Optimal capacity design for unreliable assembly-like supply chain systems

Fong-Fan Wang
Department of Industrial Engineering & Management
Hsiuping University of Science and Technology, Taichung, Taiwan
Tel: (+886) 4-2496-1123, Email: frank@hust.edu.tw

Abstract. This study investigates supply chain systems with an assembly feature. First we consider an unreliable make-to-stock (MTS) system, which is formulated as a multi-server unreliable assembly system with a machine interference characteristic. Second, we consider a hybrid make-to-stock and make-to-order (MTS-MTO) system where the supply is a kitting process and the final assembly process is unreliable and is initiated according to order arrival. For both systems steady state performance measures were derived by using algorithmic procedures to solve the underlying quasi-birth-and-death (QBD) processes. Numerical examples are provided to study the performance of unreliable assembly systems when system parameters are changed. Then we use different search algorithms, including Genetic Algorithm, Simulated Annealing and Quasi-Newton to search for optimal operating rate and repairing rate under allowable capacity range. The objective is to solve the trade-off of costs related to inventory holding, order waiting, operating, and repairing. Interestingly, all three search methods obtain very close results.

Keywords: assembly-like, machine interference, MTS-MTO, QBD, capacity design

1. INTRODUCTION

Assembly process is a common practice in production systems. In the literature assembly process is also referred to as "kitting process". Stochastic assembly process relates to the problem of uncertainty inherent in component (or part) supply and assembly processes. Characteristics of stochastic assembly models (SAM) assume that there are at least two independent part supply processes with zero or non-negligible assembly time. In the past, studies in SAM focus on theoretical derivation of versatile models with different operating and distribution assumptions.

Harrison (1973) showed that a sufficient and necessary condition for the queue of assembly production system to be stable is for the component buffers to be finite. Ramachandran and Delen (2005) relaxed the assumption of finite buffer capacity constraint on the input buffers, and showed that the system remains stable under fairly mild conditions. Altiok (1997) showed the equivalence of an assembly production system and a tandem production system under block before process mode. Bonomi (1987) proposed an approximate analysis for a class of assembly-like queues with N inputs using a decomposition approach. Latouche (1981) proposed matrix geometric method (MGM) for solving queues with paired customers. De Cuypere et al. (2013) assessed the impact of kitting interruptions, bursty part arrivals and phase-type distributed kit assembly times on the behaviour of the part buffers. Som et al. (1994) and Wilhelm and Som (1998, 1999) analyzed output stream of kits and kitting process with general distribution.

The above literature studies reliable assembly systems only. Gao et. al (2010) proposed unreliable analytical model for assemble to order system with zero kitting time. The authors analyzed the performance behavior when system parameters pertaining to arrival rate and failure rate were changed. Liu and Yuan (2001) and Yuan and Liu (2005) evaluated SAM through derivation of underlying quasi-birth-and-death (QBD) models for respective unreliable assembly systems with different kitting scenarios. On the other hand, for maintenance float design problems Lin and Chien (1995) used M/M/c/k queueing model to investigate optimal maintenance capacity in terms of server and repairmen level. Zeng and Zhang (1997) used the same queueing model to investigated similar maintenance float problem with additional spare provisioning initiative to obtain optimal design of server, spares and repair capacity. The motivation of this study is that mathematical model to combine maintenance float design with SAM is not yet available, though it is not difficult to derive. The objective of this study is to blend these two isolated research streams into one analytical framework. The purpose of such formulation is to enhance the availability of the system such that the performance of unreliable SAM may be realized and hopefully such systems may be improved through proper capacity design. Furthermore, pure make-to-stock (MTS) type production mode is fading during the past decades. Customized production according to specific customer order is gaining its popularity. To cope with this trend, we modify the traditional MTS-based SAM with make-to-order to formulate a hybrid MTS-MTO production with unreliable assembly. With regards to hybrid MTS-MTO production model without assembly feature we refer to De Cuypere et al. (2012) and Jewkes and Alfa (2009).

2. FAILURE-PRONE ASSEMBLY SYSTEMS

In the following we formulate an MTS system with unreliable assembly followed by a hybrid MTS-MTO system with unreliably assembly.

Notation

m	Number of machine
c	Number of repairmen
s	Number of spare machine
S	Maximum semi-finished product buffer level
J_A	Maximum part level for part supply A
J_B	Maximum part level for part supply B
λ_A	Arrival rate for part supply A
λ_B	Arrival rate for part supply B
θ	Failure rate for production machine
α	Failure rate for spare machine
σ	Repair rate for machine
λ	External customer arrival rate
u	Assembly rate for a kit
$\mathbf{I}$	Identity matrix with appropriate size
$\mathbf{e}$	Column vector of all ones with appropriate size
A	System availability
L	Average order waiting level
ϕ	System throughput
$\overline{B}_1$	Average part level in part buffer A
$\overline{B}_2$	Average part level in part buffer B
W	Average semi-finished product level

2.1 Pure MTS System with unreliable assembly

To represent the state space of the system we use the index scheme: $(\mathrm{i},\mathrm{j},\mathrm{k})$ as follows. i denotes system occupancy level. j denotes pairing (or kitting) level, which is indexed as $\mathrm{J_B}-\mathrm{J_A}$ throughout this paper, k denotes available (operating) machine.

2.1.1 Problem Description

The scenario is like the one shown in figure 1. Two independent Poisson arrival streams with arrival rates λ_A and λ_B are assumed for arrival processes of part A and part B respectively. Part buffers for part A and B are finite with respective capacity $\mathrm{J_A}$ and $\mathrm{J_B}$. When either of the part buffer is full corresponding supplier is blocked from part supply. The blocking is removed until the part level is less than respective maximal buffer level. When at least one part in respective buffer is available, a kit is formed with zero time and put into semi-finished product buffer for final assembly. The semi-finished buffer is assumed infinite. Assume supply process is reliable while the assembly process is unreliable. There are m identical machines with failure rate θ. c repairmen are available for repairing failed machines. Assume $\mathrm{m} \geq \mathrm{c}$ thus machine interference may happen when all repairmen are busy and machine failure happens. The system occupancy level here refers to semi-finished kitting products which need final assembly. Assume all random variables including production, failure, and repair times are mutually independent. Hereafter this is denoted as model 1.

