Evaluation of Printed Circuit Board Assembly Manufacturing Systems Using Fuzzy Colored Petri Nets

Simon C.K. Shiu*, Eric C.C. Tsang, Daniel S. Yeung and Martin B. Lam

Department of Computing, Hong Kong Polytechnic University
*Email: csckshiu@comp.polyu.edu.hk
This project is supported by a PolyU grant No. 351/689

ABSTRACT

In this paper, implementation and evaluation of a Printed Circuit Board Assembly (PCBA) manufacturing system model based on papers of Fuzzy Colored Petri Nets (FCPN) modeling technique was performed. From the Petri nets simulation results, two contributions were claimed: (1) The successful application of our approach to evaluate systems that consist of complicated concurrent processes with embedded data structures and uncertainty reasoning. (2) An approach in automatic determination of the threshold values in the fuzzy production rules using FCPN. The details of the assembling processes and various manufacturing data were gathered from computer manufacturers located in China. Different resources allocation strategies were introduced in the experiments, and simulation results were obtained. The resources included screen printers, different pick and place machines, infrared soldering machines and components insertion robots. All the experiments were built by using the Petri Net tool DESIGN/CPN [6]. A number of hierarchy pages, color sets, places, arcs and transitions representing the PCPA and fuzzy production rules were constructed in the experiments. Simulation runs of the Petri nets model were carried out. The result shows that our approach could be used to guide decision makers in the design and selection of a suitable PCBA manufacturing strategy. Future extension of our work is described at the end of this paper.

1. INTRODUCTION

Fuzzy Colored Petri Nets (FCPN) model [15,17] has been developed for description and analysis of the dynamic behavior and inexact production inference of Flexible Manufacturing Systems. The basic idea is to integrate the modeling and expressive capability of Colored Petri Nets and Fuzzy Petri Nets, where the former is good at modeling complicated concurrent processes with embedded data structures, while the later is good at modeling systems that involve approximate reasoning and uncertainty knowledge inference. The constructed model adopted in this paper was based on a factory's Surface Mount Technology process of Printed Circuit Board Assembly (PCBA) manufacturing system [2,7], and the idea of incorporating fuzzy production rules in the automated insertion process [14,18].

The manufacturing plant is located in China and its factory size is over 100 thousand square feet. There are over 500 staff working on the production floor. The SMT process includes the screen printer process, transportation operation, components pick and place process, error detection and recovery process, solder paste re-flow process and cleaning process. The fuzzy production rules used are for guiding the component insertion by the robot arm. The uncertainty reasoning was due to the fuzzifying of the x-axis, y-axis, z-axis readings of the angles of the component to be inserted with respect to the PCB board. These fuzzy variables were used as inputs to the fuzzy production rules. The rules' output was defuzzified which gave the x displacement and y displacement values. If both displacements were zero, then the component will be inserted into the PCB and the finished PCB will leave the system. Based on the above scenario, we have developed experiments using the software DESIGN/CPN to model the concurrency, synchronization, and mutual exclusion of manufacturing systems with fuzzy characterized behaviors.

This paper is organized into five main sections. The first session gives the introduction and the next session reviews our definition of FCPN. Modeling of the Printed Circuit Board Assembly (PCBA) Manufacturing System using DESIGN/CPN is given in session three. An approach in automatic determination of the threshold values in the fuzzy production rules using FCPN is described in section four. The results of our experiments and future extension of our work are described in section five.

