Information Sciences 127 (2000) 69–83
INFORMATION SCIENCES
AN INTERNATIONAL JOURNAL
www.elsevier.com/locate/ins

Fuzzy work-in-process inventory control of unreliable manufacturing systems

N.C. Tsourveloudis *, E. Dretoulakis, S. Ioannidis

Department of Production Engineering and Management, Technical University of Crete, University Campus, 73100 Chania, Greece

Abstract

We consider single and multiple part type production lines and networks with finite buffers and unreliable machines. Three fuzzy control modules, namely, *line*, *assembly*, and *disassembly* controller, are developed. The objective is to keep the work-in-process (WIP) inventory and cycle time at low levels, along with high machine utilization and throughput. This is achieved by adjusting the processing rate of each production stage so that workflow is balanced and the extreme events of machine starving or blocking are reduced. The approach is extensively tested via simulation. After a series of simulation runs, it has been observed that the proposed approach outranks other control policies in keeping the WIP inventory low. © 2000 Elsevier Science Inc. All rights reserved.

Keywords: Production lines; Production networks; Work-in-process; Fuzzy control

1. Introduction

The continuing changes in the production environment create new challenges, which companies have to face in order to stay in business. One of the essential surviving requirements is the ability of a firm to adapt its production structure according to the fast changing global market needs. Modern production technologies such as, for example, *flexible/agile* and *just-in-time* manufacturing, recognize that speedy and punctual response to market changes is associated with short cycle times and low in-process inventories. As a result,

* Corresponding author. Fax: +30-821-69410.
E-mail addresses: nikost@dpem.tuc.gr (N.C. Tsourveloudis), manos@dpem.tuc.gr (E. Dretoulakis), dervish@dpem.tuc.gr (S. Ioannidis).

0020-0255/00/$ - see front matter
PII: S 0 0 2 0 - 0 2 5 5 (0 0) 0 0 0 3 0 - X

control policies that tend to keep *work-in-process* (WIP) in low levels have drawn a great deal of attention from researchers and practitioners. WIP inventory is measured by the number of items in each buffer and should stay small because of various reasons [5,9]:

- Capital invested to inventories as long as they remain in the factory or the warehouse provides no profit.
- High in-process inventories increase cycle times and decreases responsiveness to customers.
- High in-process inventories require more space and expensive material handling equipment increasing the invested capital.
- Inventory quality decreases as the unfinished items remain to the factory because they are vulnerable to damage.

Many researchers have studied the problem of WIP management in unreliable production systems [1–3,5,9,12,13]. Conway et al. [5] examined the effects of in-process inventory in production lines. Bai and Gershwin introduced a WIP control algorithm for scheduling single [4] and multiple part-type [3] production lines. In almost all works, WIP inventories are associated with built-in parameters of the system, as for example, the number of wokstations/machines, their processing rate, the interstation buffer capacity and assumptions concerning failure and repair rate of each machine. It is also a common belief that the large size of real production systems along with the effects of failures occurring in such systems, do not allow for an analytic treatment of WIP minimization [9]. Since analytical solutions are not attainable, heuristic policies are suggested to control job flow within production systems [2–4] sometimes supported by fuzzy set theory [6,7].

In this paper, we develop a distributed fuzzy control methodology for single and multiple part-type production lines and networks. The overall control objective is to keep the WIP and cycle time as low as possible, and at the same time to maintain high machine utilization and throughput. In contrast to the traditional produce-at-capacity approach, according to which the system always operates at its maximum capacity, we control the production rate in each production stage in a way that eliminates extreme events of idle periods due to machine starving or blocking. The next section describes the three production modules and the proposed architecture of the distributed fuzzy logic control system that is used. There, we also present a fuzzy logic formulation of the WIP control problem. In Section 3, simulation results are drawn along with comparisons, and in Section 4, the contribution of this work is summarized and further work is outlined.

2. The production control modules

A production system is usually viewed as a network of machines/workstations and buffers. Items receive an operation at each machine and wait for the next operation in a buffer with finite capacity. Random machine breakdowns disturb the production process and phenomena such as starvation and/or blocking may occur. Due to a failed machine with operational neighbors, the level of the downstream buffer decreases, while the upstream increases. If the repair time is big enough, then the broken machine will either block the next station or starve the previous one. This adverse effect will propagate throughout the system.

The events that can happen in a production network are changes in buffer states and changes in machine states. The buffers can be full or empty and the machines can be up (operating) or down (under repair). When a machine is up, it can be starved if one of the preceding buffers is empty; in this case the machine is forced to produce at the rate of the machine feeding the empty buffer. Respectively, if a machine is up then it can be blocked if one of the succeeding buffers is full. When a machine breaks down then the preceding machines remain operating until one of their downstream buffer is filled. Similarly, the succeeding machines continue processing until their upstream buffers become empty.