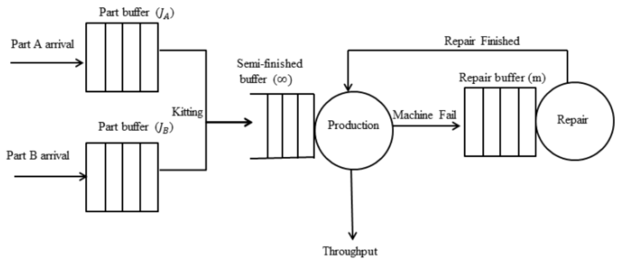

Figure 1: MTS with unreliable assembly

2.1.2 QBD Models

QBD Model and steady-state analysis for model 1 is as follows.

$$\mathbf{Q}=\begin{bmatrix} \mathbf{A}_{01} & \mathbf{A}_{00} & & & & \\ \mathbf{A}_{12} & \mathbf{A}_{11} & \mathbf{A}_{10} & & & \\ & \ddots & \ddots & \ddots & & \\ & & \mathbf{A}_{m-1,2} & \mathbf{A}_{m-1,1} & \mathbf{A}_{m-1,0} & \\ & & & \mathbf{A}_2 & \mathbf{A}_1 & \mathbf{A}_0 \\ & & & & \ddots & \ddots & \ddots \end{bmatrix} \quad (1)$$

$$\mathbf{A}'_{00}=\begin{array}{c} J_B \\ J_B-1 \\ \vdots \\ 0 \\ \vdots \\ J_A-1 \\ J_A \end{array}\begin{bmatrix} 0 & \lambda_A & & & & & \\ & 0 & \ddots & & & & \\ & & \ddots & \lambda_A & & & \\ & & & 0 & & & \\ & & & \lambda_B & \ddots & & \\ & & & & \ddots & 0 & \\ & & & & & \lambda_B & 0 \end{bmatrix}$$

$$\mathbf{A}_{00}=\mathbf{A}'_{00}\otimes\mathbf{I}_{m+1} \quad (2)$$

$$
\begin{array}{r}
J_B \\ J_B-1 \\ \vdots \\ \mathbf{A}'_{01} = \quad 0 \\ \vdots \\ J_A-1 \\ J_A
\end{array}
\begin{bmatrix}
-\lambda_A & & & & & & \\
\lambda_B & * & & & & & \\
 & \ddots & \ddots & & & & \\
 & & \lambda_B & 0 & \lambda_A & & \\
 & & & & \ddots & \ddots & \\
 & & & & & * & \lambda_A \\
 & & & & & & -\lambda_B
\end{bmatrix}
$$

where

$$* = -(\lambda_A + \lambda_B)$$

Define

$$
\begin{array}{r}
0 \\ 1 \\ \mathbf{Q}' = \quad \vdots \\ m-1 \\ m
\end{array}
\begin{bmatrix}
b_0 & c_0 & & & \\
a_1 & b_1 & c_1 & & \\
 & & \ddots & & \\
 & & a_{m-1} & b_{m-1} & c_{m-1} \\
 & & & a_m & b_m
\end{bmatrix}
$$

where for $m \geq k \geq 0$

$$a_k = k\theta, c_k = \sigma \cdot min(m-k, c), b_k = -(a_k + c_k).$$

Then

$$\mathbf{A}_{01} = \mathbf{A}'_{01} \oplus \mathbf{Q}'. \tag{3}$$

For $m \geq j \geq 1$

$$\mathbf{A}'_{j2} = diag\{u \cdot min(i,j), 0 \leq i \leq m\}$$

$$\mathbf{A}_{j2} = \mathbf{I} \otimes \mathbf{A}'_{j2} \tag{4}$$

$$\mathbf{A}_{j0} = \mathbf{A}_{00} \tag{5}$$

$$\mathbf{A}_{j1} = \mathbf{A}_{01} - \mathbf{A}_{j2} \tag{6}$$

Finally

$$\mathbf{A}_0 = \mathbf{A}_{m0}, \mathbf{A}_1 = \mathbf{A}_{m1}, \mathbf{A}_2 = \mathbf{A}_{m2} \tag{7}$$

2.1.3 MGM Solution

Herein we employ the technique illustrated in Neuts & Lucanton (1979) and Neuts (1994) for solving the QBD problem with boundary transitions. The MGM solution composes of two parts: computation of root matrix $\mathbf{R}$ and computation of stationary probability vector. For functions (1) to (7), the boundary transition rate is not homogeneous. However, the transition remain the same for occupancy level exceeding $m-1$. Therefore the following characteristic equation must hold

$$\mathbf{A}_0 + \mathbf{R}\mathbf{A}_1 + \mathbf{R}^2\mathbf{A}_2 = \mathbf{0}. \tag{8}$$

Starting from $\mathbf{0}$, $\mathbf{R}$ can be found from the following successive substitution until convergence criteria is met

$$\mathbf{R}_{k+1} = -(\mathbf{A}_0 + \mathbf{R}_k^2\mathbf{A}_2)\mathbf{A}_1^{-1}. \tag{9}$$

The convergence criteria can be either the maximum preset iteration or the difference between the previous solution and the current one is within some acceptable level, which one comes first. The MGM solution now becomes

$$\mathbf{x}_i = \mathbf{x}_{m-1}\mathbf{R}^{i+1-m}, i > m-1. \tag{10}$$

where $\mathbf{x}$ is the stationary probability vector of (1). Now $\mathbf{x}_{m-1}$ can be found by solving the boundary condition

$$
\begin{gathered}
\boldsymbol{x_0 A_{01}} + \boldsymbol{x_1 A_{12}} = \mathbf{0}, \\
\boldsymbol{x_{i-1} A_{i,0}} + \boldsymbol{x_i A_{i,1}} + \boldsymbol{x_{i+1} A_{i,2}} = \mathbf{0}, m-2 \geq i \geq 1, \\
\boldsymbol{x_{m-1} A_{m-1,0}} + \boldsymbol{x_{m-1} A_{m-1,1}} + \boldsymbol{x_{m-1} R A_2} = \mathbf{0}.
\end{gathered} \tag{11}
$$

