International Journal of Production Research,
VOL. 45, NO. 24, 15 DECEMBER 2007, 5667–5685

# PRODUCTION CONTROL AND COMBINED DISCRETE/CONTINUOUS SIMULATION MODELLING IN FAILURE PRONE TRANSFER LINES

P. Lavoie*, J.-P. Kenné* and A. Gharbi**

*Mechanical Engineering Department, École de technologie supérieure, Integrated Production Technologies Laboratory, University of Québec, 1100 Notre Dame Street West, Montreal, Que., Canada H3C 1K3.

**Automated Production Engineering Department, École de technologie supérieure, Production System Design and Control Laboratory, University of Québec, 1100 Notre Dame Street West, Montreal, Que., Canada H3C 1K3.

## ABSTRACT

In this paper, we consider a production system consisting of multiple tandem machines subject to random failures. The objective of the study is to find the production rates of the machines in order to minimize the total inventory and backlog costs. By combining analytical formalism and simulation based statistical tools such as design of experiments (DOE) and response surface methodology (RSM), an approximation of the optimal control policy is obtained. The combined discrete/continuous simulation modelling is used to obtain an estimate of the cost in a fraction of the time necessary for discrete event simulation by reducing the number of events related to parts production. This is achieved by replacing the discrete dynamics of part production by a set of differential equations that describe this process. This technique makes it possible to tackle optimization problems that would otherwise be too time consuming. We provide some numerical examples of optimization and compare computational times between discrete event and discrete/continuous simulation modelling. The proposed combination of DOE, RSM and combined discrete/continuous simulation modelling allows us to obtain the optimization results in a fairly short time period on widely available computer resources.

## 1. INTRODUCTION

This paper deals with the control problem of a stochastic manufacturing system consisting of multiple machines in flowshop configuration. The stochastic nature of the system is due to the machines that are subject to random breakdowns and repairs. The decision variables are input rates to the machines which influence the number of parts in the buffers between machines and the surplus. The surplus is the difference between cumulative production of the finished goods and cumulative demand. In a practical viewpoint, it is obvious that the work-in-process (WIP) must remain nonnegative. Since the WIP inventories and the surplus are considered as state variables, we are facing a state constrained control problem which is to choose admissible input rates of machines to minimize the inventory/backlog cost over an infinite horizon. Many authors contributed in the sphere of the production planning problem of flexible manufacturing systems. The problem becomes much more complicated with the existence of internal buffers which give rise to optimisation problems with state constraints.

Based on such complexity and observing that explicit optimal solutions as in Akella & Kumar (1986) does not exist, Lou et al. (1994) extended the problem in Akella & Kumar (1986) to a two machine flowshop configuration and conducted a rigorous study of the dynamics of the system and the related boundary conditions. An analysis of the m-machine flowshop ($m \geq 2$) in the context of obtaining a piecewise deterministic optimal control was discussed in Presman et al. (1995). Such a system has been studied in the works of Bai & Gershwin (1994) where the authors constructed a hierarchical controller to regulate the production. It is the purpose of this paper to present an alternative approach to determine the production rates of transfer lines with a large number of machines. The main structure of the framework developed here extends the work of Kenné & Gharbi (1999) and Gharbi & Kenné (2003) dealing with the implementation of simulation based experimental design in the control of manufacturing systems. Discrete event simulation can theoretically achieve any precision level in performance measures. A greater precision is obtained essentially by increasing the length of the runs. Another advantageous characteristic of discrete event simulation is that it doesn't involve restrictive assumptions like specific types of probability distributions of random process. On the other hand, simulation is sometimes deemed too time-consuming. This will be especially true in systems with high variability induced by random failures and high production rates when great precision levels are required. When the only variability in the system comes from machine failures, the length of the runs necessary to reach a certain precision level is a function of the number of failure and repair events. Consequently, for the same failure and repair distributions and required precision level, an increase in production rate will result in an increase of the number of events in a run because the production of every part creates multiple events.

In the literature concerning transfer lines, decomposition techniques occupy a very important place. They have been developed to overcome the lack exact analytical techniques, the complexity of numerical optimisation methods and the computational effort of simulation. They are approximate methods that are based on simpler problems like the exact solution for the 2 machine line developed by Gershwin & Schick (1979). The lines are decomposed in a series of 2 machine lines separated by one buffer, the first machine representing the upstream portion of the line from the buffer and the second machine representing the downstream portion of the line from the buffer. Dallery et al. (1989) introduced the DDX (for Dallery-David-Xie) algorithm that used a continuous flow of material. Burman (1995) expanded the algorithm to non-homogenous lines. Dallery & Le Bihan (1999) extended the algorithm to more general failure and repair distribution with the use of generalized exponential distributions. Schor (1995) and Gershwin & Schor (2000) developed algorithms for two problems: (i) the allocation of a fixed buffer space in order to achieve maximum throughput and (ii) the minimization of buffer space while attaining a fixed throughput. All the contributions mentioned above used the average throughput of the line as a main performance measure and evaluated the average work-in-process at the different stages. They assumed saturated demand meaning finished goods are never stocked. Bonvik et al. (2000) used a finished goods buffer to satisfy a constant demand rate in a system that allows backlogging of unsatisfied demands, under a Hybrid policy. Although in many cases, the algorithm seems to estimate well the average work-in-process, they observe that the average backlog estimate is poor when system utilization is high. While comparing different decomposition techniques, Bonvik (1996) notices that they generally over-estimate the throughput by a few percent. This can make the use of these techniques difficult in cost minimizing applications when backlog is penalized. Sadr & Malhame (2004) have used a decomposition/aggregation based dynamic programming technique to minimize incurred inventory and backlog costs for partially homogenous transfer lines. We also note that a survey of the accuracy of their technique has not been done and only results for 2 machine transfer lines are presented. To our knowledge, very few authors addressed the cost minimization problem using a decomposition technique.

Observing the complexity of the analytical solution, we combine analytical results, mixed continuous/discrete simulation modelling, design of experiments (DOE) and response surface methodology (RSM) to minimize the average inventory and backlog costs. While the combination of analytical modelling, DOE and RSM provides the tools for solving the problem efficiently, the proposed simulation model has the advantage of being much faster than discrete-event simulation, advantage which increases with variability of the system, while obtaining an accurate cost estimate with verifiable precision.

The paper is organized as follows: section 2 presents the mathematical formulation of the optimization problem and its complexity, section 3 presents the proposed approach and section 4 describes the simulation model. The validation of the model is shown in section 5, section 6 compares the cost and computational timing results between the discrete event and combined discrete/continuous simulation model and we give some numerical optimization results in section 7 .

## 2. PROBLEM STATEMENT

In this paper, we consider the flow control problem for a tandem production system with $m$ $(m \geq 2)$ unreliable machines. The system is shown in Figure 1, where $M_i$ denotes the machine $i = 1, \ldots, m$. Each machine has two states (up and down denoted by 1 and 0 respectively), resulting in a system with a $p$-state Markov chain $k(t) = (k_1^1(t), \cdots, k_1^1(t), \cdots, k_1^p(t), \cdots, k_1^p(t))$ on the probability space $(\Omega, F, P)$ with values in a finite set $M$. The capacity of the machine $M_i$ in mode $j$ at time $t$ is denoted by $k_i^j(t)$.