0-7803-4778-1/98 $10.00

2. DEFINITION OF FCPN

We defined a generalized non-hierarchical FCPN [15,17] as an 18-tuple, **FCPN** = (Σ, P, T, D, A, N, C, G, E, Th, F, W, CF, α, β, γ, φ, θ), where

$\Sigma = \{ \sigma_1, \sigma_2, \ldots, \sigma_l \}$, a finite set of non-empty types, called *color sets*, $l \geq 0$,

$P = \{ P_C, P_F \}$ a finite set of *places*,

$P_C = \{ pc_1, pc_2, \ldots, pc_m \}$, a finite set of places that model the dynamic control behavior of system, called *control places*, $m \geq 0$,

$P_F = \{ pf_1, pf_2, \ldots, pf_n \}$, a finite set of places that model the fuzzy production rules, called *fuzzy places*, $n \geq 0$, $P_C \cap P_F = \varnothing$,

$T = \{ T_C, T_F \}$, a finite set of *transitions*,

$T_C = \{ tc_1, tc_2, \ldots, tc_i \}$, a finite set of transitions that are connected to and from *control places*, called *control transition*, $i \geq 0$,

$T_F = \{ tf_1, tf_2, \ldots, tf_j \}$, a finite set of transitions that are connected to or from fuzzy places, called *fuzzy transition*, $j \geq 0$, $T_C \cap T_F = \varnothing$,

$D = \{ d_1, d_2, \ldots, d_h \}$, a finite set of *propositions*, $| P_F | = | D |$,

$A = \{ a_1, a_2, \ldots, a_k \}$, a finite set of *arcs*, $k \geq 0$, $P \cap T = P \cap A = T \cap A = \varnothing$,

$N : A \rightarrow P \times T \cup T \times P$, a *node function*, it maps each arc into a pair where the first element is the source node and the second is the destination node, the two nodes have to be of different kind,

In: an *input function* that maps each node, x, to the set of nodes that are connected to x by an input arc of x;

Out: an *output function* that maps each node, x, to the set of its nodes that are connected to x by an output arc of x,

$C : (P \cup T) \rightarrow \Sigma_{ss}$, a *color function*, i.e. it maps each place and transition to a super-set of color set,

$G : T \rightarrow$expression, a *guard function*, $\forall t \in T$:[$Type(G(t)) = Boolean \wedge Type(Var(G(t))) \subseteq \Sigma$], where $Type$(Vars) to denote the set of types $\{Type(v) | v \in Vars\}$, Vars is a set of variables, $Var(G(t))$ denotes the variables used in $G(t)$,

$E : A \rightarrow expression$, *an arc expression function*, $\forall a \in A$: $[Type(E(a)) = C(p(a))MS \wedge Type(Var(E(a))) \subseteq \Sigma]$ where $p(a)$ is the place of $N(a)$, MS stands for multi-set,

$Th = \{ \lambda_1, \lambda_2, \ldots, \lambda_s \}$ denotes a set of threshold values,

$F = \{ f_1, f_2, \ldots, f_n \}$ denotes a set of fuzzy sets,

$W = \{ w_1, w_2, \ldots, w_p \}$ denotes a set of weights,

$CF = \{ cf_1, cf_2, \ldots, cf_m \}$ denotes a set of certainty factors,

$\alpha : P_F \rightarrow [0,1]$ is an association function, a mapping from fuzzy transitions to certainty factors,

$\beta : P_F \rightarrow D$ is a bijective mapping between the proposition and the fuzzy place for each node,

$\gamma : P_F \rightarrow Th$ is an association function, a mapping from fuzzy places to threshold values. Each proposition in the antecedent is assigned a threshold value,

$\varphi : P_F \rightarrow F$ is an association function that assigns a fuzzy set to each fuzzy place,

$\theta : P_F \rightarrow W$ is an association function that assigns a weight to each fuzzy place.