Applying the iterative method suggested by Neuts (1994), one can solve the probability vector of the boundary condition as follows. Introducing $\Delta_0, \Delta_i, m-1 \geq i \geq 1$ as $-\mathrm{diag}(\mathbf{A}_{01})$ and $-\mathrm{diag}(\mathbf{A}_{i,1}), m-1 \geq i \geq 1$ respectively. Now (11) can be solved recursively by starting from $\boldsymbol{x}_i^{(0)} = \boldsymbol{\eta}$ (where $\boldsymbol{\eta}$ is a very small number) until the maximum preset iteration or the difference between the previous solution and the current one is within some acceptable level, which one comes first as shown below.

$$
\begin{gathered}
\mathbf{x}_0^{(k+1)} = \left(\mathbf{x}_0^{(k)}(\mathbf{A}_{01} + \Delta_0) + \mathbf{x}_1^{(k)}\mathbf{A}_{12}\right)\Delta_0^{-1}, \\
\mathbf{x}_i^{(k+1)} = \left(\mathbf{x}_{i-1}^{(k)}\mathbf{A}_{i,0} + \mathbf{x}_i^{(k)}(\mathbf{A}_{i,1} + \Delta_i) + \mathbf{x}_{i+1}^{(k)}\mathbf{A}_{i,2}\right)\Delta_i^{-1}, \\
m-2 \geq i \geq 1, \\
\mathbf{x}_{m-1}^{(k+1)} = \left(\mathbf{x}_{m-2}^{(k)}\mathbf{A}_{m-2,0} + \mathbf{x}_{m-1}^{(k)}\left(\mathbf{A}_{m-1,1} + \Delta_{m-1} + \mathbf{R}\mathbf{A}_2\right)\right)\Delta_{m-1}^{-1}.
\end{gathered} \tag{12}
$$

Together with the normalization equation

$$\sum_{i=0}^{m-2}\mathbf{x}_i\mathbf{e} + \mathbf{x}_{m-1}(\mathbf{I} - \mathbf{R})^{-1}\mathbf{e} = 1,$$

the stationary probability vector of boundary levels can be found. From (10) all the other probabilities can be obtained.

2.1.4 Stability condition

Under the stability condition of the studied QBD model $\boldsymbol{\pi}\mathbf{A} = \mathbf{0}, \boldsymbol{\pi}\mathbf{e} = 1, \boldsymbol{\pi}$ is stationary probability vector of $\mathbf{A}, \mathbf{A} = \mathbf{A}_0 + \mathbf{A}_1 + \mathbf{A}_2$.

According to Neuts (1994), the condition of a stable QBD model is

$$\boldsymbol{\pi}\mathbf{A}_0\mathbf{e} < \boldsymbol{\pi}\mathbf{A}_2\mathbf{e}. \tag{13}$$

It is difficult to derive closed-form solution for such condition. Numerical method may be applied to see if the model understudy is stable.

2.1.5 Performance measures

Let

$$\mathbf{p} = \boldsymbol{x_m}(\mathbf{I} - \mathbf{R})^{-1}$$

$$\mathbf{q} = \sum_{i=0}^{m-2}\boldsymbol{x_i} + \boldsymbol{x_{m-1}}(\mathbf{I} - \mathbf{R})^{-1}$$

Then system availability is

$$A = \frac{\sum_{i=0}^{S} \sum_{j=-J_A}^{J_B} \sum_{k=1}^{m} k q_{ijk}}{m} \tag{14}$$

Applying the expectation operation we obtain expected level for semi-finished product.

$$\mathrm{W} = \sum_{i=1}^{m-2} i\, \boldsymbol{x}_i \mathbf{e} + \boldsymbol{x}_{m-1}\left((m-1)(\mathbf{I}-\mathbf{R})^{-1} + \mathbf{R}(\mathbf{I}-\mathbf{R})^{-2}\right)\mathbf{e} \tag{15}$$

Throughput and average part buffer are

$$\phi = \sum_{i=1}^{m-1} \sum_{j=-J_A}^{J_B} \sum_{k=1}^{m} x_{ijk}\, min(i,k)\, u + \sum_{j=-J_A}^{J_B} \sum_{k=1}^{m} p_{jk}\, ku \tag{16}$$

$$\overline{B}_1 = \sum_{j=-1}^{-J_A} \sum_{k=0}^{m} (-j) q_{jk} \tag{17}$$

$$\overline{B}_2 = \sum_{j=J_B}^{1} \sum_{k=0}^{m} j q_{jk} \tag{18}$$

2.2 Hybrid MTS-MTO with unreliably assembly

We use the same index scheme as in 2.1 but with different meaning. Here occupancy level refers to external arrival order, which is denoted as l. For each occupancy level (l), i denotes semi-finished product level; j is the same as 2.1; k denotes number of machine waiting for repair.

2.2.1 Problem Description

The scenario is like the one shown in figure 2. The part supply process and basic assumptions are the same as model 1. However, the semi-finished buffer is now assumed finite with fixed capacity S. Assume when semi-finished product buffer is full supplier is blocked from supply material. The blocking is removed when the semi-finished product level is less than maximal semi-finished product level. Assume external customer order arrives according to a Poisson process with rate λ. Assembly production starts only when order buffer is not empty, machine is available, and semi-finished buffer is not empty. To enhance the system availability spare inventory with capacity s is installed. Assume failure rates for spares are α, which is smaller than θ to represent the warm standby case in reliability literature. Hereafter this is denoted as model 2.