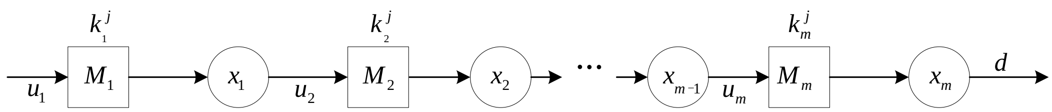



Figure 1: An m-machine one-part-type system

We use $u_i(t)$ to denote the input rate to $M_i$ and $x_i(t)$ to denote the number of parts in the buffer between $M_i$ and $M_{i+1}$, $(i = 1, \cdots, m-1)$. Finally the difference between the cumulative production and the cumulative demand, called surplus, is denoted by $x_m(t)$. The dynamics of the system can be written as follows:

$$\frac{d}{dt}(x_i(t)) = u_i(t) - u_{i+1}(t), \qquad x_i(0) = x_i, \quad i = 1,\cdots,m\text{-}1 \tag{1}$$

$$\frac{d}{dt}(x_m(t)) = u_m(t) - u_{m+1}, \qquad x_m(0) = x_m \tag{2}$$

where $u_{m+1} := d$ is a given constant demand rate. In matrix notation, the system of equations (1)-(2) becomes:

$$\frac{d}{dt}(x(t)) = A \cdot u(t), \qquad x(0) = x \tag{3}$$

where $A$ is an $m \times (m+1)$ matrix, $x(t) = (x_1(t),\cdots,x_m(t))$, $u(t) = (u_1(t),\cdots,u_{m+1}(t))$, $x = (x_1,\cdots,x_m)$. Since the number of parts in the internal buffers cannot be negative, we impose to the state $x_i(t)$ to be nonnegative (i.e., $x_i(t) \geq 0$ for $i = 1,\cdots,m-1$). Related to such a constraint, $x(t) \in S = [0,\infty)^{m-1} \times \mathrm{R}$. For $k = (k_1^j,\cdots,k_m^j),\ k_i^j \geq 0$, let $U(k) = \left\{ u = (u_1,\cdots,u_m):\ \ 0 \leq u_i(t) \leq k_i^j u_{\max}^i \right\}$ where $u_{\max}^i$ is the maximal production rate of the machine $M_i$. The set of admissible controls with respect to $x(t) \in S$ is given by $\Upsilon(x,k)$ defined as follows:

$$\Upsilon(x,k) = \left\{ u' = (u_1,\cdots,u_m):\ \ u' \in U(k);\ x_i = 0 \Rightarrow u_i - u_{i+1} \geq 0,\ i = 1,\cdots,m\text{-}1 \right\}$$

The control problem consists of finding an admissible control policy $u(\cdot)$ that minimizes the cost function $J(\cdot)$ given by:

$$J(x,k,u) = \mathrm{E}\left\{ \int_0^\infty e^{-\rho t} \left[ g(x(t)) \right] dt \mid x(0) = x,\ k(0) = k \right\} \tag{4}$$

where $g(x(t))$ denotes the cost of inventory/backlog and $\rho$ is the discount rate. The value function of the planning problem is given by:

$$v(x,k) = \inf_{u \in \Upsilon(x,k)} J(x,k,u) \tag{5}$$

It is shown in relevant control literature (see Presman et al. (1995), Lou et al. (1994) and Kenné & Boukas (1997) for details) that the value function given by equation (5) is strictly convex in $x$ and continuously differentiable. Moreover, it satisfies a set of coupled partial derivatives equations, namely HJB equations, derived from the application of the dynamic programming approach. Because we need an appropriate boundary condition, we can not use classical methods such as the one based on the viscosity solution. We shall use the concept of directional derivatives (DD) for our investigation, applied to the value function which was shown in the literature to be a convex function. We obtain that the resulting optimality conditions, called HJBDD (Hamilton-Jacobi-Bellman in terms of DD) can be given by:

$$\rho v(x,k) = \min_{u \in \Upsilon(x,k)} \left\{ v'_{Ax}(x,k) + g(x) + Q(\cdot) v(x,\cdot) \right\} \tag{6}$$

with $f'_p(x) = \langle \nabla f(x), p \rangle$ where $\nabla f(x)$ is the gradient of $f(x)$ and $\langle \cdot,\cdot \rangle$ is the scalar product.

The optimal control policy $u^*(\cdot)$ is obtained when the value function is available. But an analytical solution of the HJBDD equations given by (6) is almost impossible to obtain in a general case. One way to find the solution of such equation is to apply numerical methods. Using the Kushner approach (see Kushner & Dupuis (1992) and Kenné & Boukas (1997)), we obtain a discrete Markovian decision control problem with finite state space and finite action characterized by the following $m$-dependent dimension:

$$Dim(m) = p \times 3^{m \times p} \times \prod_{j=1}^{m} N_h(x_j) \tag{7}$$

where $p = 2^m$ and $N_h(x_j) = \text{card}\left[ G_h(x_j) \right]$ with $G_h(x_j)$ describing the numerical grid for the state variable $x_j$ related to the buffer $j$. Each machine has two states (i.e., $p = 2^m$ states for a $m$-machine manufacturing system) and its production rate can take three values namely maximal production rate, demand rate and zero for each mode (i.e., $3^{m \times p}$ states for a $m$-machine, $p$-mode manufacturing system). Numerical algorithms such as policy iteration or policy improvement (see Kushner & Dupuis (1992)) can not be implemented on today's computers for large transfer lines classified in the control literature as complex systems.

Due to such a complexity, different analytical approaches, such as hierarchical control, and heuristical methods have been used to obtain the approximation of the optimal control policy for complex systems. Nevertheless, the solution of simpler problems, like the one machine one part type system of Akella & Kumar (1986), points to the so-called hedging-point policy which consists of building up a defined inventory level whenever it is possible. Extending this policy to the tandem line control problem, we obtain the so-called decentralized hedging point policy

(DHP). The following structure is adopted from available results, obtained from the aforementioned approaches applied to small size systems:

$$u_i(\cdot) = \begin{cases} u_{\max} & if \;\; x_i(t) < Z_i \\ d_i(t) & if \;\; x_i(t) = Z_i \\ 0 & if \;\; x_i(t) > Z_i \end{cases} \Bigg\} i = 1, \ldots, m. \tag{8}$$

with $u_i(t)$ being the production rate of machine $M_i$ at time $t$ and $d_i(t)$ the demand rate at buffer $B_i$ at time $t$. Demand $d_m(t)$ is constant at rate $d$. This policy is based on the hedging point policy, $Z_i$ being the local hedging point and $d_i(t)$ the local demand. This type of policy is deemed a good way of obtaining a satisfying sub-optimal policy. It is also called finite buffer policy. It is equivalent to the Kanban policy when it is implemented through the use of production authorization cards. Since the production rates are fixed by this control policy, the optimization now consists of minimizing the average total cost in regards to the input vector $Z = (Z_1, \ldots, Z_m)$. In the next section, we present the proposed approach for obtaining the optimal parameters of the control structure given by (8).