3. MODELING OF PCBA MANUFACTURING SYSTEM USING FCPN

PCBA typically consists of a series of processes, which include surface mount process, component insertion, wave soldering, touch-up, add-on, and testing. In Surface Mount section, it involves solder paste printing, components pick and place operation, solder paste re-flow and cleaning processes. In solder paste printing process, solder paste is printed on the board, and the components are assembled on the board through components pick and place machine. After finishing the pick and place operation, the board transports to infrared machine for solder paste re-flow process. Finally, the board is cleaned in cleaning process. The component insertion sub-page of the PCBA is represented as FCPN in Figure 1. The detail modeling of all the processes of the PCBA manufacturing system can be obtained from [2,7]. Three computer boards, namely P100, P133 and P166 are manufactured respectively. Since their manufacturing processes are the same, a set of machines is shared among the boards in order to reduce the machine costs. In order to capture information about this PCBA manufacturing process, the following variables are defined in DESIGN/CPN:

```
color PCB = with P100|P133|P166;
color realnumber = real;
color integer = int;
color str = string;
color Feeder = with feeder;
color Holder = with holder;
color Robot = with robot;
color Components = with c1|c2|c3|c4|c5;
```

```
color Robot_Component   = product Robot *
    Components;
color PCBA = product PCB * Components;
color ReadyForInsert = product Robot_Component *
    PCB * Holder;
color first = real;
color second = real;
color third = real;
color fourth = real;
color fifth = real;
color sixth = real;
color seventh = real;
color eighth = real;
color ninth = real;
color flag = bool with (no, yes);
color randomnumber = int with 1..20 declare ms;
color Reading = product flag * randomnumber;
color FuzzyReading = with Small | Medium | Big |
    VeryBig;
color fuzzyset = product first * second * third * fourth *
    fifth * sixth * seventh * eighth * ninth;
color finish = product realnumber * realnumber;
color threshold = product integer * realnumber *
    realnumber;
color weight = finish;
color stringset = product str * str;
color counter = integer;
var pcb : PCB;
var pcba : PCBA;
var component : Components;
var ready : ReadyForInsert;
var fv, fx, fy : fuzzyset;
var tx, ty, t2 : fuzzyset;
var t3, t7, t8 : fuzzyset;
var fin, def : finish;
var robot_comp : Robot_Component;
var readin : Reading;
var readout : Reading;
var reading : Reading;
var rg, freading : FuzzyReading;
var realin, realout : realnumber;
var thresin : threshold;
var w : weight;
var adjust, ok, insert : stringset;
var in1, in2, in3, in4, out : fuzzyset;
var realin1, realin2, out1, out2 : realnumber;
var xvn, xvs, xvm, xvb, yvn, yvs, yvm, yvb : fuzzyset;
var a1,a2,a3,a4,a5,a6,a7,a8,a9,a10 : threshold
var b1,b2,b3,b4,b5,b6,b7,b8,b9,b10 : threshold
var x,y : counter;
```

In Figure 1, there are three kinds of printed circuit board and each finished PCB required five types of components. The universe of discourse of x-displacement and y-displacement are both {1, 500, 1000, 1500, 2000, 2500, 3000, 3500, 4000, 4500} and the