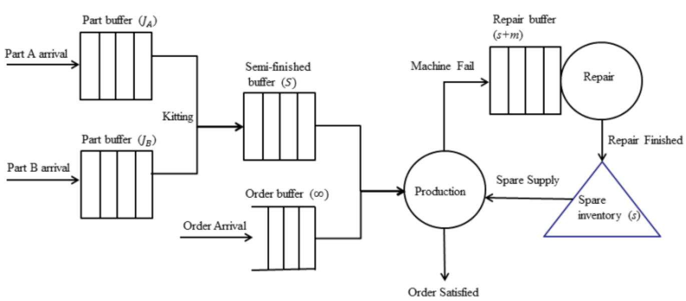

Figure 2: Hybrid MTS-MTO with unreliable assembly

2.2.2 QBD MODELS

The QBD Model for model 2 is

$$\widetilde{\mathrm{Q}} = \begin{bmatrix} \widetilde{\mathbf{A}}_{01} & \widetilde{\mathbf{A}}_0 & & & & \\ \widetilde{\mathbf{A}}_{12} & \widetilde{\mathbf{A}}_{11} & \widetilde{\mathbf{A}}_0 & & & \\ & \ddots & \ddots & \ddots & & \\ & & \widetilde{\mathbf{A}}_{m-1,2} & \widetilde{\mathbf{A}}_{m-1,1} & \widetilde{\mathbf{A}}_0 & \\ & & & \widetilde{\mathbf{A}}_2 & \widetilde{\mathbf{A}}_1 & \widetilde{\mathbf{A}}_0 \ddots \\ & & & & \ddots & \ddots \end{bmatrix} \tag{19}$$

The failure rate when k servers are under repair is

$$\lambda_k = \begin{cases} m\theta + (s-k)\alpha, & 0 \le k \le s, \\ (s+m-k)\theta, & s+1 \le k \le s+m. \end{cases}$$

The repair rate when k servers are under repair is

$$\sigma_k = \begin{cases} k\sigma, & 1 \le k < c, \\ c\sigma, & c \le k. \end{cases}$$

Define

$\widetilde{\mathbf{Q}}'$

$$= \begin{array}{c} 0 \\ 1 \\ \vdots \\ s \\ s+1 \\ \vdots \\ s+m-1 \\ s+m \end{array} \begin{bmatrix} * & \lambda_0 & & & & & & \\ \sigma_1 & * & \lambda_1 & & & & & \\ & \ddots & * & \ddots & & & & \\ & & \sigma_s & * & \lambda_s & & & \\ & & & \sigma_{s+1} & * & \lambda_{s+1} & & \\ & & & & \ddots & * & \ddots & \\ & & & & & \sigma_{s+m-1} & * & \lambda_{s+m-1} \\ & & & & & & \sigma_{s+m} & * \end{bmatrix} \tag{20}$$

where

$$* = -(\lambda_k + \sigma_k), 0 \le k \le s+m$$

Then

$$\widetilde{\mathbf{A}}_0 = \lambda\mathbf{I}$$

$$\widetilde{\mathbf{A}}_{01} = \begin{array}{c} \mathbf{0} \\ \mathbf{1} \\ \vdots \\ \boldsymbol{S-1} \\ \boldsymbol{S} \end{array} \begin{bmatrix} \widetilde{\mathbf{A}}'_{01} & \widetilde{\mathbf{A}}'_{00} & & & \\ & \widetilde{\mathbf{A}}'_{01} & \widetilde{\mathbf{A}}'_{00} & & \\ & & \ddots & \ddots & \\ & & & \widetilde{\mathbf{A}}'_{01} & \widetilde{\mathbf{A}}'_{00} \\ & & & & \widetilde{\mathbf{A}}''_{01} \end{bmatrix} - \widetilde{\mathbf{A}}_{\mathbf{0}} \tag{21}$$

$$\widetilde{\mathbf{A}}'_{00} = \mathbf{A}'_{00} \otimes \mathbf{I}$$

$$\widetilde{\mathbf{A}}'_{01} = \mathbf{A}'_{01} \oplus \widetilde{\mathbf{Q}}'$$

$$\widetilde{\mathbf{A}}''_{01} = \mathbf{I} \otimes \widetilde{\mathbf{Q}}'$$

Where $\mathbf{A}'_{01}$ and $\mathbf{A}'_{00}$ are the same as in 2.1.2.

For $m \ge l \ge 1$ the processing rate when k servers are under repair, i semi-finished products are available and l external demands are queued to be served is

$$u_{l,k}^{i}=\begin{cases}min(m,i,l)u, & s\geq k\geq 0,\\ min(s+m-k,i,l)u, & s+m\geq k\geq s+1.\end{cases}$$

Denote

$$\widetilde{\mathbf{A}}_{l2}^{i}=diag\{u_{l,k}^{i},0\leq i\leq S,0\leq k\leq s+m\}$$

Consider the relation between l and k first and then consider their relation with i, $\widetilde{\mathbf{A}}_{l2}^{i}$ can be simplified as

$$\widetilde{\mathbf{A}}_{l2}^{i}=diag\begin{Bmatrix}min(i,l)u,\cdots,min(i,l)u,min(i,l-1)u,\\ \cdots,min(i,2)u,min(i,1)u,0\end{Bmatrix}$$

$$\widetilde{\mathbf{A}}_{l2}=\begin{array}{c}0\\1\\ \vdots\\ S-1\\ S\end{array}\begin{bmatrix}\mathbf{0} & & & & \\ \mathbf{I}\otimes\widetilde{\mathbf{A}}_{l2}^{1} & \mathbf{0} & & & \\ & \ddots & \ddots & & \\ & & \mathbf{I}\otimes\widetilde{\mathbf{A}}_{l2}^{S-1} & \mathbf{0} & \\ & & & \mathbf{I}\otimes\widetilde{\mathbf{A}}_{l2}^{S} & \mathbf{0}\end{bmatrix} \tag{22}$$

$$\widetilde{\mathbf{A}}_{l1}=\widetilde{\mathbf{A}}_{01}-diag(\widetilde{\mathbf{A}}_{l2}\mathbf{e}) \tag{23}$$

$$\widetilde{\mathbf{A}}_{0}=\mathbf{A}_{m0},\widetilde{\mathbf{A}}_{1}=\mathbf{A}_{m1},\widetilde{\mathbf{A}}_{2}=\mathbf{A}_{m2} \tag{24}$$

To solve (19) to (24) we again apply the MGM solution together with iterative method for solving boundary equations. Alternatively, we can use another recursive method as illustrated in Neuts (1994) for solving boundary equations. The detail is omited. The stability condition is similar to 2.1.4 .