## 3. PROPOSED APPROACH

Because the complexity of the problem presented makes it practically impossible to solve analytically or numerically today, we propose a simulation optimization approach to find the input parameters of the control policy presented in (8). The performance measure is the expected inventory and backlog cost $\bar{C}(Z)$ for a given input capacity vector $Z$ given by equation (9).

$$\bar{C}(Z) = c^+ \bar{x}^+ + c^- \bar{x}^- \tag{9}$$

where $c^+$ and $c^-$ are the unit cost for inventory and backlog respectively and $\bar{x}^+$ is the average in-process and finished goods inventory calculated as:

$$\bar{x}^+ = \lim_{T \to \infty} \int_0^T x^+(t) dt / T \tag{10}$$

with $T$ being the planning horizon and $x^+(t)$ the sum of the buffer levels at time t as shown by:

$$x^+(t) = \sum_{i=1}^{m-1} x_i(t) + \max(x_m(t), 0) \tag{11}$$

We assume that the unit cost for inventory is constant along the line. The instantaneous ( $x^{-}(t)$ ) and average ( $\bar{x}^{-}$ ) backlog levels are respectively given by equations (12) and (13).

$$x^{-}(t) = \max(0, -x_m(t)) \quad (12)$$

$$\bar{x}^{-} = \lim_{T\to\infty} \int_0^T x^{-}(t)dt / T \quad (13)$$

Figure 2 illustrates the concepts of average inventory and backlog for a one machine and a finished goods buffer using continuous flow. Figure 2 a) shows the trajectory of the surplus in time while Figure 2 b) shows the equivalent average inventory and Figure 2 c) the equivalent average backlog for the same sequence of events.

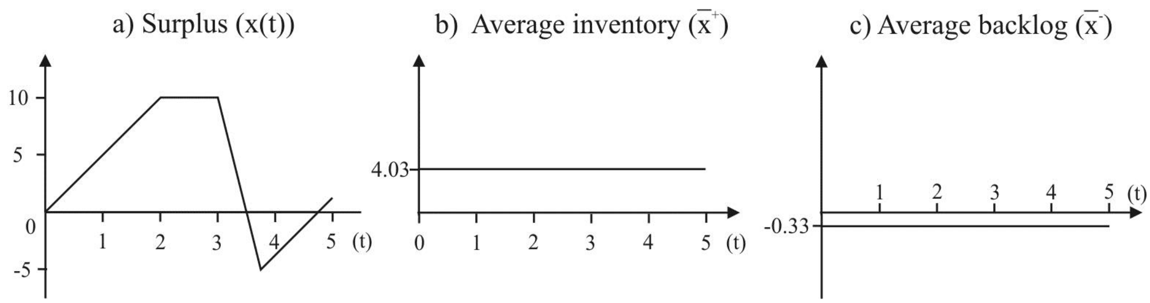



Figure 2: Average inventory and backlog

The optimization problem of section 2 is now reduced to minimizing equation (9) in regards to input $Z$. The value of $Z$ that accomplishes this is noted $Z^*$. In order to find $Z^*$, we must combine simulation and an optimization technique. We believe simulation is still the most effective tool today, being precise and flexible enough to estimate the operating cost of a general system. There is abundant literature illustrating the power of simulation as a tool for design, planning, scheduling and control of complex systems. In the literature, not many simulation optimization techniques have been proposed for the transfer line control problem. Typically, all configurations of a certain domain of a control mechanism are simulated and the best solution is taken from as in Bonvik (1996). This is only possible for small transfer lines with small domains for each parameter. For example, with a four machine line and very few values for each parameter, Bonvik's case gave several thousand’s of cases to be simulated. A six machine line with only 10 possible values for each parameter gives 1 000 000 cases. We rapidly conclude that this technique cannot be used for general cases. Kleijnen & Gaury (2003) proposed the use of genetic algorithms (GA) to optimize the short term robustness of different control mechanisms but also note that GAs are "notoriously slow" and that their proposed approach could be improved by using faster optimization heuristics and simulation models .

With the development of quality engineering, DOE related tools (see Montgomery (2005)) also represent an important class of simulation optimization techniques. RSM has the advantage of proposing a polynomial model of the cost function in relation to the parameters of the proposed control law. Through nonlinear programming, we are able to quantify the sub-optimal control law of the stochastic optimal control problem stated previously. Further more, this polynomial model enables us to conduct sensitivity analyses without necessitating additional simulation experiments. This is not possible using direct search algorithms such as genetic algorithms, tabu search or gradient based techniques. Given the convexity properties of the value function, as mentioned in section 2, it can be approximated by a second degree function precisely enough when the domain is chosen correctly. An efficient way of obtaining the approximating second degree function is by using a Box-Wilson type experimental design as described in Montgomery (2005). This type of design has the advantages of being orthogonal, rotatable and allowing the estimation of squares of factors and all interactions between two factors while needing much less experiments than the 3 level factorial design. From experience we know that many interactions and squares of factors are statistically significant. Although DOE enables the efficient design of the experiments to be conducted and RSM allows us to obtain an estimated response surface, we still need a tool precise enough to estimate the average operating cost of the system

Discrete event simulation gives all the flexibility needed to solve this problem. However, the computational effort of discrete event simulation has often made it impractical. Because of the discrete nature of parts in discrete event simulation, there are numerous events linked to operations on the parts like the beginning and end of an operation. Figure 3 and Figure 4 show a simple one machine example with $u_{\max} = 2$ and $d = 1$. These figures are provided to illustrate the difference in number of events between discrete events and combined discrete/continuous simulation modelling.

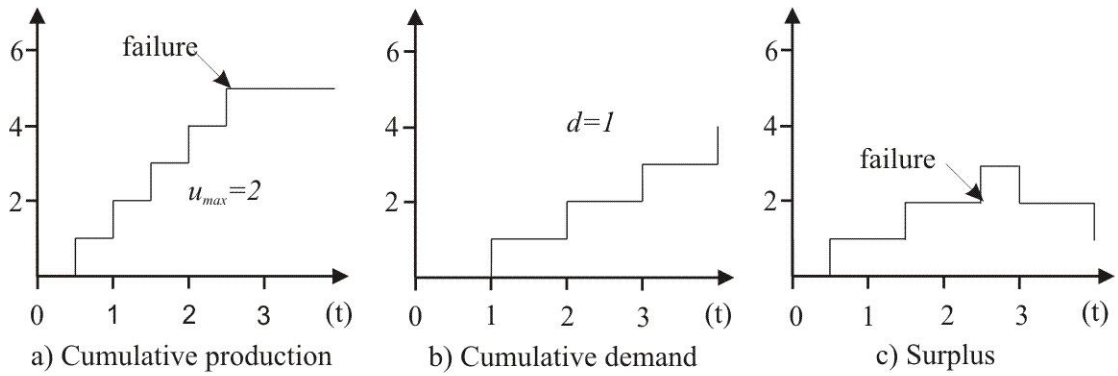



Figure 3:  Discrete event simulation modelling graphical example

Figure 3 (a) presents the cumulative production, 3 (b) the cumulative demand and Figure 3 (c) shows the resulting surplus, which is the difference between 3 (a) and 3 (b).  We see that the production and demand processes evolve by steps in the discrete model.  This results in the generation of numerous events.  In this figure, we see that events corresponding to the production of 5 parts, 4 demands and 1 failure are generated.  We notice in Figure 4 that for the same observation time, much less events are generated than in the discrete model shown at Figure 3. We see that only the failure event is generated.

