine $m_2$ is down. Machine $m_2$ is never blocked.
6) If machine $m_i$ is up and neither starved nor blocked, it processes parts with deterministic cycle time.

For convenience, in this paper, we assume that the up- and downtimes of all machines follow the gamma distribution and use the continuous flow model [3].

### B. Problem Description

In this study, each machine in the model above is defined by 4 independent parameters: $T_{up,i}$, $T_{down,i}$, $CV_{up,i}$ and $CV_{down,i}$. Since $e_i$ can be obtained by $T_{up,i}$ and $T_{down,i}$ as (1), equivalently, a machine is defined by independent parameters $e_i$, $T_{down,i}$, $CV_{up,i}$ and $CV_{down,i}$. Either way, to identify the values of these parameters (a total of 8), at least 8 uncorrelated measurements are needed. As argued in [12]–[17], it may face various challenges to collect the direct measurements of operation up- and downtime in practice and an alternative approach of using system performance metrics derived from the parts flow data is recommended. Under system model assumptions 1)-6), we define the following five system performance metrics:

- *Production rate*, $PR$: the average number of parts produced by $m_2$ per cycle time during steady state.
- *Work-in-process*, $WIP$: the average number of parts contained in buffer $b$ during steady state.
- *Probability that buffer $b$ is empty*, $P_0$: the fraction of time that no part is in $b$ during steady state.
- *Probability that buffer $b$ is full*, $P_N$: the fraction of time that $N$ parts are in buffer $b$ during steady state.
- *Probability of unchanged buffer state*, $B_0$: the probability that the number of parts in buffer $b$ is not changed compared with the last cycle time.

In addition, we define four levels of buffer occupancy:

Buffer occupancy level 1: 0 (not including 0) to 25% of $N$;

Buffer occupancy level 2: 25% of $N$ to 50% of $N$;

Buffer occupancy level 3: 50% of $N$ to 75% of $N$;

Buffer occupancy level 4: 75% of $N$ to $N$ (not including $N$);

and then, we define

- *Probability of buffer occupancy level $k$*, $P_{Lk}$: the fraction of time that the occupancy of buffer $b$ is at level $k$, $k = 1, 2, 3, 4$, during steady state.

It follows immediately that $P_0 + P_{L1} + P_{L12} + P_{L3} + P_{L4} + P_N = 1$. This implies that these six probabilities only have the degree-of-freedom equal to five.

Note that these performance metrics can be all measured by monitoring the parts flow in the system. While it is possible to define and use other parts flow-based performance metrics, in this paper, we consider to use $PR^*$, $WIP^*$, $P_0^*$, $P_N^*$, $P_{L1}^*$, $P_{L2}^*$, $P_{L3}^*$ and $B_0^*$, where $\cdot^*$ represents the true system performance metrics at hand. Then, the problem addressed in this paper is:

**Problem 1:** *For a serial line defined by assumptions 1)-6) with buffer capacity $N$, estimate machine parameters $e_i$, $T_{down,i}$, $CV_{up,i}$, and $CV_{down,i}$, $i = 1, 2$ for given $PR^*$, $WIP^*$, $P_0^*$, $P_N^*$, $P_{L1}^*$, $P_{L2}^*$, $P_{L3}^*$, and $B_0^*$.*

## III. MODEL PARAMETER ESTIMATION

### A. Neural network surrogate models of performance metrics

To solve **Problem 1**, we first need to derive the quantitative relationships between the system parameters and performance metrics under the non-exponential reliability model assumption. In this paper, we propose to develop a surrogate model for calculating each performance metric as close-formed functions in terms of system parameters, $F_Y(e_i, T_{down,i}, CV_{up,i}, CV_{down,i}, N)$, where $Y$ represents the performance metrics defined in Section II-B. In this study, the surrogate models are constructed using neural networks (NNs). The construction of the NN for the surrogate model of each performance metric is shown in Fig. 2.