The vast majority of production systems can be decomposed into basic modules or subsystems. Here, we introduce three control modules for transfer *line*, *assembly* and disassembly networks, respectively. These modules are schematically presented in Fig. 1.

The transfer line module includes a machine M_i which takes unfinished items from an upstream buffer $B_{j,i}$ and after processing, sends them to a downstream buffer $B_{i,l}$ (Fig. 1(a)). The assembly operation is presented in Fig. 1(b). A machine M_i obtains two or more parts or subassemblies, following an assembly factor $\delta_{j,i}$ from more than one upstream buffers $B_{j,i}$, brings them together to form a single unit, which is sent to a downstream buffer $B_{i,l}$. The disassembly operation involves a machine M_i taking unfinished single units from one upstream buffer $B_{j,i}$, separates them to two or more parts or subassemblies following a disassembly factor $d_{i,j}$, and sends them to downstream buffers $B_{i,k}$, as shown in Fig. 1(c). The main advantage, as mentioned earlier, is that if these subsystems get connected to each other, can be used for modeling and control of manufacturing networks of random geometry.

2.1. Fuzzy control representation

Each of the subsystems presented in the previous section can be seen as the fuzzy controller presented in Fig. 2. The input variables of each controller are:

- the *buffer level* b_{ij} and b_{ik} of the upstream and downstream buffers,
- the *state* ms_i of machine M_i,
- the *production surplus* x_i of M_i, which is the difference between actual production and demand.

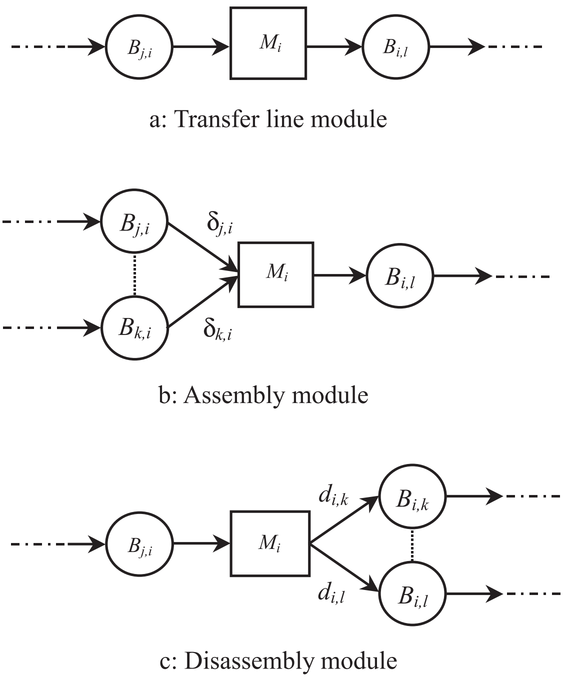


Fig. 1. Production control modules.

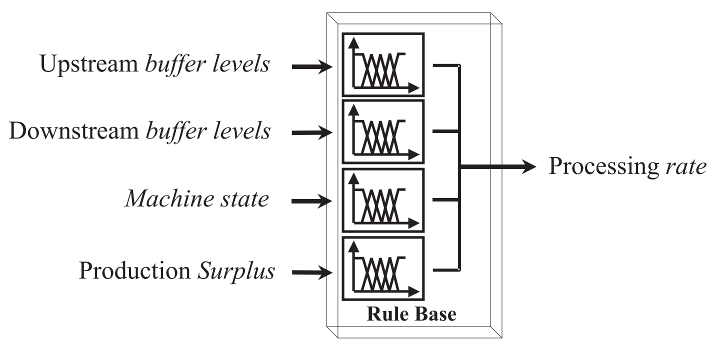


Fig. 2. Inputs and output of the fuzzy controller.

The output variable of every controller is the *processing rate* r_i of each machine M_i.

The buffer levels, surplus and the processing rate of each machine take linguistic variations with certain membership functions. The machine state ms_i is crisp and can be 1 (up) or 0 (down).

The control objective in all cases is to meet the demand and the same time to keep WIP as low as possible. This is achieved by regulating the processing rate at every time instant, according to the following general rules:

1. If there is no sign of machine starving or blocking, then keep the production surplus close to zero. In other words, produce at a rate more or less equal to demand.
2. If an undesirable event (upstream or downstream buffer full or empty) is about to occur, then ignore surplus levels and try to prevent starving or blocking by increasing or decreasing the production rate accordingly.