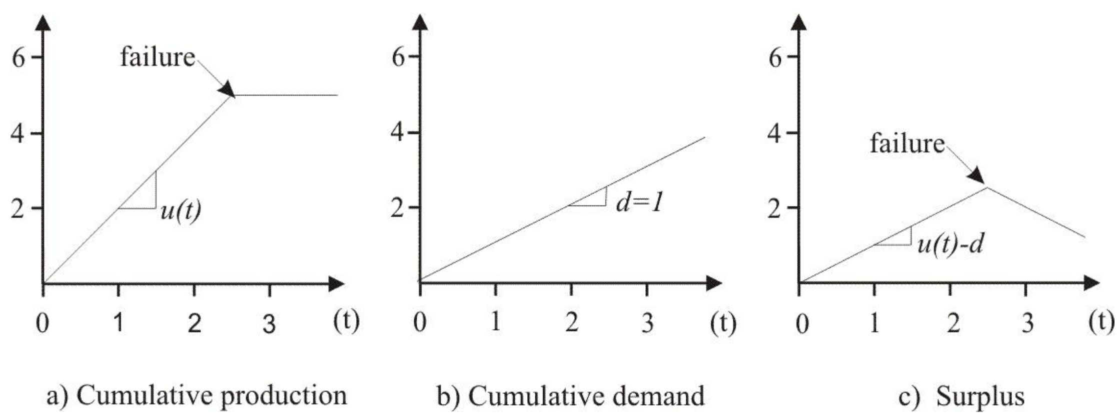



Figure 4:  Combined discrete/continuous simulation modelling graphical example

If the failure event happened at time $t = 100$, there would have been the production of 200 units and 100 demand events before the failure, while the combined discrete/continuous model would still only need to generate the failure event.  Instinctively, for a certain relative precision level of the performance indicators, we can acknowledge that the more variability in the system or the higher the production rate, the more events will be needed in a simulation run. The number of events also increases according to the number of machines in the system. The greater the number of events, the greater the computational time needed. The number of experiments in a design also increases with the number of parameters (machines). In some cases, the experimental design will take days or even weeks to be completed given the desired precision and computer resources used. The problem would be even more obvious if we were using a discrete optimization technique to solve a problem with a large domain for $Z$. Based on the observations made at Figure 3 and Figure 4, we propose the use of combined discrete/continuous modelling. In this model, the discrete part production and stock dynamics are replaced by a set of differential equations that describe the process. Therefore, instead of generating events for every part, the model skips to machine state changing events (failures and repairs) as well as control policy threshold crossing events. The maximum and minimum allowable time steps are specified in the simulation model. The maximum time step will be taken when no discrete event takes place within this time step. Otherwise, the length of the time step will be equal to the time to the next discrete event (machine state changing or threshold crossing event). The differential equations are based on equations (1) and (2) with the production rates set by the control equation (8). They also include binary variables that describe the states of the machines. The state equations do not require the use of the integration algorithm. Only the time persistent statistics equations (10) and (13) require the use of this algorithm, which keeps the time necessary for integration relatively small. As we will see in section 6, this reduction in events will result in greatly reduced computational time. In the next section, we describe the combined discrete/continuous modelling, time advancing algorithm and the data collection for time persistent statistics. This modeling decision becomes necessary in many cases to complete the optimization of the control policy parameters.

## 4. DISCRETE/CONTINUOUS SIMULATION MODELLING

The combined discrete/continuous simulation model is developed using the Visual SLAM language (Pritsker & O'Reilly (1999)) with C sub-routines. The Visual SLAM portion is composed of various networks describing specific tasks (failure and repair events, threshold crossing of inventory variables, etc...). The model is shown at Figure 5 with the following descriptions of the different blocks.

1) The INITIALIZATION block sets the values of $Z$, the demand rate and the machine parameters such as $u_{\max}$, mean time to failure ($MTTF$) and mean time to repair ($MTTR$). The maximum and minimum time step specifications for integration of the cumulative variables and allowable errors are also assigned at this step as well as the simulation time $T$ and the time for the warm up period after which statistics are cleared.

2) The DEMAND RATE is constant and defined in the INITIALIZATION block. It is shown here as an individual block to facilitate comprehension since it is constantly used as an input in the state equations.

3) The CONTROL POLICY is implemented through the use of observation networks that raise a flag whenever one of the thresholds is crossed. The production rates of the machines are then set according to equation (8).

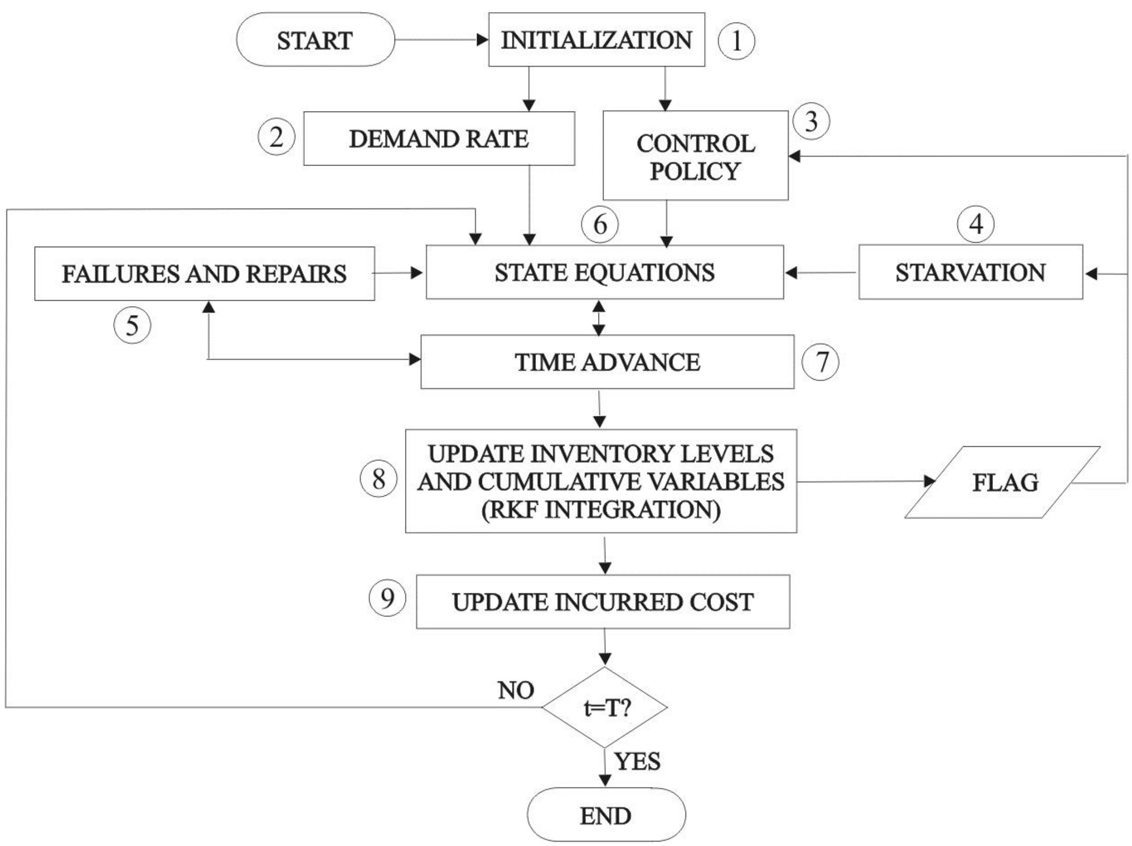



Figure 5: Simulation block diagram

4) The STARVATION of the machines is implemented with the use of observation networks. Whenever one of the in-process buffers becomes empty, a flag is raised. Another signal is sent when material becomes available for operation. Starvation is integrated in the state equations by the means of binary variables multiplying the production rates .

5) The FAILURES AND REPAIRS block samples the times to failure and times to repair for the machines from their respective probability distributions. The operational states of the machines are incorporated in the state equations by the means of logical variables multiplying the production rates.

6) The STATE EQUATIONS are equations (1), (2), (11) and (12) defined as a C language insert. They describe the inventory and backlog variables using the production rates set by the control policy and the binary variables from the failure and repair and starvation networks.

7) The TIME ADVANCE block uses an algorithm provided by Visual SLAM. It is a combination of discrete event scheduling (failures and repair), continuous variable threshold crossing events and time step specifications.