2.2.3 Performance Measures

Applying the same expectation formula as in 2.1.5 it is easy to obtain system occupancy level for external order L as in (15). The average semi-finished product is

$$\mathrm{W}=\sum_{i=1}^{S}\sum_{j=J_B}^{-J_A}\sum_{k=0}^{s+m} i\,\tilde{q}_{ijk} \tag{25}$$

where $\mathbf{q}$ is the same as in 2.1.5. Derivation for $\overline{\mathrm{B}}_1$ $\overline{\mathrm{B}}_2$ is similar to 2.1.5.

3. OPTIMAL CAPACITY DESIGN

In the optimization procedure we assume part supply rate, order arrival rate, and machine failure rate are uncontrollable. These assumptions are reasonable in reality. Assume all other system parameters are controllable, For model 1 our objective is to trade-off throughput revenue with costs related to part inventory, semi-finished product inventory and machine and repair operation by optimizing the part buffer size, machine number, repairmen number, production rate, and repair rate. The objective function is expressed as (26).

Model 1 Optimization problem:

$$Max\ \ TP=p\phi-\left[c_h\left(\overline{B}_1+\overline{B}_2\right)+c_wW+c_mm+c_cc+c_uu+c_\sigma\sigma\right] \tag{26}$$

In (26), p is per unit selling price. c_h is per unit holding cost per unit time for part buffer. c_w is per unit per unit time for semi-finished product. c_m and c_c are per unit accrued machine investment and repairman hiring cost per unit time. c_u and c_σ are per unit production and repair cost. Furthermore, decision maker may have to consider other common interest such as maintaining system availability to some desired level. For model 2 we want to trade-off additional spare investment cost and external order waiting cost in addition to afore-mentioned costs. Here the throughput is not of concern since we assume external arrival rate is fixed. Our objective function is expressed as (27).

Model 2 Optimization problem:

$$Min\ \ TC=c_h\left(\overline{B}_1+\overline{B}_2\right)+c_wW+c_lL+c_mm+c_cc+c_ss+c_uu+c_\sigma\sigma \tag{27}$$

Here c_l is order waiting cost per unit per unit time. c_s is per unit accrued spare investment cost per unit time. Since all the performance measures and system stability are related to system design parameters, it becomes extremely complicated if not impossible if we use traditional optimization method such as Lin and Chien (1995) and Zeng and hang (1997) to derive optimality. Therefore we propose a combined enumeration and global search algorithm as listed in figure 3 for the optimization problem of model 2.

```
Let f* = ∞. Set upper-bound and lower bounds x̄ and x̲ for
J_A, J_B, S, m, c, s, u, σ, respectively.
For J_A = J̲_A to J̄_A
  For J_B = J̲_B to J̄_B
    For S = S̲ to S̄
      For m = m̲ to m̄
        For c = c̲ to c̄
          For s = s̲ to s̄
            Apply meta-heuristics or quasi-Newton search.
            If stability condition not met, Then
              f(J_A, J_B, S, m, c, s, u, σ,) = ∞,
            End
            If f(J_A, J_B, S, m, c, s, u, σ,) < f*, Then
              x* = (J_A, J_B, S, m, c, s, u, σ).
            End
          End
        End
      End
    End
  End
End
```

End

Figure 3 Pseudo-code of our optimization algorithm

Since the search space is continuous for each capacity enumeration we use Genetic Algorithm (GA) for continuous variable and Simulated Annealing (SA) to search for u, σ, which is listed in figures 4 and 5.

Step 1	Generate initial random population of chromosomes between allowable range of operating rate and repairing rate.
Step 2	When maximum iteration not met do the following:
Step 2a	Sort according to the fitness value in ascending order. Select only some best chromosomes to mate using rank weighting.
Step 2b	Crossover operation.
Step 2c	Mutation operation.
Step 3	Report the first (which is the best) solution.

Figure 4 GA for our continuous variable search

In Step 2b we use the following linear random combination for exploring new values assuming there are n genes for each chromosome and crossover happens at jth variable (gene).

$$\tilde{x}_{1j} = x_{1j} - \beta(x_{1j} - x_{2j})$$
$$\tilde{x}_{2j} = x_{2j} + \beta(x_{1j} - x_{2j})$$

Before crossover

Parent1:	x_{11}	$\cdots$	$x_{1,j-1}$	x_{1j}	$x_{1,j+1}$	$\cdots$	$x_{1,n1}$
Parent2:	x_{21}	$\cdots$	$x_{2,j-1}$	x_{2j}	$x_{2,j+1}$	$\cdots$	$x_{2,n1}$

After crossover

Child1:	x_{11}	$\cdots$	$x_{1,j-1}$	$\tilde{x}_{1j}$	$x_{2,j+1}$	$\cdots$	$x_{2,n1}$
Child2:	x_{21}	$\cdots$	$x_{2,j-1}$	$\tilde{x}_{2j}$	$x_{1,j+1}$	$\cdots$	$x_{1,n1}$

Step 1	Let $p = 0, k = 0$. Choose starting solution x^k and an initial temperature T^p.
Step 2	Do the following until the maximum iteration for current temperature is met.
Step 2a	Search for new random solution x^{k+1}. Calculate $\Delta f = f(x^{k+1}) - f(x^k)$.
Step 2b	If $\Delta f \leq 0$, update x^{k+1} as new current solution. Else update x^{k+1} as new current solution with probability $e^{-\Delta f/\ T^p}$.
Step 2c	$k = k + 1$. Repeat step 2a.
Step 3	Reduce temperature using appropriate procedure. $p = p + 1$. Repeat step 2.