A buffer tends to be empty when the upstream machine is either under repair or producing in a slower rate than the downstream machine. Similarly, a buffer tends to fill when the downstream machine is either under repair or producing in a slower rate than the upstream machine. The controllers keep buffers neither full nor empty regulating the machine rates. When a buffer tends to be full, the controller is increasing the rate of the downstream machine and decreasing the rate of the upstream machine. In the same way, when a buffer tends to be empty, the controller is increasing the rate of the upstream machine and decreasing the rate of the downstream machine. The information needed to synchronize the operation of the production network is transferred to each module controller by the change of buffer levels. Every event occurring in the production network is affecting the levels of buffers close to the area of the event. In this way, the production system operates in satisfactory rate while the WIP is kept in low levels.

In fuzzy controllers, the control policy is described by linguistic IF–THEN rules with appropriate mathematical meaning [8]. The rule base of the *line control module* contains rules of the following form:

$$\begin{aligned}&\text{IF } b_{j,i} \text{ is } LB^{(k)} \text{ AND } b_{i,l} \text{ is } LB^{(k)} \text{ AND } ms_i \text{ is } LMS_i^{(k)} \text{ AND } x_i \text{ is } LX^{(k)},\\&\quad \text{THEN } r_i \text{ is } LR_i^{(k)},\end{aligned} \tag{1}$$

where k is the rule number ($k = 1, \ldots, 18$), i the number of machine or workstation, LB a linguistic value of the variable *buffer level b* with term set $B = \{Empty,\ Almost\ Empty,\ OK,\ Almost\ Full,\ Full\}$, ms_i denotes the state of machine i, which can be either 1 (operative) or 0 (stopped) and consequently $MS = \{0, 1\}$. LX represents the value that surplus x takes, and it is chosen from the term set $X = \{Negative,\ OK,\ Positive\}$. The *production rate r* takes linguistic values LR from the term set $R = \{zero,\ Low,\ Normal,\ High\}$. The mathematical meaning of the kth rule, for $LMS_i^{(k)} = 1$, can be given as a fuzzy relation $FR^{(k)}$ on $B \times X \times R$, which in the membership function domain is

$$\mu_{FR^{(k)}}(b_{j,i}, b_{i,l}, x_i, r_i) = f_{\to}[\mu_{LB^{(k)}}(b_{j,i}), \mu_{LB^{(k)}}(b_{i,l}), \mu_{LX^{(k)}}(x_i), \mu_{LR^{(k)}}(r_i)], \tag{2}$$

where $f_{\rightarrow} = \min$ for rules of the Mamdani type [8]. Obviously, whenever $LMS_i^{(k)} = 0$ the production rate r takes the *Zero* value from the R term set.

Let us now assume that the machine is not stopped, and the actual buffer levels of the upstream and downstream buffers can be represented as $b^*_{j,i}$ and $b^*_{i,l}$ with membership functions $\mu^*_B(b_{j,i})$ and $\mu^*_B(b_{i,l})$, respectively. The production surplus at a given time instant is denoted as x^*_i with membership $\mu^*_{X(x_i)}$. The membership function of the conjunction of the three inputs, for AND = min, is

$$\mu^*_{\mathrm{AND}}(b_{j,i}, b_{i,l}, x_i) = \mu^*_B(b_{j,i}) \wedge \mu^*_B(b_{i,l}) \wedge \mu^*_X(x_i). \tag{3}$$

The production rate r^*_i, e.g., the control action at every time instant is given by

$$r^*_i = \frac{\sum r_i \mu^*_R(r_i)}{\sum \mu^*_R(r_i)}, \tag{4}$$

where $\mu^*_R(r_i)$ is the membership function of the aggregated production rate, which is computed by applying the max–min composition on the outcome of (2) and (3). That is,

$$\mu^*_R(r_i) = \max_{b_{j,i}, b_{i,l}, x_i} \min[\mu^*_{\mathrm{AND}}(b_{j,i}, b_{i,l}, x_i), \mu_{FR^{(k)}}(b_{j,i}, b_{i,l}, x_i, r_i)]. \tag{5}$$

Similarly, the generic rule of the *assembly* and *disassembly control modules* can be written as follows:

$$\text{IF } b_{j,i} \text{ is } LB^{(k)} \text{ AND} \ldots \text{AND } b_{i,l} \text{ is } LB^{(k)} \text{ AND } ms_i \text{ is } LMS_i^{(k)} \text{ AND } x_i \text{ is } LX^{(k)}, \quad \text{THEN } r_i \text{ is } LR_i^{(k)}. \tag{6}$$

The formulation presented in this section can be expanded to multiple-part-type production lines or networks without major modifications. The structure of the fuzzy controller remains the same since a multiple-part-type system can be decomposed into single part type systems.