8) The UPDATE INVENTORY LEVELS AND CUMULATIVE VARIABLES block is used once the time step is chosen. The cumulative variables are integrated using the Runge-Kutta-Fehlberg (RKF) method as described in Pritsker & O'Reilly (1999).

9) The UPDATE INCURRED COST block calculates the incurred cost according to the levels of the different variables and the unit costs $c^-$ and $c^+$.

The simulation ends when current simulation time t reaches the defined simulation period $T$. To obtain the average backlog and inventory levels, the cumulative variables are divided by $T$. One understands that the simulation time $T$ cannot be infinite as in equations (10) and (13). Therefore, we ran offline simulations to determine the time necessary for the system to reach its steady state. We found that for our models, this corresponded approximately to 10 000 times MTTF. This duration is therefore used for all the simulations. Multiple replications are conducted. The resulting samples will be useful in performing the hypothesis test between the discrete and combined discrete/continuous simulation results. In order to verify that both the discrete and combined discrete/continuous simulation models produce the same results, we will perform a hypothesis test on the average costs obtained with both models. The null hypothesis for the test is $H_0 : \tilde{C}_1 = \tilde{C}_2$, with $\tilde{C}_1$ and $\tilde{C}_2$ being the asymptotical costs over an infinite horizon for the two compared models. Therefore, the alternative hypothesis is $H_1 : \tilde{C}_1 \neq \tilde{C}_2$. We use a significance level $\alpha = 5\%$. The standard distribution of the difference between two means, assuming that $H_0$ is true is distributed according to:

$$Z_h = \frac{(\bar{C}_1 - \bar{C}_2)}{\sqrt{\frac{S_1^2}{n_1} + \frac{S_2^2}{n_2}}} \tag{14}$$

with $\bar{C}_1$ and $\bar{C}_2$ being the sample averages, $S_1$ and $S_2$ the sample variances calculated by means of equation (15) with $C_{ij}$ being the individual run results of each samples.

$$S_i = \sqrt{\frac{\sum\left(C_{ij} - \bar{C}_i\right)^2}{n-1}};\ i = 1, 2.;\ j = 1, \ldots, n. \tag{15}$$

We reject $H_0$ if $Z_h > z_{\alpha/2}$ or $Z_h < -z_{\alpha/2}$, otherwise we accept $H_0$. In the next section, we present the validation of the simulation model.

## 5. VALIDATION OF THE SIMULATION MODEL

To verify the accuracy of the model, we proceed in two steps: first the dynamics of the stocks is verified graphically to see if the model works according to the control policy and then it was compared with results presented in Chiang et al. (1999). This comparison aims to validate the accuracy of the model with existing results because it is of no use to obtain results quickly if they are incorrect. Figure 6 is a graphical illustration of the trajectories of the buffer levels and surplus.

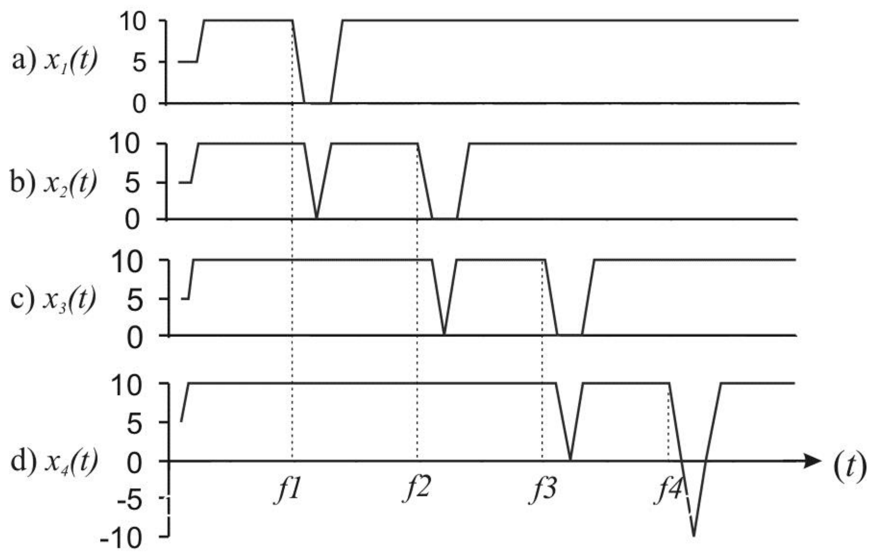



Figure 6: Stock dynamics

Figure 6 (a)-(d) represent the evolution of the stocks $x_i$, $i = 1, 2, 3, 4.$, in a $m = 4$ workstation transfer line. The illustration shows 4 failures happening sequentially from the first machine to the last (*f1*-*f4*). In the beginning, all stocks contain 5 units. Since all machines are up and no machine has reached it's hedging level, they produce parts at rate $u_{\max}$. The first stock to fill itself is $x_4$. When it is full, stock $x_3$ fills itself and so on. When $M_1$ fails (*f1*), $M_2$ keeps $x_2$ at hedging point as long as it can using parts from $x_1$. When $x_1$ becomes empty, the failure effect propagates to $M_2$ by starving it. When the failure is repaired, the first buffer to fill is $x_2$. When it is full, the next buffer to fill itself is $x_1$. We see that only buffer $x_4$ takes negative values, being the finished goods buffer.

To validate our model, it was compared with Chiang et al. (1999). In this study, multiple lines are simulated to obtain maximum throughput under a certain configuration and saturating demand. We have compared our model with the $m$ =7 machine transfer line of case 1 presented in Chiang et al. (1999), being the only homogenous line case in this study. Machines have Markovian failure and repair processes. The transfer line is homogenous with $MTTF$ =10, $MTTR$ =10/9, $Z_i = 2$ and $u_{\max}$ =1. Simulations are run for 32 000 000 time units after a

32 000 000 time units warm-up period, the time unit used being 1/200 of MTTR. $\tilde{PR}$ is the resulting estimated production rate. Zero initial conditions are assumed. The results are shown at Table 1.

Table 1: Validation results for average throughput

| Model | $m$ | $u_{max}$ | $MTTF$ | MTTR | $\tilde{PR}$ |
|---|---|---|---|---|---|
| Chiang et al. (1999) | 7 | 1 | 10 | 1.1 | .7682 |
| Proposed model | 7 | 1 | 10 | 1.1 | .7684 |
| | | | | Diff. | 0.02% |

We see that the difference between the two results is of 0.02%. We therefore consider the difference insignificant and conclude that the models generate identical results and our model is validated. In the next section we present the comparison between discrete event and mixed discrete/continuous simulation in regards to computational time and average cost estimation.

## 6. COMPARATIVE STUDY OF SIMULATION MODELLING APPROACHES

In this section, we compare the cost obtained and the necessary computational time for a discrete event and the proposed combined discrete/continuous model. The objective is to validate the use of the combined discrete/continuous model for cost evaluation in discrete manufacturing systems optimization while measuring the computational time advantage of the combined discrete/continuous model. This comparison has a double objective : 1) We want to measure the computational time advantage of the combined model over the discrete model and 2) Verify we do not sacrifice the accuracy of the results in order to obtain this reduction in computational time. We have evaluated six cases, three four machine lines and three six machine lines. Considered transfer lines are homogenous (i.e all the machines have identical $MTTF$, $MTTR$ and $u_{\max}$ parameters). These cases are presented in Table 2.