Figure 5 SA search

We refer to Lindfield and Penny (2012) when implementing the above search algorithms. Quasi-Newton procedure for searching u, σ undue each enumeration is listed in figure 6 (cf. Wang, 2015).

Step 1	Choose an initial solution vector $u^{(0)}$. Set tolerance level ϵ, and initial guess of Hessian, usually we let $\mathbf{H}^{(0)} = \mathbf{I}$. Let n = 0.
Step 2	Calculate search direction $\mathbf{d}$ from the following simultaneous equation: $\mathbf{H}^{(n)}\mathbf{d} = -f'(\mathbf{u}^{(n)})$. Find next solution $\mathbf{u}^{(n+1)}$ along $\mathbf{d}$ using optimal local search.
Step 3	If $f'(\mathbf{u}^{(n+1)})[f'(\mathbf{u}^{(n+1)})]^T < \epsilon$, Stop, optimum $\mathbf{u} = \mathbf{u}^{(n+1)}$. Else go to step 4.
Step 4	Calculate $\Delta\mathbf{u} = \mathbf{u}^{(n+1)} - \mathbf{u}^{(n)}$ and $\Delta\mathbf{g} = f'(\mathbf{u}^{(n+1)}) - f'(\mathbf{u}^{(n)})$. Update $\mathbf{H}^{(n+1)} = \mathbf{H}^{(n)} + \frac{\Delta\mathbf{g}(\Delta\mathbf{g})^T}{(\Delta\mathbf{g})^T\Delta\mathbf{u}} - \frac{\mathbf{H}^{(n)}\Delta\mathbf{u}(\Delta\mathbf{u})^T\mathbf{H}^{(n)}}{(\Delta\mathbf{u})^T\mathbf{H}^{(n)}\Delta\mathbf{u}}$. $n = n + 1$. Repeat step 2.

Figure 6 Pseudo-code of Quasi-Newton

4. NUMERICAL EXAMPLES

For brevity we only report some experimental results. First we use the derivation in section two to investigate the performance of the models. Then we illustrate the optimization of model 2. We use the following parameter setting for model 1:

$$2 \leq J_A,\quad J_B \leq 5, S = \infty, m = 3, c = 2, \lambda_A = \lambda_B = 2.5,$$
$$\theta = 1, u = 5, \sigma = 1.$$

For model 2, we set the parameters as

$$2 \leq J_A = J_B = J \leq 5, 3 \leq S \leq 7, m = 2, c = 2, s = 2, \lambda = 1,$$
$$\lambda_A = \lambda_B = 2.5, \theta = 1, \alpha = 0.5, u = 1.25, \sigma = 1.$$

Figure 7 refers to model 1. Figure 8 to 10 refer to model 2. Both experiments pass stability test in (13). Note that due to symmetry $\overline{B}_1 = \overline{B}_2 = B$. Figure 7 shows more part buffer contribute to more semi-finished products. Figures 8 and 9 show more S and J reduce order occupancy but increase average work-in-process which include part and semi-finished. This is reasonable since more buffers reduce the blocking and contribute to quick order consumption. From Figures 8 and 9 we see the apparent trade-off effect between L and WIP. If only costs related to these two factors are concerned, i.e., $\text{Min TC} = c_h(2B) + c_w W$, figure 10 shows (J, S)=(2, 5) is optimal for $c_h = c_w = 1, c_l = 10$.

Now We use (27) to optimize model 2. First we observe more production and repair capacity reduce order occupancy with the price of more capacity investment cost. The final result will depend on trade-off between buffer capacity (J, S), capacity investment (m, c, s), and rates (u, σ). To experiment we use the following parameter setting:

$$c_h = c_w = 1, c_l = 10, c_m = c_c = c_s = 5, \lambda_A = \lambda_B = 2.5,$$
$$\lambda = 1, c_u = c_\sigma = 2, \theta = 1, \alpha = 0.5.$$