3. Simulation results and comparative testing

In this section, we test the proposed control approach and compare its performance to other well-known control approaches. We assume that the flow of parts within the system is continuous. In the continuous-flow simulation, the discrete production is approximated by the production of a liquid product [11]. The assumptions we made for all simulations are as follows:

1. Machines fail randomly with a probability $p_{i,}$, which is given by

$$p_i = \frac{r_i}{c_0}, \quad i = 1, \ldots, N, \tag{7}$$

where r_i is the processing rate of machine M_i and c_0 is a constant. Machines that work at their maximum have higher failure probability.
2. Machines are repaired randomly with probability pr_i. We assume unlimited repair personnel. There is always somebody to start working on a failed machine.
3. Time to failure and time to repair are geometrically distributed.
4. All machines operate at known, but not necessarily equal, rates. Each machine produces in a rate $r_i \leqslant \mu_i$, where μ_i is the maximum processing rate of machine M_i.
5. The initial buffers (B_I) are infinite sources of raw material and consequently the initial machines are never starved.
6. The last buffer (B_F) has infinite storage capacity, so the last machine is never blocked.
7. Buffers between adjacent machines M_i, M_j have finite capacities $BC_{ij},\ i,j = 1,\ldots,N$.
8. Set-up times or transportation times are negligible or are included in the processing time.

Matlab's Fuzzy Logic Toolbox [10] and *Simulink* were the software tools for building and testing all simulations. The performance of the fuzzy WIP control approach is evaluated through the following test cases.

3.1. Test case 1: single-part-type transfer lines

The developed fuzzy controller is first tested for the case of a single product transfer line presented in Fig. 3. The production line consists of five machines and four interstation buffers. The first buffer, denoted as B_I, is an infinite source while the last buffer B_F has infinite storage capacity. The system is balanced. All machines have the same processing time, $\tau_i = 0.5$ ($i = 1,\ldots,5$), and same failure and repair probabilities, $p_i = 0.1$ and $pr_i = 0.5$, respectively.

The transfer line of Fig. 3 is identical to one presented by Bai and Gershwin (Test case 5 in [4]), and it was selected to facilitate comparisons. The performance of our controller is compared to:
1. The classical produce-at-capacity approach, according to which the machines produce in their maximum rate when they are operational (up, not blocked, not starved). This is similar to what is known as bang–bang control.
2. The approach was presented by Bai and Gershwin in [2,4] and elsewhere. Their method is based on the determination of a desirable production surplus value, the *hedging point*.

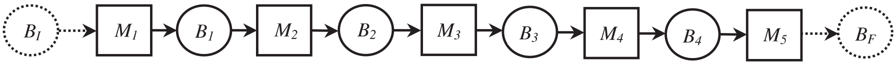


Fig. 3. The transfer line of Test case 1.

When the machine is up, the control law used by Bai and Gershwin is summarized in the following:

- If the actual surplus is less than the hedging point, then the machine should produce at its maximum rate.
- If surplus is equal to the hedging point, then the production rate should be equal to demand.
- If surplus is greater than the hedging point, then stop producing.

The proposed single-part-type transfer line controller contains 18 fuzzy IF–THEN rules. Their structure was discussed in a previous section. A part of the actual rulebase is presented in Table 1. Main control objective is to maximize the utilization of machines by avoiding starvation and/or blocking. Simultaneously, production surplus should be close to zero in order to satisfy the demand.

The control methods are examined for five different demand values. For each demand, various simulation runs were performed with different random number seeds. All results are averaged over the number of simulation experiments. Buffer capacities are given and presented in Table 2. Fig. 4 represents graphically the WIP inventory of each method. It can be seen that the proposed approach keeps the in-process inventories significantly lower than the other two methods for all demands. Numerical results are presented in Table 3.

3.2. Test case 2: multiple-part-type transfer lines

In this section, we demonstrate the performance of the transfer line controller we suggest, for the case of multiple-part-type production lines. The production system under consideration consists of three machines and produces three product types. Buffers with finite storage capacity are located

Table 1
Part of controller's rulebase[a]

Rule	IF	LB_i	AND	LB_{i+1}	AND	MS_i	AND	X_i	THEN	R_i
1		OK		Almost full		1		OK		Low
2		Any		Full		1		Any		Zero
3		Not empty		Not full		1		Negative		High
4		OK		OK		1		OK		Normal
5		Full		Almost empty		1		OK		High
6		Any		Any		0		Any		Zero
7		Not empty		Empty		1		Any		High

[a] LB_i: level of buffer i, LB_{i+1}: level of buffer $i+1$, MS_i: state of machine i, X_i: surplus of machine i, R_i: production rate of machine i.