Table 2: Machine parameters for the presented cases

| CASE | $MTTF_i$ | $MTTR_i$ | $m$ | $c^-$ | $c^+$ | $d$ | $m$ | $u_{\max}$ |
|---|---|---|---|---|---|---|---|---|
| 1 | 100 | 3 | 4 | 100 | 1 | 1 | 4 | 1.1 |
| 2 | 700 | 21 | 4 | 100 | 1 | 1 | 4 | 1.1 |
| 3 | 1500 | 45 | 4 | 100 | 1 | 1 | 4 | 1.1 |
| 4 | 100 | 3 | 6 | 100 | 1 | 1 | 6 | 1.1 |
| 5 | 700 | 21 | 6 | 100 | 1 | 1 | 6 | 1.1 |
| 6 | 1500 | 45 | 6 | 100 | 1 | 1 | 6 | 1.1 |

At the beginning of each experiment, a warm-up period corresponding to $100 \cdot MTTF$ after which the cumulative variables were reinitialized was used. This warm-up period is a common practice in simulation to eliminate the data from a transient period resulting from the start-up of the system. The data collecting portion of the simulations ($T$) were run for a time period equal to $10000 \cdot MTTF$ in every case, which is the time that showed necessary to reach steady state. By using MTTF as a parameter for the simulation length, we obtain simulation runs with an average of failure and repair events per machine equal in all cases. Table 3 shows the results for the combined discrete/continuous and discrete event models, the average cost obtained ($\tilde{C}$), the computational time ($C.T.$) in seconds and the value of $Z_h$ computed from (14). The $H_0$ column presents the results of the hypothesis test, Y meaning that $H_0$ is accepted and N meaning it is rejected.

Table 3: Cost comparison and timing results

| Case | Model | $Z_1$ | $Z_2$ | $Z_3$ | $Z_4$ | $Z_5$ | $Z_6$ | $\tilde{C}$ | $n$ | C.T. | $Z_h$ | $H_0$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Discr. | 20 | 20 | 20 | 20 | -- | -- | 74.27 | 30 | 420 | -1.96 | Y |
| | Cont. | 20 | 20 | 20 | 20 | -- | -- | 74.57 | 30 | 23 | | |
| 2 | Discr. | 70 | 70 | 70 | 70 | -- | -- | 417.68 | 30 | 2300 | -0.72 | Y |
| | Cont. | 70 | 70 | 70 | 70 | -- | -- | 422.33 | 30 | 33 | | |
| 3 | Discr. | 150 | 150 | 150 | 150 | -- | -- | 906.28 | 30 | 4300 | 0.45 | Y |
| | Comb. | 150 | 150 | 150 | 150 | -- | -- | 899.63 | 30 | 33 | | |
| 4 | Discr. | 20 | 20 | 20 | 20 | 20 | 20 | 109.93 | 30 | 750 | 1.28 | Y |
| | Comb. | 20 | 20 | 20 | 20 | 20 | 20 | 109.73 | 30 | 55 | | |
| 5 | Discr. | 70 | 70 | 70 | 70 | 70 | 70 | 557.73 | 30 | 4100 | 0.07 | Y |
| | Comb. | 70 | 70 | 70 | 70 | 70 | 70 | 557.08 | 30 | 75 | | |
| 6 | Discr. | 150 | 150 | 150 | 150 | 150 | 150 | 1183.86 | 30 | 8200 | -0.08 | Y |
| | Comb. | 150 | 150 | 150 | 150 | 150 | 150 | 1185.36 | 30 | 75 | | |

We notice that the computational times of the combined discrete/continuous model are much smaller than that of the discrete model while the hypothesis tests show no reason to reject the null hypothesis stating that both models produce identical results. We also notice that the greater the values of $MTTF$ and $MTTR$, the longer the computational time for the discrete model. However, the values of $MTTF$ and $MTTR$ do not affect significantly the computational time of the combined discrete/continuous model because it depends mainly on the number of failure and repair events generated throughout the run, which was made to be equal for all lines with the same number of workstations. In the cases presented here, we chose to vary the values of $MTTF$, $MTTR$ and buffer levels accordingly but keeping $d$ and $u_{\max}$ constant. The same result would have been obtained by raising $d$, $u_{\max}$ and the buffers but keeping $MTTF$ and $MTTR$ constant. The ratio of discrete event to combined discrete/continuous computational times is in the hundreds in some cases, while computational times reach more than an hour and a half per run. If only a few experiments are needed, this can be acceptable. But in studies that involve several hundreds or thousands of experiments, such as optimization problems, this is highly unpractical. For example, an experimental design similar to the one used in the next section, ran with the discrete model for cases 5 and 6 would have respectively taken 1 and 2 weeks to complete the 230 experiments instead of 3 hours in the case of the combined discrete/continuous model. In a real system, where one could want to compare different design options or compare multiple control policies at their optimal configuration, considering that a design can be repeated several times, to adjust domain specifications for example, the computational time will be counted in months with the discrete model. With today's continuously decreasing product life cycle durations, it is hardly thinkable that such a delay would be acceptable. However, the computational time of the combined discrete/continuous model will still be counted in hours or days at the most. It could even be run every week or every month to update the policy according to changing production plans or machine parameters. In the case of longer lines the advantage is even greater. For example, one simulation for a 10 machine line with parameters similar to case 6 would take about 13 700 seconds while the combined discrete/continuous model would take under 180 seconds. Noting that the CCD used in our experiments replicated 5 times would necessitate 5230 runs, this would add up to 827 days for the discrete event model but only 11 days for the combined discrete/continuous model. The advantage of the combined model is clear.

These experiments were conducted on an Intel Pentium IV mobile 1.8GhZ processor. In all combined discrete/continuous cases, the maximum step size allowed was set to the same value as *MTTF* which gave the lowest computational times for our model. The computational times given here depend highly on the computer used to conduct the experiments. Therefore, their main use is for comparison purposes. The computational time gives an insight at the approximate time that a simulation experiment will take at the time this paper is written using widely available computer resources. Memory space necessary for running the simulations and storing the results is negligible.

## 7. OPTIMIZATION RESULTS AND ANALYSIS

In the previous section, we have observed that discrete event simulation is much more time consuming than combined discrete/continuous simulation. It was also noticed that for all the cases tested, the resulting cost was statistically identical for both models. Because the time to conduct the optimization of the discrete model is in the order of days and even weeks, this model is greatly prohibiting. The use of the combined discrete/continuous model becomes necessary to solve this problem in a reasonable time frame. In this section we present the optimization results for the 6 same cases, presented in Table 2. These are numerical examples to illustrate the use of the proposed technique.

The experimental design is obtained and analyzed using Statgraphics software. The function approximation we want to obtain is of second degree. An efficient design to obtain such an approximation is the Central Composite Design (CCD, Box-Wilson). For more details, we refer the reader to Montgomery (2005). The experimental design used is a Box-Wilson type design with 2 center points replicated 4 times giving 130 runs for the 4 machine line and 230 runs for the 6 machine line. From such a design we will obtain 2 responses: one surface estimating the average inventory and another one the average backlog. These responses are in the form of equation (16).

$$Y \approx \beta_0 + \sum_{i=1}^{m} \beta_i Z_i + \sum_{i=1}^{m} \sum_{\substack{j=1 \\ j \geq i}}^{m} \beta_{ij} Z_i Z_j \qquad (16)$$

When an acceptable portion of the variance of each response is explained by the models, these functions are multiplied by the corresponding unit cost, ($c^+$ for the average inventory and $c^-$ for the average backlog) and added together. The resulting function is then minimized using nonlinear programming as described in Venkataraman (2002).