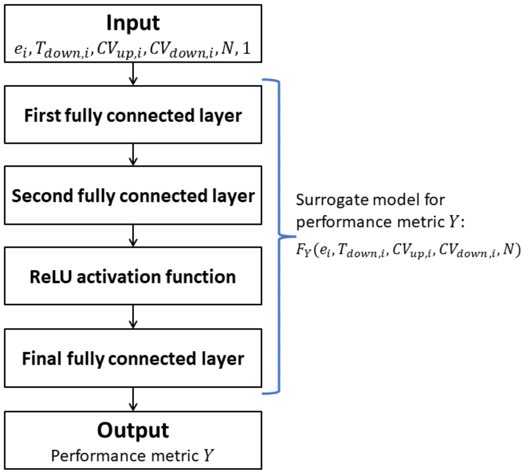


Fig. 2. Construction of the neural network for the surrogate model of each performance metric

The NN model for each performance metric consists of an input layer, two hidden layers, and an output layer following the activation function and the final fully connected layer. In the input layer, the number of nodes is set to the number of predictors plus one bias node. The number of neurons in the hidden layers depends on the complexity of the task and the amount of training data. In the hidden layers, each neuron is connected to all neurons in the preceding layer by an associated numerical weight. The weight connecting two neurons regulates the magnitude of the signal that is transmitted between them. Finally, all neurons pass onto the ReLU activation function and the final fully connected layer, and then, a single node exports from the output layer, which is the numerical response, the performance metric $Y$ under consideration (e.g., $PR$, $WIP$). To train the NN models, we use limited-memory Broyden-Fletcher-Goldfarb-Shanno (LBFGS) quasi-Newton algorithm [20] as optimizer. Since 8 different performance metrics are considered in this study, 8 independent NN models are developed. The numbers of neurons in hidden layers are selected as $(64, 32)$ for all NNs.

### B. Algorithm for parameter estimation

Based on the surrogate models of the performance metrics considered in **Problem 1**, we define the error function $\mathbf{F}$ as

$$\mathbf{F}(\mathbf{x}, N) = [f_1(\mathbf{x}, N), ..., f_8(\mathbf{x}, N)]^T$$
$$= \begin{bmatrix} F_{PR}(e_i, T_{down,i}, CV_{up,i}, CV_{down,i}, N) - PR^* \\ F_{WIP}(e_i, T_{down,i}, CV_{up,i}, CV_{down,i}, N)/N - WIP^*/N \\ F_{P_0}(e_i, T_{down,i}, CV_{up,i}, CV_{down,i}, N) - P_0^* \\ F_{P_N}(e_i, T_{down,i}, CV_{up,i}, CV_{down,i}, N) - P_N^* \\ F_{P_{L1}}(e_i, T_{down,i}, CV_{up,i}, CV_{down,i}, N) - P_{L1}^* \\ F_{P_{L2}}(e_i, T_{down,i}, CV_{up,i}, CV_{down,i}, N) - P_{L2}^* \\ F_{P_{L3}}(e_i, T_{down,i}, CV_{up,i}, CV_{down,i}, N) - P_{L3}^* \\ F_{B_0}(e_i, T_{down,i}, CV_{up,i}, CV_{down,i}, N) - B_0^* \end{bmatrix}, \tag{2}$$

where $\mathbf{x} = (e_i, T_{down,i}, CV_{up,i}, CV_{down,i})$, $i = 1, 2$. Then, **Problem 1** can be reformulated as the following constrained optimization problem:

**Problem 2:** *Find machine parameters* $\mathbf{x}$ *that minimizes the 2-norm of error function* $\mathbf{F}$ *over a certain box-constraint set* $\mathbf{X} = \{\mathbf{x} \in \mathbb{R}^8 | L_i < x_i < U_i, i = 1, ..., 8\}$, *i.e.,*

$$\begin{aligned} &\min_{\mathbf{x}} f(\mathbf{x}, N) = ||\mathbf{F}(\mathbf{x}, N)||^2, \\ &\text{s.t. } \mathbf{x} \in \mathbf{X}, \end{aligned} \tag{3}$$

*where* $L_i$*'s,* $U_i$*'s are the lower- and upper bounds for the machine parameters.*

To solve this constrained optimization problem, a particle swarm optimization (PSO) algorithm based on [21] and [22] is applied (see the pseudo-code of Algorithm 1 below).