Table 2
Buffer sizes and demand levels for Test case 1

Demand d	Buffer capacity			
	BC_1	BC_2	BC_3	BC_4
1.6	6	6	8	5
1.4	3	3	10	1
1.2	1	4	5	1
1.0	1	1	1	1
0.6	1	1	1	1

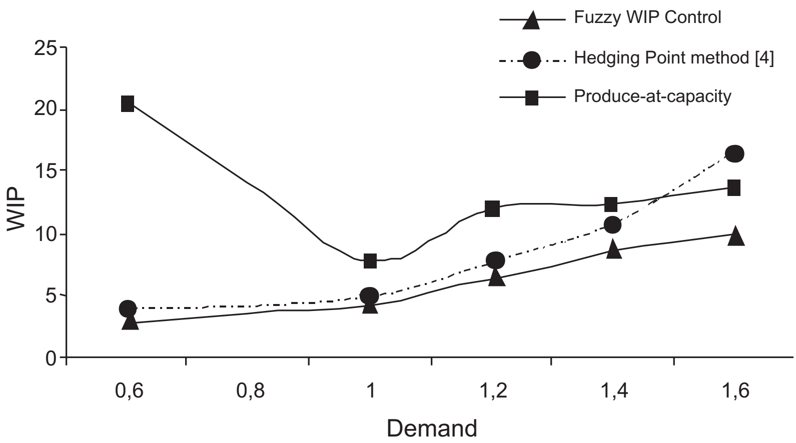


Fig. 4. In-process inventory for Test case 1.

Table 3
Average buffer level and WIP inventory for Test case 1

Demand	Average buffer levels for fuzzy WIP control				WIP inventory		
	B_1	B_2	B_3	B_4	Fuzzy WIP control	Hedging point method [4]	Produce at capacity
1.6	0.97	1.42	2.77	0.77	9.93	16.5	13.67
1.4	0.74	0.67	3.22	0.35	8.48	10.7	12.36
1.2	0.78	1.36	1.02	0.21	6.37	7.5	11.96
1.0	0.83	0.43	0.26	0.21	4.23	5	7.71
0.6	0.52	0.33	0.19	0.26	2.8	3.9	20.6

between machines. The first buffer for each product is assumed to be an infinite source while the last is an infinite sink. All machines are subject to random failures and repairs with known rates.

System data and structure are identical to an example presented in [3] by Bai and Gershwin. Each machine is "virtually" divided into as many sub-machines (or *partial machines* [3]) as the number of part types to be processed. Consequently, the three-part-type system under study, illustrated in Fig. 5, can be approximated by three single-type systems similar to one analyzed in the previous section. Partial machines are presented in Fig. 5 with dotted line squares. Each machine i performs operations on parts of j type and it is divided into m_{ij} partial machines (here $i = j = 1, \ldots, 3$). Each m_{ij} does one operation on the type j part, which then waits in the b_{ij} buffer for an operation at the $m_{i+1,j}$ partial machine. The demand for parts of type j is d_j. The failure and repair rate of machine i is p_i and r_i, respectively. The processing times τ_{ij} are chosen as follows [3]:

$$\begin{aligned} &\tau_{1,1} = 0.5, \quad \tau_{2,1} = 0.3, \quad \tau_{3,1} = 0.4, \\ &\tau_{1,2} = 0.3, \quad \tau_{2,2} = 0.2, \quad \tau_{3,2} = 0.3, \\ &\tau_{1,3} = 0.4, \quad \tau_{2,3} = 0.4, \quad \tau_{3,3} = 0.5. \end{aligned}$$

Failure rates are $p_1 = 0.1$, $p_2 = 0.01$ and $p_3 = 0.2$, and repair rates are $r_1 = 0.5$, $r_2 = 0.8$, $r_3 = 0.6$. Buffer sizes are all equal to 1 and demand is assumed to be constant over time for each part type ($d_1 = 0.8$, $d_2 = 0.6$, $d_3 = 0.3$). The WIP was calculated for each product after multiple simulation runs using different seeds. These results are shown below

$$\mathrm{WIP}_1 = 1.369, \quad \mathrm{WIP}_2 = 0.988, \quad \mathrm{WIP}_3 = 0.667.$$

The fuzzy WIP controller satisfied the demand (which is selected low anyway) and kept the total WIP 20% less than in [3].