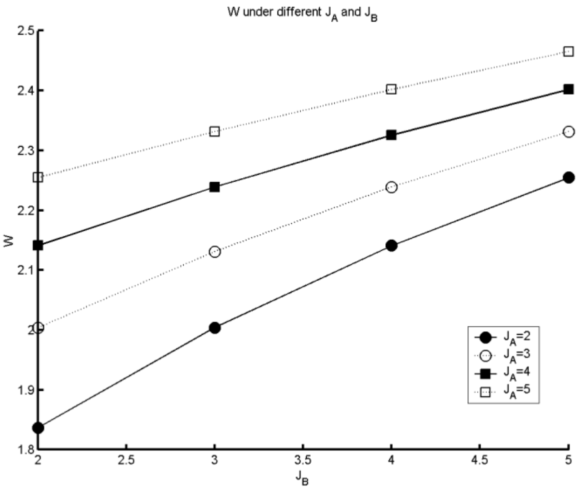

Figure 7: Expected WIP (W) under different J_A and J_B

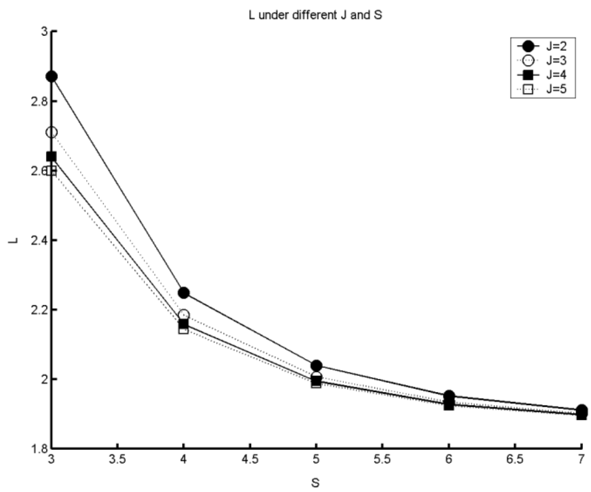

Figure 8: Occupancy level (L) under different J and S

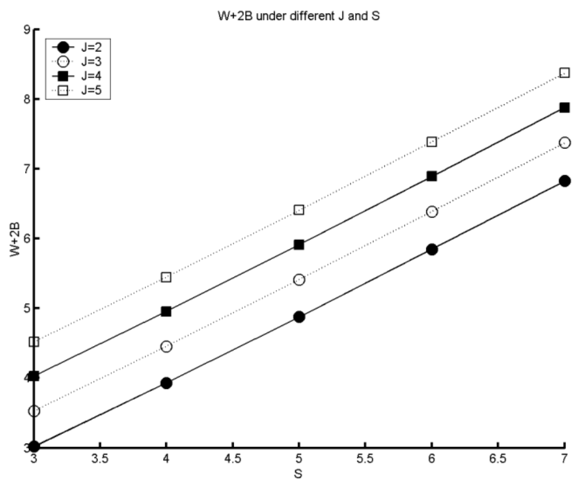

Figure 9: Expected WIP (W+2B) under different J and S

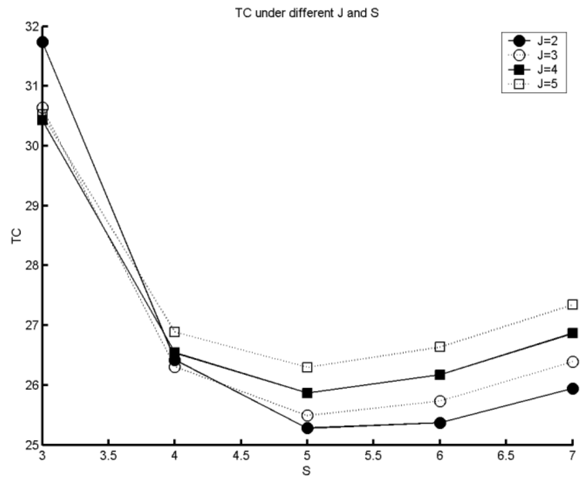

Figure 10: TC as a function of J and S

For demonstration purpose, assume the following capacity range:

$$1 \leq \mathrm{J}, \mathrm{S}, \mathrm{m}, \mathrm{c}, \mathrm{s} \leq 2, 1 \leq \mathrm{u}, \sigma \leq 20.$$

The result is shown in table 1. Denotation of superscript numbers in table 1 is 1: u, 2: σ, 3: TC. In the table besides GA, SA we use QN to denote Quasi-Newton search. Interestingly all numerical search procedures obtain very close results: $[\mathrm{J}, \mathrm{S}, \mathrm{m}, \mathrm{c}, \mathrm{s}] = [2, 2, 1, 1, 1]$ with $[\mathrm{u}, \sigma]$ near $[4.3, 2.6]$ is optimal. The minimal cost is 39.35.

5. Discussion

We model unreliable stochastic assembly systems extended Latouche (1981). To conclude, we propose a queueing model combining maintenance float design with stochastic assembly model which is new in the literature. We derive both pure MTS and hybrid MTS-MTO control mechanisms under the common matrix analytical development framework. After performance evaluation we use traditional search method and state-of-the-art meta-heuristics to search for optimal rates. The homogeneity of the numerical result indicates the proposed search methods are capable of finding optimal (or near optimal) solution in reasonable computation time for sophisticated queueing problems where objective function values are not explicitly expressed but instead derived from a numerical procedure.

However, the cases of more than two suppliers, multi-stage, and non-exponential processing are often encountered in real applications. To handle the above situations, we also develop another generalized stochastic Petri net (GSPN) model. We numerically verify the equivalence between the proposed mathematical and GSPN models. Although GSPN model can be used to explore non-exponential service times, N (≥ 2) suppliers and multi-stage problem, the long computation time and curse of dimensionality problem when state space becomes large remain to be solved in the future study.

ACKNOWLEDGMENTS

This work is financially supported by MOST 104-2622-E-164-002-CC3.

REFERENCES

Altiok, T. (1997) *Performance analysis of manufacturing systems*, Springer.

Bonomi, F. (1987) An approximate analysis for a class of assembly-like queues, *Queueing Systems*, **1**, 289-309.

De Cuypere, E. De Turck, K., Fiems, D. (2012), A queueing theoretic approach to decoupling inventory, in Khalid A.B. et. al (Eds.), *Analytical and Stochastic Modeling Techniques and Applications*, 150-164.

De Cuypere, E. De Turck, K., Fiems, D. (2013) Performance analysis of a kitting process as a paired queue, *Mathematical Problems in Engineering*, Article ID 843184, 10 pages.

Gao, C., Shen, H., Cheng, T.C.E. (2010) Order-fulfillment performance analysis of an assemble-to-order system with unreliable machines, *Int. J. Production Economics*, **126**, 341-349.

Harrison, M. (1973) Assembly-like Queues, Journal of Applied Probability, **10(2)**, 354-367.

Jewkes, E.M., Alfa, A.S. (2009) A queueing model of delayed product differentiation, *European Journal of Operational Research*, **199**, 734-743.

Latouche, G. (1981) Queues with paired customers, *Journal of Applied Probability*, **18(3)**, 684-696.

Lin, C., Chien, T. W. (1995) Maintenance system design problems for a flexible manufacturing system, *Computers & Industrial Engineering*, **28(1)**, 93-105.

Lindfield, G.R., Penny, J.E.T. (2012) Numerical Methods using MATLAB, Academic Press, Elsevier.

Lipper E.H., Sengupta, B. (1986) Assembly-like queues with finite capacity: bounds, asymptotics and approximations, *Queueing Systems*, **1**, 67-83.