**Algorithm 1** Multi-start Particle Swarm Optimization Algorithm (M-PSO)

**for** $n = 1, \dots, D$ **do**
  **Initialization**: Randomly create $K_p$ particles with the initial particle position $\mathbf{x}_{(k)}^{(0)} \in \mathbf{X}$ and initial velocity $\mathbf{v}_{(k)}^{(0)} \in \mathbf{V}$, where $\mathbf{V} = \{\mathbf{v} \in \mathbb{R}^8 | L_i - U_i \leq v_i \leq U_i - L_i, i = 1, \dots, 8\}$. Set the stall counter $c = 0$, and iteration step $j = 0$.
  **Evaluation**: Find the best position $\mathbf{d}^{(0)}$ among $\mathbf{x}_{(k)}^{(0)}$ i.e., $\mathbf{d}^{(0)} = \arg\min_{\mathbf{x}_{(k)}^{(0)}} f(\mathbf{x}, N)$, and let $p_{(k)} = \mathbf{d}^{(0)}$, $k = 1, \dots, K_p$.
  **while** $j < J_{max}$ **do**
    **for** $k = 1, ..., K_p$ **do**
      Randomly create neighborhood subset $S$ of $K_n$ particles other than $\mathbf{x}_{(k)}^{(j)}$ and find the best position $g_{(k)}$ among $S$;
      Update the velocity: $\mathbf{v}_{(k)}^{(j+1)} = W\mathbf{v}_{(k)}^{(j)} + y_1 u_1 \circ (p_{(k)} - \mathbf{x}_{(k)}^{(j)}) + y_2 u_2 \circ (g_{(k)} - \mathbf{x}_{(k)}^{(j)})$, where $u_1$ and $u_2$ are randomly picked in $(0, 1)$;
      Update the position: $\mathbf{x}_{(k)}^{(j+1)} = \mathbf{x}_{(k)}^{(j)} + \mathbf{v}_{(k)}^{(j+1)}$;
      **for** $i = 1, ..., 8$ **do**
        **if** $x_i \notin \mathbf{X}_i$ **then**
          $x_i = \arg\min_b |b - x_i|$, $b = \{U_i, L_i\}$;
          **if** $v_i \notin \mathbf{V}_i$ **then**
            $v_i = 0$;
          **end if**
        **end if**
      **end for**
      **if** $f(\mathbf{x}_{(k)}^{(j+1)}, N) < f(p_{(k)}, N)$ **then**
        $p_{(k)} = \mathbf{x}_{(k)}^{(j+1)}$;
      **end if**
    **end for**
    Update $\mathbf{d}^{(j+1)}$ and $f_{best}^{(j+1)} = f(\mathbf{d}^{(j+1)}, N)$;
    **if** $f_{best}^{(j)} - f_{best}^{(j+1)} > 0$ **then**
      $c = \max(0, c - 1)$;
      **if** $c < 2$ **then**
        $W = \min(2W, U_w)$;
      **end if**
      **if** $c > 5$ **then**
        $W = \max(W/2, L_w)$;
      **end if**
    **else**
      $c = c + 1$ and $K_n = \min(K_n + K_n^{(0)}, K_p)$;
    **end if**
    **if** $|f_{best}^{(j)} - f_{best}^{(j+1)}| \geq \epsilon_f$ **and** $T < T_{max}$ **then**
      $j = j + 1$;
    **else**
      Break;
    **end if**
  **end while**
  $\hat{\mathbf{x}}_n = \mathbf{d}^{(j)}$.
**end for**
**Return** $\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, ..., \hat{\mathbf{x}}_D$.