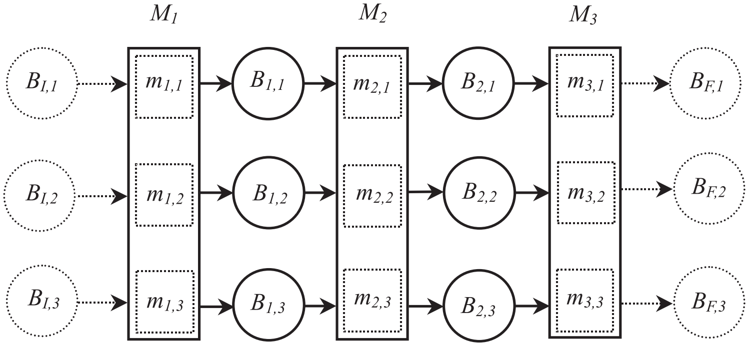


Fig. 5. Test case 2: 3-machine, 3-part-type production line.

3.3. Test case 3: single-part-type production networks

The fuzzy WIP controller was tested for production lines in Test cases 1 and 2. In this section, we demonstrate the performance of our approach in assembly/disassembly networks. We consider the system presented in Fig. 6. The network under consideration can be analyzed into two lines, one disassembly and two assembly subsystems. For each type of the subsystems we use the corresponding control module, described in Section 2. The subsystems are connected to each other through common buffers. All machines have the same maximum production rate, which is $\mu_i = 20$ parts per time unit. All buffer capacities are equal to $BC_{ij} = 50$ parts, apart from B_{I} and B_{F}. The failure probability of all machines is given by

$$p_i = \frac{r_i}{100}. \tag{8}$$

The machine repair probability is $pr_i = 0.4$. The assembly and disassembly factors are equal to one, $\delta_{j,i} = d_{i,j} = 1$. The buffer levels at any time instant is given by

$$b_{j,i}(t_{k+1}) = b_{j,i}(t_k) + [r_j(t_k) - r_i(t_k)](t_{k+1} - t_k), \tag{9}$$

where $t_k,\ t_{k+1}$ are the times when control actions (changes in processing rates), happen. The production of a machine M_i is

$$pr_i(t_{k+1}) = pr_i(t_k) + r_i(t_k)(t_{k+1} - t_k). \tag{10}$$

The mean machine rate mr_i is given by

$$mr_i = \frac{pr_i(T)}{T}, \tag{11}$$

where T is the total simulation time.

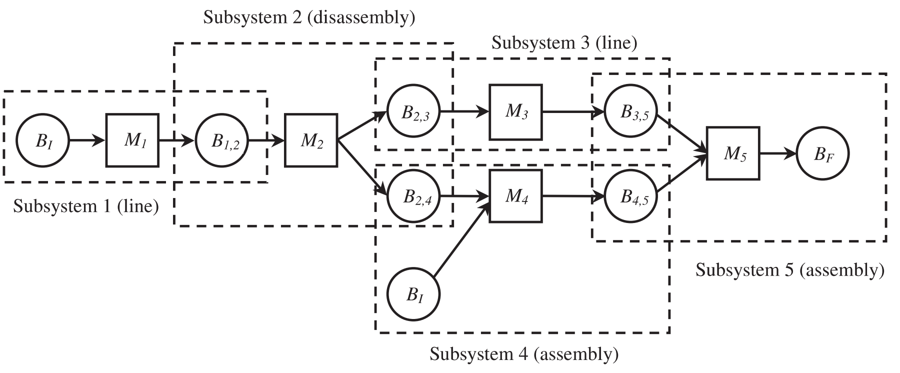
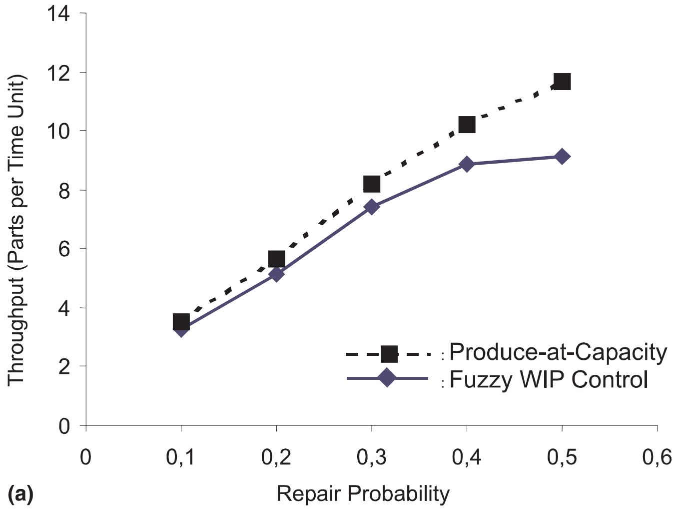


Fig. 6. The production network of Test case 3: (a) throughput versus repair probability; (b) total WIP versus buffer capacity; (c) cycle time versus processing rates; (d) system utilization.