Table 4 shows the ANOVA for the inventory response of the 4 machine line corresponding to case 1. We notice that the $R^2$ coefficient is almost 100%. The inventory level is explained almost entirely by this function. If the P-Value is higher than 5%, (for a 95% confidence level) we consider the source non significant. We notice that the F-Ratios for the individual factors are very important in comparison with the interactions and the factor squares. This means that the surface is almost a hyper-plane, with a slight curve. The inventory level increases almost linearly with the factor levels. Table 4 and Table 6 show only the Sources of variance which were considered significant in the analysis. The resulting function is shown at equation (16). The values of the coefficients for the inventory response are presented in Table 5.

Table 4: ANOVA for inventory response

| Source | Sum | Df | Mean Square | F-Ratio | P-Value |
|---|---|---|---|---|---|
| A:Z1 | 535.57 | 1 | 535.57 | 648994.49 | 0 |
| B:Z2 | 466.263 | 1 | 466.263 | 565009.09 | 0 |
| C:Z3 | 427.654 | 1 | 427.654 | 518223.38 | 0 |
| D:Z4 | 399.539 | 1 | 399.539 | 484154.93 | 0 |
| AA | 0.463335 | 1 | 0.463335 | 561.46 | 0 |
| AB | 0.1114 | 1 | 0.1114 | 134.99 | 0 |
| BB | 0.0693565 | 1 | 0.0693565 | 84.05 | 0 |
| CC | 0.0096789 | 1 | 0.00967892 | 11.73 | 0.0009 |
| blocks | 1.89796 | 9 | 0.210884 | 255.55 | 0 |
| Total error | 0.0924258 | 112 | 0.00082523 | | |
| Total (corr.) | 1831.8 | 129 | | | $R^2$ = 99.995 % |

Table 5: Polynomial coefficients for the inventory response surface

| $\beta_0$ | $\beta_1$ | $\beta_2$ | $\beta_3$ | $\beta_4$ | $\beta_{11}$ | $\beta_{12}$ | $\beta_{22}$ | $\beta_{33}$ |
|---|---|---|---|---|---|---|---|---|
| -11.548 | 1.5151 | 1.2988 | 1.1001 | .9897 | -2.547E-2 | -9.329E-3 | -9.856E-3 | -3.682E-3 |

For the backlog results, the variance of the residuals increased with the mean value of the backlog. In order to obtain uniformity of the variance and increase the $R^2$ coefficient, a transformation was used. The transformation that works best in our case is the fractional exponent transformation. For more detail on transformations and their selection we refer the reader to Montgomery (2005). Table 6 shows the results of the ANOVA for the backlog with a ½ exponent. We notice that $R^2$ =98.72% which is still very high. Nevertheless, such a high level of explanation is necessary since the penalty for the average backlog level is much higher than that of the inventory level in the cases studied. The response function coefficients are given in Table 7.

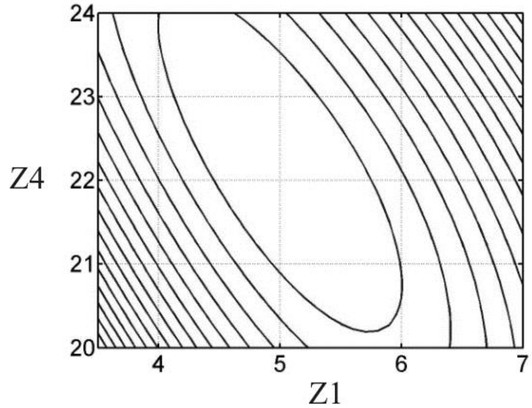
Table 6: ANOVA for backlog$^{1/2}$

| Source | Sum of Squares | Df | Mean Square | F-Ratio | P-Value |
|---|---|---|---|---|---|
| A:Z1 | 0.178689 | 1 | 0.178689 | 1619.78 | 0 |
| B:Z2 | 0.1502 | 1 | 0.1502 | 1361.53 | 0 |
| C:Z3 | 0.137066 | 1 | 0.137066 | 1242.47 | 0 |
| D:Z4 | 0.152808 | 1 | 0.152808 | 1385.17 | 0 |
| AA | 0.00570514 | 1 | 0.00570514 | 51.72 | 0 |
| AB | 0.00937076 | 1 | 0.00937076 | 84.94 | 0 |
| AC | 0.00289 | 1 | 0.00289 | 26.2 | 0 |
| AD | 0.000468939 | 1 | 0.000468939 | 4.25 | 0.0417 |
| BB | 0.00182986 | 1 | 0.00182986 | 16.59 | 0.0001 |
| BC | 0.00406175 | 1 | 0.00406175 | 36.82 | 0 |
| BD | 0.000795622 | 1 | 0.000795622 | 7.21 | 0.0084 |
| CC | 0.000872344 | 1 | 0.000872344 | 7.91 | 0.0059 |
| CD | 0.00146932 | 1 | 0.00146932 | 13.32 | 0.0004 |
| blocks | 0.263524 | 9 | 0.0292804 | 265.42 | 0 |
| Total error | 0.0118039 | 107 | 0.000110317 | | |
| Total (corr.) | 0.921627 | 129 | | | $R^2$ = 98.72% |

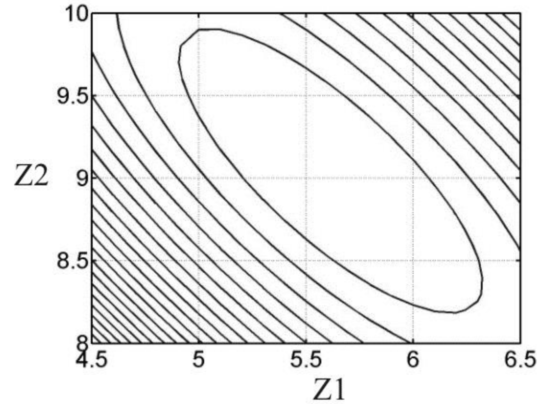
Table 7: Polynomial coefficients for the BACKLOG$^{1/2}$ response surface

| $\beta_0$ | $\beta_1$ | $\beta_2$ | $\beta_3$ | $\beta_4$ | $\beta_{11}$ | $\beta_{12}$ | $\beta_{13}$ | $\beta_{14}$ |
|---|---|---|---|---|---|---|---|---|
| 2.307 | 1.5151 | -0.1052 | 0.09815 | -0.08816 | -0.04093 | 2.827E-3 | 1.503E-3 | 6.053E-4 |

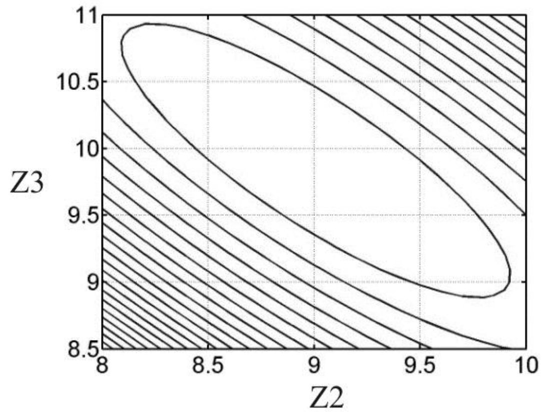
| $\beta_{22}$ | $\beta_{23}$ | $\beta_{24}$ | $\beta_{33}$ | $\beta_{34}$ |
|---|---|---|---|---|
| 1.600E-3 | 1.781E-3 | 7.884E-4 | 1.105E-3 | 1.071E-3 |




a) Contour plot of cost vs Z1 and Z4
(Z2=9.2, Z3=9.6)




b) Contour plot of cost vs Z1 and Z2
(Z3=9.6, Z4=20.8)




c) Contour plot of cost vs Z2 and Z3
(Z1=5.4, Z4=20.8)

Figure 7 (a), (b) and (c): Contour plots of the cost function surface

Figures 7 (a), (b), and (c) show different contour plots of the resulting cost function.