Note that, since multiple (non-unique) optimal solutions may exist, we start the algorithm with $D$ different sets of random initial points so that the distribution of optimized solutions can be further investigated. In addition, in Algorithm 1, we denote the number of particles as $K_p$, and the neighborhood size as $K_n$. The other hyperparameters include the initial inertia $W \in [L_w, U_w]$, self-adjustment weight $y_1 \in (0, +\infty)$ and social adjustment weight $y_2 \in (0, +\infty)$. Besides, we set the maximum number of iteration steps as $J_{max}$, and, if the running time (denoted as $T$) of solving a problem exceeds $T_{max}$, or the best value of the objective function cannot decrease $\epsilon_f$, then the algorithm is terminated.

## IV. Numerical Experiments and Illustration

For the numerical experiments, a simulation program of the serial production line with gamma reliability machines is created that runs with the first 10,000 cycle times as warm-up time and with the next 300,000 cycle times being the time period to statistically evaluate the performance metrics. Then, 20,000 different two-machine lines are randomly generated. The system parameters are randomly selected from the following ranges: $e_i \in [0.7, 0.95]$, $T_{down,i} \in [3, 20]$, $CV_{up,i}, CV_{down,i} \in [0.25, 1]$ and $N \in [\max\{T_{down,1}, T_{down,2}\},\ \ 3 \cdot \max\{T_{down,1}, T_{down,2}\}]$. For each line, 15 replications of the simulation program are executed and the average performance metrics from these runs are computed using the parts flow data collected from the buffer. All the computations are conducted in MATLAB on a Dell Inspiron 3671 workstation with Intel(R) Core(TM) i7-9700 CPU 3.00GHz processor and 16 GB of RAM.

### A. Accuracy of surrogate models

Among the 20,000 two-machine lines generated above, we randomly select 15,000 as the training dataset and the remaining 5,000 as the testing dataset. From the training data, we train the NN model for each performance metric, i.e., with one of the performance metrics being the response and the system parameters ($e_i$'s, $T_{down,i}$'s, $CV_{up,i}$'s, $CV_{down,i}$'s and $N$, $i = 1, 2$) being the predictors. Then, we can obtain the NN surrogate model-based expression of each performance metric as a function of the system parameters, $F_Y(\mathbf{x}, N)$, where $Y \in \{PR, WIP, P_0, P_N, P_{L1}, P_{L2}, P_{L3}, B_0\}$. With the NN surrogate models and given system parameters, we compute the performance metrics and evaluate the errors compared with the true ones using

$$\epsilon_{PR} = \frac{|\widehat{PR} - PR^*|}{PR^*} \cdot 100\%, \qquad \epsilon_{P_0} = |\widehat{P}_0 - P_0^*|,$$
$$\epsilon_{WIP} = \frac{|\widehat{WIP} - WIP^*|}{N} \cdot 100\%, \quad \epsilon_{P_N} = |\widehat{P}_N - P_N^*|,$$
$$\epsilon_{B_0} = \frac{|\widehat{B_0} - B_0^*|}{B_0^*} \cdot 100\%, \qquad \epsilon_{P_{Lk}} = |\widehat{P}_{Lk} - P_{Lk}^*|,\ k = 1, 2, 3. \tag{4}$$

where $\widehat{\cdot}$ denotes the estimated performance metrics.

Table I shows the estimation errors of performance metrics for two-machine gamma serial lines using our NN based-surrogate models. As one can see, given the system parameters, all performance metrics can be estimated with high accuracy using our surrogate models. This implies, our NN surrogate models are sufficiently accurate for performance metrics calculation in such systems.

TABLE I
Estimation errors of NN surrogate models

| | $\epsilon_{PR}$ (%) | $\epsilon_{WIP}$ (%) | $\epsilon_{P_0}$ | $\epsilon_{P_N}$ |
|---|---|---|---|---|
| Training data | 0.0837 | 0.1932 | 0.0017 | 0.0016 |
| Testing data | 0.0901 | 0.2753 | 0.0021 | 0.0020 |
| | $\epsilon_{B_0}$ (%) | $\epsilon_{P_{L1}}$ | $\epsilon_{P_{L2}}$ | $\epsilon_{P_{L3}}$ |
| Training data | 0.1095 | 0.0018 | 0.0017 | 0.0017 |
| Testing data | 0.1276 | 0.0023 | 0.0022 | 0.0021 |