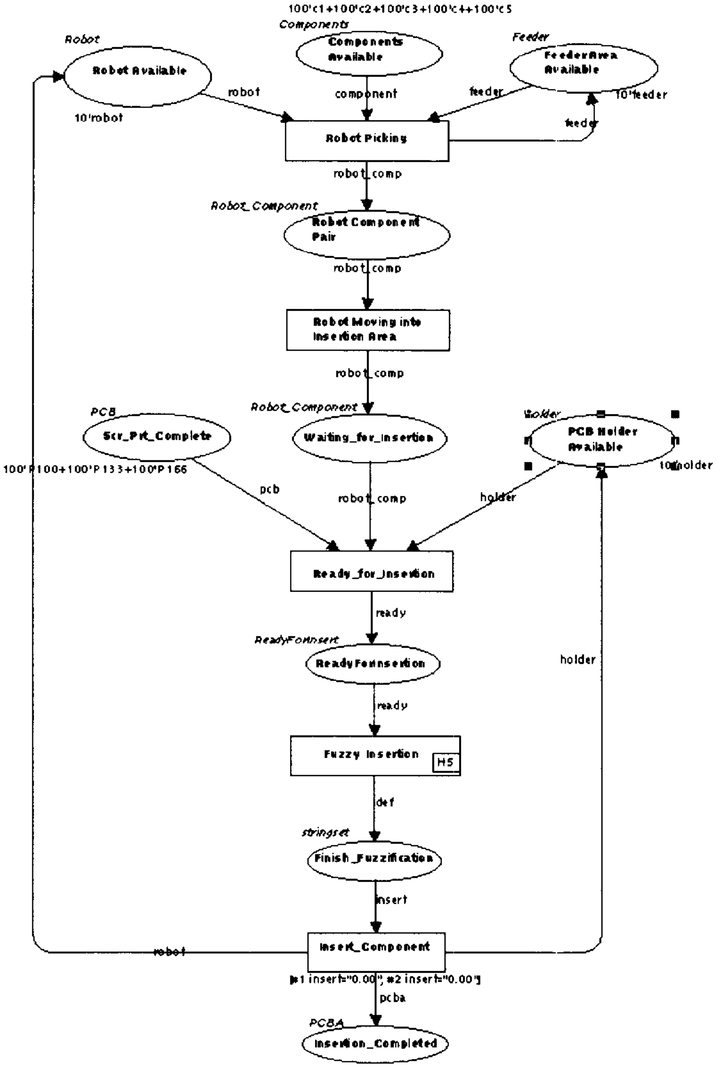

Figure 1: Component Insertion Sub-Page of the PCBA

universe of discourse of velocity is {5, 10, 15, 20, 25, 30, 35, 40, 45}and the color "randomnumber" is used for mapping the numerical displacement reading to the fuzzy one, that is, 1 to 5 represent Small, 6 to 10 represent Medium, 11 to 15 represent Big and 16 to 20 represent Very Big. The color "flag" is used for controlling the loop back algorithm and the color "stringset" is used for comparing the values after defuzzification since there is no support in the round off function for the real number in the DESIGN/CPN.