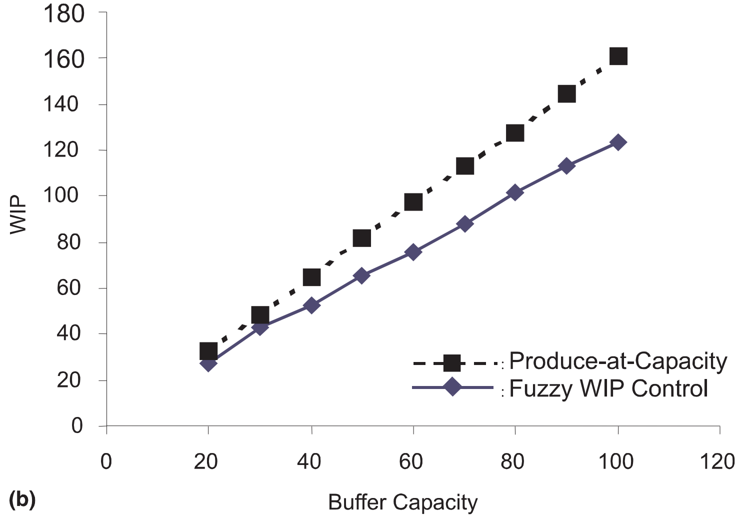
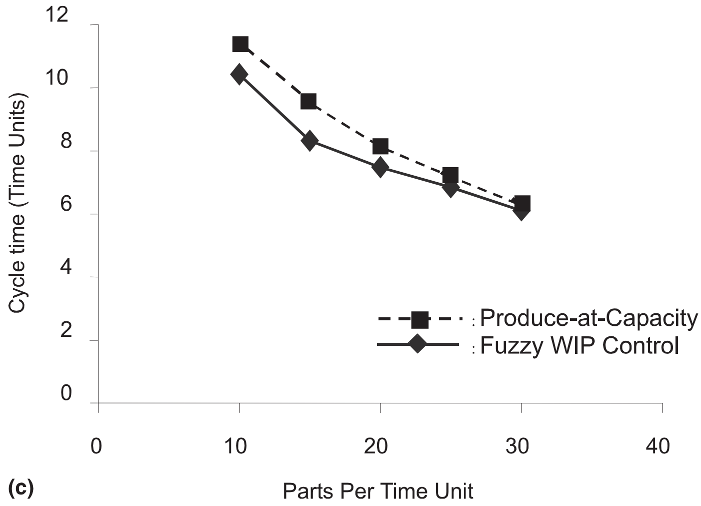


Fig. 6. (*Continued*).

The proposed approach is tested against the produce-at-capacity policy. Two continuous flow simulation models were implemented, one for each tested policy. Comparative results for the WIP, cycle time and throughput, are shown in Fig. 6(a)–(c). Percentages of idle and operating times for both methods are presented in Fig. 6(d). It can be seen that for the network of Test case 3, the proposed distributed fuzzy WIP control system reduces substantially the



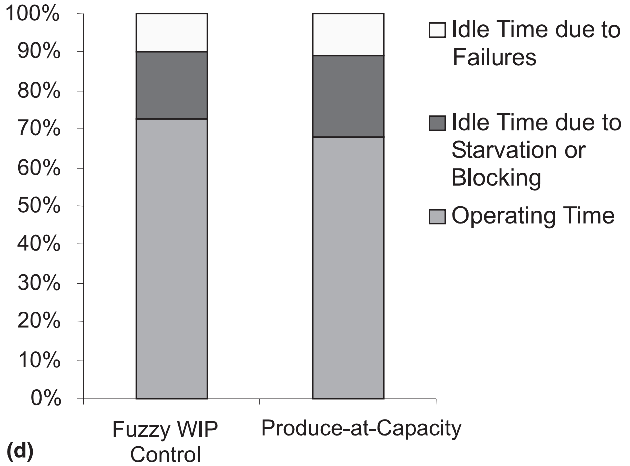


Fig. 6. (*Continued*).

in-process inventory and cycle time, while the throughput reduction compared to produce-at-capacity policy is less than 10%.