### B. Accuracy of performance metrics resulting from estimated machine parameters

In this experiment, we randomly select 2,000 two-machine gamma lines from the testing dataset above. Given the true performance metrics as input, we search for the machine parameters using the M-PSO algorithm described in Section III-B and the NN surrogate models to calculate the performance metrics for each iterative solution found in the optimization process. For each line, we obtain $D = 200$ optimized solutions from M-PSO with $K_p = 100$ particles. For other hyperparameters of this algorithm, we set $K_n^{(0)} = 25$, $W = U_w = 1.1$, $L_w = 0.1$, $y_1 = y_2 = 1.5$, $\epsilon_f = 10^{-7}$, $J_{max} = 10000$, and $T_{max} = 300s$.

Note that, while we obtain 200 different solutions of estimated machine parameters from M-PSO for each line, not all of them are necessarily *valid*. Here, we define valid solutions $\hat{x}$ as those satisfying $f(\hat{x}, N) < 10^{-4}$ and $\hat{x} \in \mathbf{X}$. On average, for each line we studied, about 171 out of 200 solutions found by M-PSO are valid. Table II shows the average errors of performance metrics under the *valid* M-PSO-estimated machine parameters. The errors are evaluated based on (4) using both the NN surrogate model and simulation as the baseline. As one can see, the machine parameters obtained can indeed provide an almost perfect match to the observed performance metrics under true system parameters.

TABLE II
Average of performance metrics estimation errors

| | NN-based Err. | Simulation-based Err. |
|---|---|---|
| $PR$ | 0.0489% | 0.1287% |
| $WIP$ | 0.0497% | 0.3081% |
| $P_0$ | 0.0005 | 0.0031 |
| $P_N$ | 0.0004 | 0.0030 |
| $P_{Lk}$ ($k = 1, 2, 3$) | 0.0005 | 0.0027 |
| $B_0$ | 0.0582% | 0.1951% |
| $f(\hat{x}, N)$ | $5.88 \times 10^{-6}$ | $8.64 \times 10^{-5}$ |

### C. Distribution of estimated machine parameters

Although the valid solutions of estimated machine parameters found by M-PSO can fit the system performance metrics with very high accuracy, the underlying machine parameters may be (quite) different from the true parameters that we intend to identify. Indeed, for all valid solutions

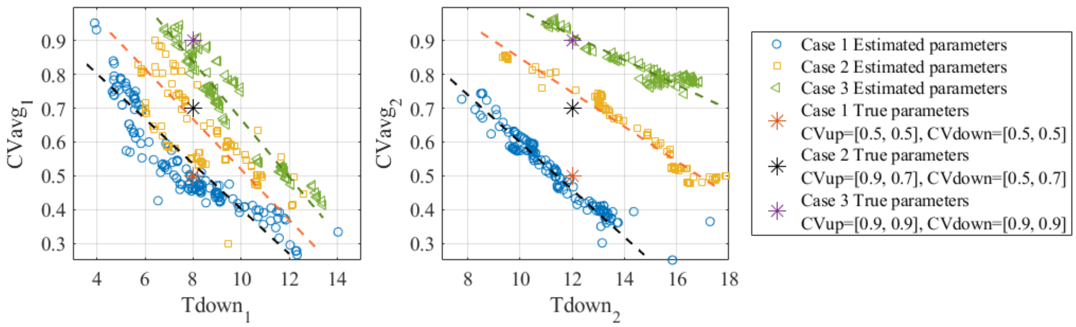


Fig. 3. Illustration of linear relationship between $T_{down,i}$ and $CV_{avg,i}$