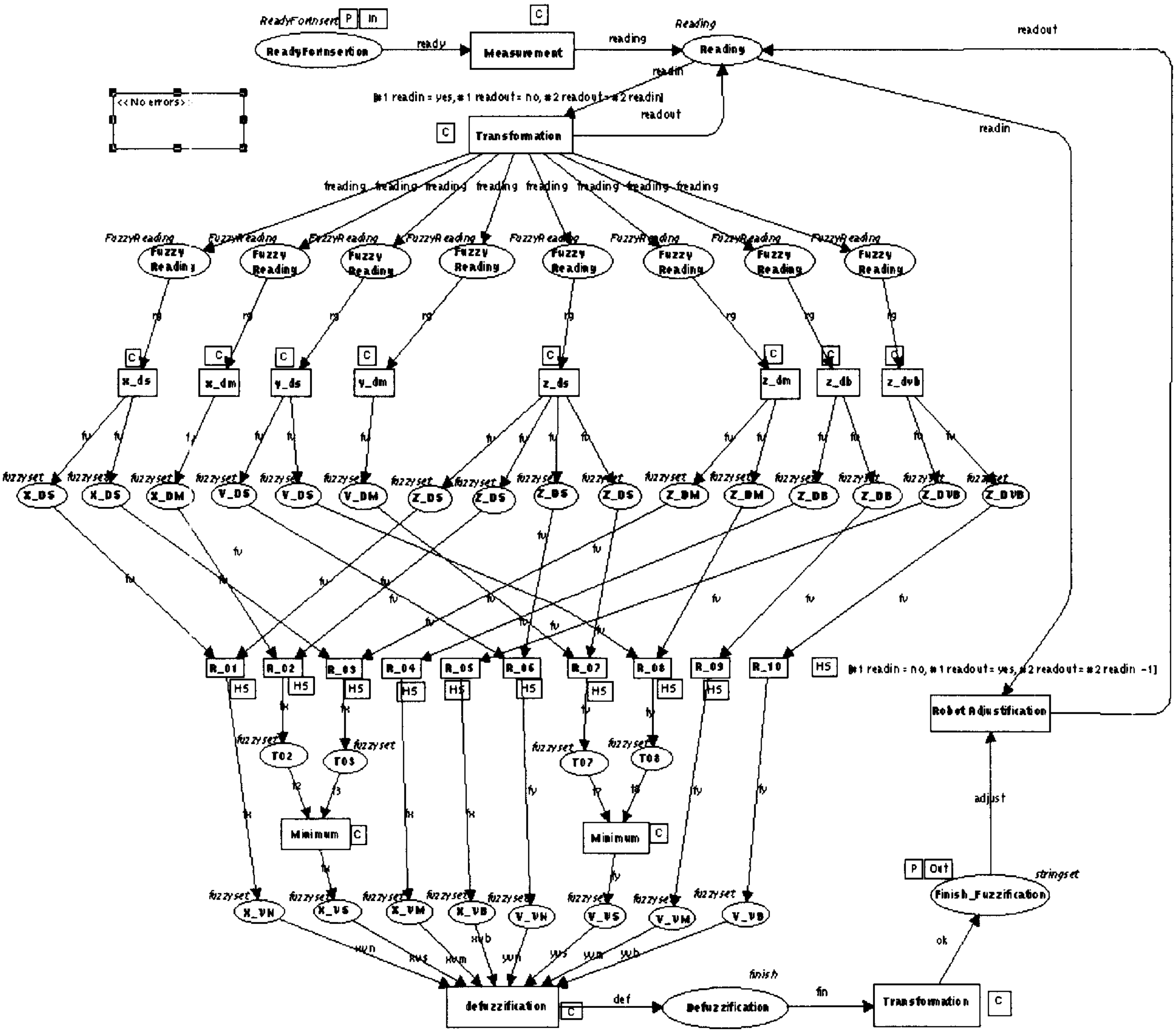

Figure 2 : FCPN Model of the Fuzzy Production Rules

The fuzzy production rules used are as follows:

1. X's displacement is Small and Z's displacement is Small, then X's velocity is Zero.
2. X's displacement is Medium and Z's displacement is Small, then X's velocity is Low.
3. X's displacement is Small and Z's displacement is Medium, then X's velocity is Low.
4. Z's displacement is Big, then X's velocity is Medium.
5. Z's displacement is Very Big, then X's velocity is Big.
6. Y's displacement is Small and Z's displacement is Small, then Y's velocity is Zero.
7. Y's displacement is Medium and Z's displacement is Small, then Y's velocity is Low.
8. Y's displacement is Small and Z's displacement is Medium, then Y's velocity is Low.
9. Z's displacement is Big, then Y's velocity is Medium.
10. Z's displacement is Very Big, then Y's velocity is Big.

The above is modeled in DESIGN/CPN as Figure 2. At first, the reading of PCB is measured, and creates a tuple {Boolean (Yes), randomly integer (between 1 to 20)}. Then, the reading is transformed to fuzzy reading and the original token is returned to the place except the Boolean value which is changed from "Yes" to "No". This will prevent the transition fire again since there is a guard in which only Boolean "Yes" is allowed to fire the transition. The creation of this guard is to model the following feedback loop:

1. Sensor data as input
2. Firing of the Fuzzy Production Rules to see if components are ready to be inserted into the board
3. If not ready to insert, then adjust the angles of the board, and obtain a new set of sensor data, GOTO step 1.

Different rules will be fired according to their threshold, and after defuzzification the outcome will compare to the preset value, e.g. "0.00". If it cannot pass the test, the token will be feed backward for next cycle, otherwise, the components can be inserted onto the board. Note that in Figure 2, there is a MIN operation between the output rule sets {Rule2, Rule3} and {Rule7, Rule8} because their output linguistic variable are both "LOW". In order to safe guard the accuracy of the output fuzzy set, a MIN (AND) operation was performed.