The results of the optimization for the six cases using the combined discrete/continuous model are presented in Table 8. We do not provide the full analysis for all cases as it would take up too much space and not add any further value to the analysis.

Table 8: Parameters and optimization results

| CASE | $MTTF$ | $MTTR$ | $m$ | $Z_1^*$ | $Z_2^*$ | $Z_3^*$ | $Z_4^*$ | $Z_5^*$ | $Z_6^*$ | $\tilde{C}^*$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 100 | 3 | 4 | 5.4 | 9.2 | 9.6 | 20.8 | -- | -- | 44.1 |
| 2 | 700 | 21 | 4 | 37.5 | 64.2 | 67.1 | 145.6 | -- | -- | 308.9 |
| 3 | 1500 | 45 | 4 | 80.3 | 137.6 | 143.9 | 312.0 | -- | -- | 661.9 |
| 4 | 100 | 3 | 6 | 4.5 | 6.7 | 9.1 | 9.7 | 10.4 | 24.1 | 57.5 |
| 5 | 700 | 21 | 6 | 31.6 | 46.8 | 63.6 | 67.8 | 72.9 | 168.8 | 402.7 |
| 6 | 1500 | 45 | 6 | 67.8 | 100.4 | 136.2 | 145.4 | 156.3 | 361.8 | 862.9 |

We notice that the optimal values of the parameters increase along the transfer line. We believe this is because of the large penalty imposed on backlog. This penalty justifies an important finished goods buffer to prevent backlogging. We also notice that the first buffer is quite smaller than the other ones. This can be justified by the fact that the first machine is never starved. Therefore, machine $M_1$ only needs to build up an inventory to decouple machine $M_2$ from propagation of its failures, but not starvation. The intermediate buffers have buffer levels that are more similar while they increase slightly as we move downstream.

## CONCLUSION

In this paper, we have studied the production rate control problem for a tandem manufacturing system with machines subject to random failures. We have shown that the complexity of the analytical problem prohibits us from solving its related HJB equations. Using results in simpler systems, a structure for a sub-optimal control policy or heuristic policy is given. To obtain the optimal values for the input parameters of the heuristic policy, we have used a combination of simulation, DOE and RSM and provide some numerical examples. These results were obtained within an hour for the four machine lines and 2 for the 6 machine lines. We believe this technique enables the treatment of longer lines, problems which would be otherwise be very difficult to tackle analytically or numerically. In order to reduce the computational time of the discrete event simulation model, a combined discrete/continuous model has been proposed which divides the computational times by hundreds in the given numerical examples. Even as computers will evolve, the ratio of the discrete model computational time to the combined discrete/continuous model will remain in the same order of magnitude. We also believe that this will enable the optimization of long lines in feasible time frames. Future work will include searching for heuristics to reduce the computational effort for optimizing long lines and using the proposed modelling and optimization technique to compare different production control mechanisms such as kanban, CONWIP and Hybrid on a wide set of cases. Both of these subjects will necessitate a great number of experiments and thus will benefit largely from the reduction in computational time.

REFERENCES


Akella, R., & Kumar, P. R. (1986). Optimal Control of Production Rate in a Failure Prone Mnufacturing System. *IEEE Transactions on Automatic Control, AC-31*(2), 116-126.

Bai, S. X., & Gershwin, S. B. (1994). Scheduling manufacturing systems with work-in-process inventory control: Multiple-part-type systems. *International Journal of Production Research, 32*(2), 365-385.

Banks, J. (1998). *Handbook of simulation principles, methodology, advances, applications, and practice*. New York, N.Y.: J. Wiley and Sons.

Bonvik, A. M. (1996). *Performance analysis of manufacturing systems under hybrid control policies.* Unpublished Thesis Ph. D. --Massachusetts Institute of Technology Sloan School of Management 1996.

Bonvik, A. M., Dallery, Y., & Gershwin, S. B. (2000). Approximate analysis of production systems operated by a CONWIP/finite buffer hybrid control policy. *International Journal of Production Research, 38*(13), 2845-2869.

Burman, M. H. (1995). *New results in flow line analysis.* Unpublished Thesis Ph. D. --Massachusetts Institute of Technology Dept. of Electrical Engineering and Computer Science 1995.

Chiang, S. Y., Kuo, C. T., & Meerkov, S. M. (1999). *c-Bottlenecks in Serial Production Lines: Identification and application.* (No. CGR-99-05). Ann Arbor, Michigan: University of Michigan, Department of Electrical Engineering and Computer Science.

Dallery, Y., David, R., & Xie, X. L. (1989). Approximate Analysis of Transfer Lines with Unreliable Machines and Finite Buffers. *IEEE Transactions on Automatic Control, 34*(9), 943-953.

Dallery, Y., & Le Bihan, H. (1999). An improved decomposition method for the analysis of production lines with unreliable machines and finite buffers. *International Journal of Production Research, 37*(5), 1093-1117.

Gershwin, S. B., & Schick, I. C. (1979). Analytic Methods for Calculating Performance Measures of Production Lines with Buffer Storages. *Proc IEEE Conf Decis Control Incl Symp Adapt Processes 17th, Jan 10-12 1979*, 618-624.

Gershwin, S. B., & Schor, J. E. (2000). Efficient algorithms for buffer space allocation. *Annals of Operations Research, Performance Evaluation and Optimization of Production Lines. International Workshop, 19-22 May 1997, 93*, 117-144.

Gharbi, A., & Kenné, J. P. (2003). Optimal production control problem in stochastic multiple-product multiple-machine manufacturing systems. *IIE Transactions, 35*(10), 941-952.

Kenné, J. P., & Boukas, E. K. (1997, June 1997). *Production Rate Control and Corrective Maintenance Planning Problem of a Failure Prone  Manufacturing System.* Paper presented at the ACC conference, Albuquerque.

Kenné, J. P., & Gharbi, A. (1999). Experimental design in production and maintenance control problem of a single machine, single product manufacturing system. *International Journal of Production Research, 37*(3), 621-637.

Kushner, H. J., & Dupuis, P. G. (1992). *Numerical Methods for Stochastic Control Problems in Continuous Time*. New-York: Springer-Verlag.

Lou, S., Sethi, S. P., & Zhang, Q. (1994). Optimal feedback production planning in a stochastic two-machine flowshop. *European Journal of Operational Research, 73*(2), 331-345.

Montgomery, D. C. (2005). *Design and analysis of experiments* (6th ed.). New York: Wiley.

Presman, E., Sethi, S., & Zhang, Q. (1995). Optimal Feedback Production Planning in a Stochastic N-machine Flowshop. *Automatica, 31*(9).

Pritsker, A. A. B., & O'Reilly, J. J. (1999). *Simulation with Visual SLAM and AweSim* (2nd ed.). New York, N.Y.,  West Lafayette, Ind.: J. Wiley and Sons Systems Publishing.

Sadr, J., & Malhame, R. P. (2004). Decomposition/aggregation-based dynamic programming optimization of partially homogeneous unreliable transfer lines. *IEEE Transactions on Automatic Control, 49*(1), 68-81.

Schor, J. E. (1995). *Efficient algorithms for buffer allocation.* Unpublished Thesis M.S. -- Massachusetts Institute of Technology Dept. of Electrical Engineering and Computer Science 1995.

Venkataraman, P. (2002). *Applied optimization with MATLAB programming*. New York, N.Y.: J. Wiley and Sons.