of estimated machine parameters, the estimated machine efficiencies are typically very close to the true ones with the average estimation errors of $\hat{e}_1$ and $\hat{e}_2$ being 0.2850% and 0.2874%, respectively. However, M-PSO estimated parameters $\hat{T}_{down,i}$, $\widehat{CV}_{up,i}$, and $\widehat{CV}_{down,i}$ may distribute all over their feasible ranges defined in formulation (3). This phenomenon is illustrated in Fig. 3 for three cases with $N = 20$, $\mathbf{e} = [0.8, 0.9]$, $\mathbf{T}_{down} = [8, 12]$ and the $CV$'s are set as different values. In this figure, $CV_{avg,i} = \frac{1}{2}(CV_{up,i} + CV_{down,i})$, $i = 1, 2$, and the true system parameters are marked by the asterisks. It is interesting to observe that the valid estimated machine parameters (from the perspective of matching system performance metrics) exhibit a negative linear pattern between $CV_{avg}$ and $T_{down}$. To evaluate the linear relationships, for each case, we create the linear regression models $F_i(T_{down,i}) : CV_{avg,i} = b_{0,i} + b_{1,i}T_{down,i}$ and then, compute the $p$-value of coefficient $b_{1,i}$ and correlation coefficient $R^2$ of these models. For the 2,000 cases in this experiment, we find significant linear relationships ($p$-value$<$ 0.05) between $T_{down}$ and $CV_{avg}$ in 92.8% cases (1,856 out of 2,000), and on average, $R^2 = 0.76$. This observation leads to the following

**Numerical Fact 1.** *For a two-machine gamma production line, given $e_i$'s, $N$, and the fitted functions $F_i(T_{down,i})$ : $CV_{avg,i} = b_{0,i} + b_{1,i}T_{down,i}$, if we randomly select a set of parameters of $m_1$ ($T_{down,1}$, $CV_{up,1}$, $CV_{down,1}$) that satisfies function $F_1$, then we can find at least a set of parameters of $m_2$ ($T_{down,2}$, $CV_{up,2}$, $CV_{down,2}$) satisfying $F_2$ that lead to practically the same performance metrics (with errors $f(\hat{x}, N) < 10^{-4}$). The same result also holds for $m_2$.*

*Justification:* To justify this numerical fact, 200 two-machine gamma lines out of the 2,000 lines are randomly selected. The valid estimated machine parameters have been identified in Subsection IV-B. Since the machine efficiencies are already estimated with high accuracy, for each case, we directly use the average values of $\hat{e}_1$ and $\hat{e}_2$ of the valid solutions as the estimated machine efficiency. Next, we randomly select $T_{down,1}$ for $m_1$ and calculate the corresponding $CV_{avg,1}$ based on $F_1(T_{down,1})$. $CV_{up,1}$ and $CV_{down,1}$ are randomly chosen to match the average value $CV_{avg,1}$. During this process, all parameters must be ensured in their respective feasible ranges. With the selected parameters of $m_1$, we search for the best combination of $m_2$ parameters along linear function $F_2(T_{down,2})$ that minimizes the performance metrics estimation error function $f(\hat{\mathbf{x}}, N)$ defined in (3). The same process is then performed by randomly selecting the parameters of $m_2$ on linear function $F_2(T_{down,2})$ and searching for the best combination of $m_1$ parameters along linear function $F_1(T_{down,1})$. For each case, this experiment is conducted 10 times to obtain a total of 20 combinations of machines parameters (10 from fixing $m_1$ parameters and searching for $m_2$'s and 10 from fixing $m_2$ parameters and searching for $m_1$'s), all of which satisfy linear functions $F_1(T_{down,1})$ and $F_2(T_{down,2})$. As a result, all $20 \times 200 = 4,000$ machine parameter combinations obtained above satisfy $f(\hat{\mathbf{x}}, N) < 10^{-4}$. In other words, for each line, the system performance metrics under all 20 combinations of machine parameters studied are practically indistinguishable. □

To further understand the linear relationship of $T_{down}$ vs. $CV_{avg}$ and illustrate Numerical Fact 1, we take Case 1 and Case 3 used in Fig. 3 as examples to implement the numerical experiment described in the justification of Numerical Fact 1 and randomly pick 4 combinations (out of 20) into Table III. Under these estimated machine parameters, we compute