3.4. Remarks

From the simulation results in all previous test cases it can be concluded that the fuzzy WIP controller (for lines and/or networks) keeps lower in-process inventories compared to the other approaches examined, that is, produce-at-capacity and hedging point [3,4,9] methods. The main advantage of the fuzzy

WIP controller is that it approximates the way human operators adjust the processing rate so as to minimize idle periods due to starving or blocking. Two possible drawbacks can be identified. The first concerns the ease of the WIP controller implementation. Real-time regulation of processing rate requires on-line monitoring of buffer levels and production surplus. This might be unrealistic in practice. The second remark is associated with the decision space complexity of the fuzzy WIP controller. The produce-at-capacity policy follows just one control rule: *IF machine is not down*, *THEN produce at the maximum rate*. Despite the fact that the bang–bang behavior of this policy is not appropriate for regulation problems, it has gained wide acceptance in production control practice because of its simplicity (and high throughput). Similarly, the hedging point method uses only three control rules to adjust machine's processing rate, in contrast to the rulebase of the fuzzy WIP controller for transfer lines (either single or multiple part types) which contains 18 rules. It should be noted that although it might be a problem in systems of larger size, we have not observed any significant delay in computations that can be attributed to the number of rules used in our controllers.

4. Conclusions

We have presented a distributed fuzzy controller, which keeps production close to demand and WIP inventories in low levels by regulating the processing rate of each machine. The proposed control system consists of three independent modules and can be applied to production networks of general topology. The structural advantage of the approach presented here is that it allows for an operator-like knowledge representation and reasoning. Numerous simulation experiments verified controller's good performance. A continuous-flow simulator is used to compare the proposed WIP controller with produce-at-capacity and hedging point methods. For the test cases examined, it turned out that the fuzzy policy provides lower WIP, higher system utilization, and smaller product cycle time.

An interesting extension, to be considered in the future, would consist of examining the performance of WIP controller when applied to reentrant systems, in which parts may visit some machines more than once. More complex patterns of demand should be considered.

Acknowledgements

This work was partly supported by the Greek Ministry of Industry, Energy, and Technology under Grant ΥΠΕΡ 95/132.

References

[1] R. Akella, P.R. Kumar, Optimal control of production rate in a failure prone manufacturing system, IEEE Transactions on Automatic Control 31 (2) (1986) 116–126.

[2] S.X. Bai, S.B. Gershwin, Scheduling manufacturing systems with work-in-process inventory control: formulation of single part type systems, in: Proceedings of the 29th IEEE Conference on Decision and Control, Hawaii, December 1990.

[3] S.X. Bai, S.B. Gershwin, Scheduling manufacturing systems with work-in-process inventory control: multiple-part-type systems, International Journal of Production Research 32 (2) (1994) 365–386.

[4] S.X. Bai, S.B. Gershwin, Scheduling manufacturing systems with work-in-process inventory control: single-part-type systems, IIE Transactions 27 (1995) 599–617.

[5] R. Conway, W. Maxwell, J.O. McClain, L. Joseph Thomas, The role of work-in-process inventory in serial production lines, Operations Research 36 (2) (1988) 229–241.

[6] L. Custodio, J. Sentieiro, C. Bispo, Production planning and scheduling using a fuzzy decision system, IEEE Transactions on Robotics and Automation 10 (2) (1994) 160–168.

[7] E. Dretoulakis, S. Ioannidis, N.C. Tsourveloudis, Distributed intelligent control of unreliable manufacturing systems, in: Proceedings of the Third IMACS/IEEE International Multiconference on Circuits, Systems, Communication and Computers, Athens, Greece, July 1999.

[8] D. Driankov, H. Hellendoorn, M. Reinfrank, An Introduction to Fuzzy Control, Springer, New York, 1993.

[9] S.B. Gershwin, Manufacturing Systems Engineering, Prentice-Hall, NJ, 1994.

[10] N. Gulley, J.-S.R. Jang, Fuzzy Logic Toolbox User's Guide, The MathWorks, 1995.

[11] V.S. Kouikoglou, Y.A. Phillis, A continuous-flow model for production networks with finite buffers unreliable machines and multiple products, International Journal of Production Research 35 (2) (1997) 381–397.

[12] A. Sharifnia, Production control of a manufacturing system with multiple machine states, IEEE Transactions on Automatic Control 33 (7) (1988) 621–625.

[13] H.P. Wiendahl, Load-oriented Manufacturing Control, Springer, Hannover, 1995.