Liu, L., Yuan, X.M. (2001) Throughput, flow times, and service level in an unreliable assembly system, *European Journal of Operational Research*, **135**, 602-615.

Neuts, M.F., Lucanton, D.M. (1979) A Markovian queue with servers subject to breakdowns and repairs, *Management Science*, **25(9)**, 849-861.

Neuts, M.F. (1994) *Matrix-Geometric Solutions in Stochastic Models*, Dover Publication Inc., New York.

Ramachandran, S., Delen, D. (2005) Performance analysis of a kitting process in stochastic assembly systems, *Computers & Operations Research*, **32**, 449-463.

Ramakrishnan, R., Krishnamurthy, A. (2012) Performance evaluation of a synchronization station with multiple inputs and population constraints, *Computers & Operations Research*, **39**, 560-570.

Som, P., Wilhelm, W.E. and Disney, R.L. (1994) Kitting process in a stochastic assembly system, *Queueing Systems*, **17**, 471-490.

Wang, F.F. (2015) Approximation and Optimization of a Multi-server Impatient Retrial Inventory-queueing System ith Two Demand Classes, *Quality Technology & Quantitative Management*, **12(3)**, 267-290.

Wilhelm, W.E., Som, P. (1998) Analysis of a single-stage, single-product, stochastic, MRP-controlled assembly system, *European Journal of Operational Research*, **108**, 74-93.

Wilhelm, W.E., Som, P. (1999) Analysis of stochastic assembly with GI-distributed assembly time, *INFORMS Journal on Computing*, **11(1)**, 104-116.

Yuan, X.M., Liu, L. (2005) Performance analysis of assembly systems with unreliable machines and finite buffers, *European Journal of Operational Research*, **161**, 854-871.

Zeng, A.Z, Zhang, T. (1997) Queuing model for designing an optimal three dimensional maintenance float system, *Computers & Operations Research*, **24(1)**, 85-95.

Table 1: Optimal capacity design for hybrid MTS-MTO with unreliable assembly model

J	S	m	c	s	GA			SA			QN		
1	1	1	1	1	7.85 [1]	3.66 [2]	56.76 [3]	7.87 [1]	3.66 [2]	56.76 [3]	7.84 [1]	3.65 [2]	56.76 [3]
1	1	1	1	2	7.64	3.20	59.75	7.58	3.19	59.75	7.59	3.18	59.75
1	1	1	2	1	7.70	2.56	58.77	7.69	2.57	58.77	7.69	2.57	58.77
1	1	1	2	2	7.47	2.03	61.77	7.42	2.05	61.76	7.44	2.04	61.76
1	1	2	1	1	7.63	3.28	60.53	7.66	3.43	60.53	7.66	3.43	60.53
1	1	2	1	2	7.61	3.23	64.66	7.52	3.24	64.66	7.54	3.25	64.66
1	1	2	2	1	7.46	2.17	62.26	7.49	2.20	62.26	7.49	2.20	62.26
1	1	2	2	2	7.44	1.99	66.41	7.39	1.98	66.41	7.39	1.98	66.41
1	2	1	1	1	4.58	2.81	41.68	4.59	2.81	41.68	4.58	2.82	41.68
1	2	1	1	2	4.44	2.71	45.47	4.39	2.57	45.46	4.41	2.57	45.46
1	2	1	2	1	4.51	1.91	44.30	4.49	1.95	44.30	4.49	1.94	44.30
1	2	1	2	2	4.32	1.56	47.99	4.30	1.64	47.99	4.31	1.63	47.99
1	2	2	1	1	3.76	2.83	44.74	3.80	2.83	44.74	3.81	2.83	44.74
1	2	2	1	2	3.65	2.75	49.17	3.68	2.75	49.17	3.70	2.75	49.17
1	2	2	2	1	3.45	2.18	47.22	3.71	1.80	47.02	3.68	1.81	47.02
1	2	2	2	2	3.78	1.72	51.37	3.57	1.69	51.37	3.56	1.69	51.37
2	1	1	1	1	6.35	3.24	48.82	6.38	3.25	48.82	6.41	3.22	48.82
2	1	1	1	2	6.15	2.94	52.20	6.17	2.88	52.19	6.19	2.86	52.19
2	1	1	2	1	6.25	2.27	51.17	6.26	2.26	51.17	6.28	2.26	51.17
2	1	1	2	2	6.12	1.83	54.47	6.08	1.83	54.47	6.06	1.84	54.47
2	1	2	1	1	6.18	3.34	52.94	6.28	3.08	52.87	6.25	3.08	52.87
2	1	2	1	2	6.10	2.89	57.19	6.14	2.96	57.19	6.15	2.95	57.19
2	1	2	2	1	6.15	1.94	54.91	6.12	1.97	54.91	6.10	1.97	54.91
2	1	2	2	2	6.01	1.81	59.20	6.04	1.80	59.20	6.01	1.81	59.20
2	**2**	**1**	**1**	**1**	**4.36**	**2.56**	**39.35**	**4.33**	**2.65**	**39.35**	**4.32**	**2.64**	**39.35**
2	2	1	1	2	4.16	2.44	43.24	4.17	2.43	43.24	4.16	2.43	43.24
2	2	1	2	1	4.61	1.72	42.18	4.22	1.82	42.12	4.23	1.83	42.12
2	2	1	2	2	4.23	1.73	45.93	4.06	1.55	45.89	4.05	1.55	45.89
2	2	2	1	1	3.55	2.67	42.45	3.55	2.68	42.45	3.55	2.67	42.45
2	2	2	1	2	3.40	2.56	46.94	3.43	2.63	46.94	3.45	2.61	46.94
2	2	2	2	1	3.43	1.76	44.87	3.41	1.73	44.87	3.42	1.72	44.87
2	2	2	2	2	3.18	1.61	49.27	3.32	1.61	49.26	3.30	1.62	49.26
CPU time (in seconds)					850.40			7006.64			104.39		