TABLE III
SELECTED ESTIMATIONS OF CASES 1 AND 3

| | | $\hat{\mathbf{T}}_{down}$ | $\widehat{\mathbf{CV}}_{up}$ | $\widehat{\mathbf{CV}}_{down}$ |
|---|---|---|---|---|
| Case 1 | Est. 1 | [5.00, 12.53] | [0.60, 0.42] | [0.80, 0.53] |
| | Est. 2 | [9.00, 12.91] | [0.45, 0.38] | [0.45, 0.52] |
| | Est. 3 | [6.61, 10.00] | [0.85, 0.75] | [0.25, 0.40] |
| | Est. 4 | [5.21, 13.00] | [0.76, 0.30] | [0.72, 0.50] |
| Case 3 | Est. 1 | [7.00, 13.74] | [0.95, 0.72] | [0.88, 0.97] |
| | Est. 2 | [10.00, 11.10] | [0.70, 1.00] | [0.85, 0.92] |
| | Est. 3 | [7.14, 12.00] | [1.00, 0.85] | [0.95, 0.95] |
| | Est. 4 | [7.83, 15.00] | [0.81, 0.60] | [0.91, 0.95] |

the system performance metrics using simulation and the resulting $PR$ and $WIP$ are plotted in Fig. 4. Although the estimated machine parameters are quite different from the true ones of the system, the performance metrics under these parameters are all very close to the ones under the true parameters, for not only the original buffer capacity but also varied ones. This observation further verifies Numerical Fact 1 and implies that such property could be useful in designing machine operating regimes in production systems analysis and control.

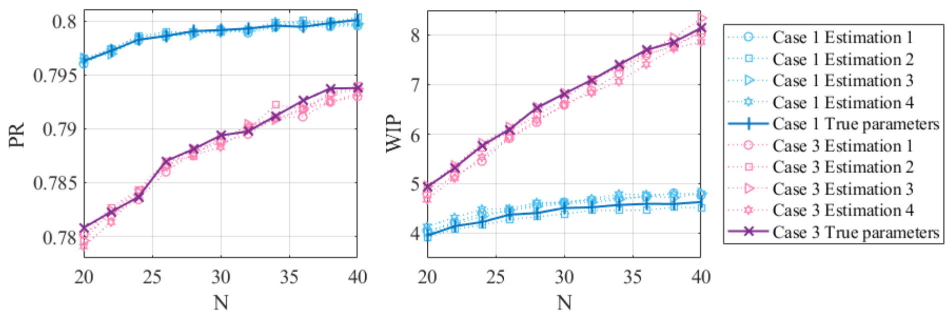


Fig. 4. Estimated $PR$ and $WIP$ resulting from different parameter estimations under $N$ expansion

## V. Conclusion

In this paper, we apply the parts flow performance metrics-based approach to estimate the parameters of synchronous two-machine non-exponential production line models. The system performance metrics used include production rate, work-in-process, the probability distribution of buffer occupancy, etc, all of which can be derived from the parts flow data collected from the buffer. To accomplish this, we first build neural network-based surrogate models as a means to perform the analytical expressions-based calculation of system performance metrics (instead of using time-consuming simulations). Then, we formulate a constrained optimization problem to find the optimized combinations of machine parameters that can fit the given system performance metrics. To solve this optimization problem, a multi-start particle swarm optimization (M-PSO) algorithm is applied. Numerical experiments show that although the estimated machine parameters obtained using the M-PSO algorithm may be quite different from the true ones, they can still lead to almost exactly the same system performance metrics compared to those observed under true machine parameters. Moreover, the negative linear patterns of machine parameters leading to the same system performance metrics are further analyzed and illustrated. In future work, we will extend this study to multiple-machine production line models and systems with asynchronous machines.

## References


[1] S. B. Gershwin, *Manufacturing Systems Engineering*. Prentice Hall, Englewood Cliff, NJ, 1994.

[2] D. D. Yao, *Stochastic Modeling and Analysis of Manufacturing Systems*. Springer-Verlag, New York, NY, 1994.

[3] J. Li and S. M. Meerkov, *Production Systems Engineering*. Springer, 2009.

[4] L. Zhang, C. Wang, J. Arinez, and S. Biller, "Transient analysis of Bernoulli serial lines: Performance evaluation and system-theoretic properties," *IIE Transactions*, vol. 45, no. 5, pp. 528–543, 2013.

[5] Z. Jia, L. Zhang, J. Arinez, and G. Xiao, "Performance analysis for serial production lines with Bernoulli machines and real-time WIP-based machine switch-on/off control," *International Journal of Production Research*, vol. 54, no. 21, pp. 6285–6301, 2016.

[6] ——, "Finite production run-based serial lines with Bernoulli machines: Performance analysis, bottleneck, and case study," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 1, pp. 134–148, 2015.

[7] G. Chen, L. Zhang, J. Arinez, and S. Biller, "Energy-efficient production systems through schedule-based operations," *IEEE Transactions on Automation Science and Engineering*, vol. 10, no. 1, pp. 27–37, 2012.

[8] J. Arinez, S. Biller, S. M. Meerkov, and L. Zhang, "Quality/quantity improvement in an automotive paint shop: A case study," *IEEE Transactions on automation science and engineering*, vol. 7, no. 4, pp. 755–761, 2009.

[9] M. Zandieh, M. Joreir-Ahmadi, and A. Fadaei-Rafsanjani, "Buffer allocation problem and preventive maintenance planning in non-homogenous unreliable production lines," *The International Journal of Advanced Manufacturing Technology*, vol. 91, no. 5-8, pp. 2581–2593, 2017.

[10] J.-H. Lee, C. Zhao, J. Li, and C. T. Papadopoulos, "Analysis, design, and control of Bernoulli production lines with waiting time constraints," *Journal of Manufacturing Systems*, vol. 46, pp. 208–220, 2018.

[11] K. Park and J. Li, "Improving productivity of a multi-product machining line at a motorcycle manufacturing plant," *International Journal of Production Research*, vol. 57, no. 2, pp. 470–487, 2019.

[12] Y. Sun, T. Zhu, L. Zhang, and P. Denno, "Parameter identification for bernoulli serial production line model," *IEEE Transactions on Automation Science and Engineering*, 2020.

[13] Y. Sun and L. Zhang, "Parameter identification for multiple-machine Bernoulli lines using statistical learning methods," in *2020 IEEE 16th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2020.

[14] J. Tu, T. Zhu, Y. Bai, and L. Zhang, "Estimation of machine parameters in exponential serial lines using feedforward neural networks," in *2020 IEEE 16th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2020.

[15] Y. Sun and L. Zhang, "Parameter identification for synchronous two-machine exponential production line model," in *2021 IEEE 17th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2021, pp. 1884–1889.

[16] ——, "A novel approach to modeling of production system: A case study at a small/medium-sized manufacturer," in *2022 IEEE 18th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2022, pp. 1848–1853.

[17] ——, "Application of a novel approach of production system modelling, analysis and improvement for small and medium-sized manufacturers: a case study," *International Journal of Production Research*, pp. 1–21, 2022.

[18] S. Ching, S. M. Meerkov, and L. Zhang, "Assembly systems with non-exponential machines: Throughput and bottlenecks," *Nonlinear Analysis: Theory, Methods & Applications*, vol. 69, no. 3, pp. 911–917, 2008.

[19] M. Yang, L. Zhang, and P. Denno, "Transients of synchronous serial production lines with non-exponential reliability machines," in *2018 IEEE 14th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2018, pp. 1507–1512.

[20] J. Nocedal and S. Wright, *Numerical optimization*. Springer Science & Business Media, 2006.

[21] M. E. H. Pedersen, "Good parameters for particle swarm optimization," *Hvass Lab., Copenhagen, Denmark, Tech. Rep. HL1001*, pp. 1551–3203, 2010.

[22] E. Mezura-Montes and C. A. C. Coello, "Constraint-handling in nature-inspired numerical optimization: past, present and future," *Swarm and Evolutionary Computation*, vol. 1, no. 4, pp. 173–194, 2011.