Each individual production rule is further represented by a FCPN sub-page which is shown in Figure 3, (i.e. fuzzy production rule no. 1). Altogether, we have built a total 13 Hierarchy Pages, 28 Color Sets, 144 Places, 288 Arcs and 48 Transitions. The hardware we used to perform the experiments is SUN's Ultra Sparc I Workstation with 64MB RAM, 143 MHz Processor with a Sparc floating point processor and 2G hard disk. Operation system used is Solaris 5.5.1. and DESIGN/CPN used is version 3.0.2.

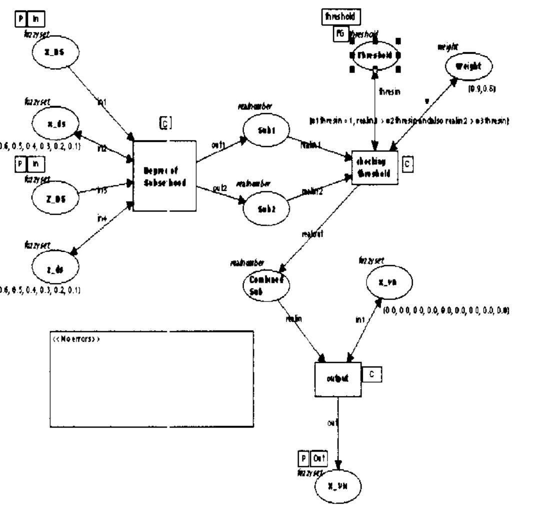

Figure 3 : FCPN Sub-page of Fuzzy Production Rule 1

In Figure 3, the place "threshold" is a fusion place which are shared with all other rules. By using the fusion place concept in colored petri nets, we can define a global data structure to represent all the rules' threshold values at the same time.

4. AUTOMATIC DETERMINATION OF THE THRESHOLD VALUES USING FCPN

Fuzzy reasoning is one of the approximate reasoning methods in which the matching of a given fact with a proposition in the antecedent of a rule is approximate rather than exact, a threshold value I is initially assigned to each proposition in the antecedent to ensure that the degree of similarity between this proposition and its

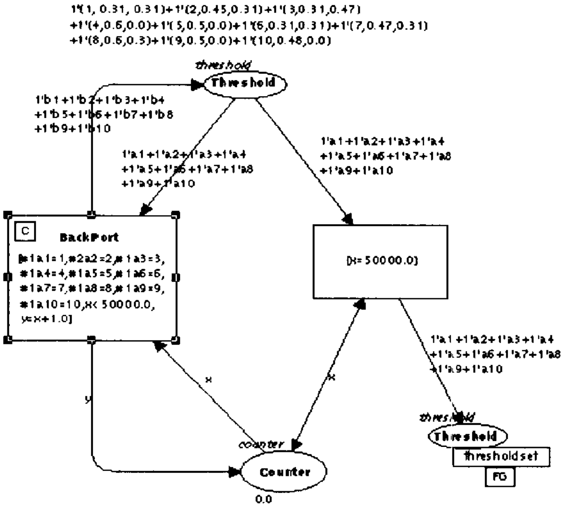

Figure 4: Back-propagation algorithm in DESIGN/CPN

given fact is greater than or equal to its corresponding threshold value. The assignment of a threshold value to each proposition in the antecedent of a Fuzzy Production Rule is to ensure that not only the result of an approximate reasoning method is reasonable but also to prevent or reduce rule mis-firing.

The threshold values in our Fuzzy Production Rules are arbitrary set initially. Then, we use the back-propagation algorithm [12] to learn the threshold value by a set of correct readings from the sensors. No target weight for these threshold values need to be assigned as we would expect the network to tune these threshold values so as to allow all the training samples to be fired. The implementation of the back-propagation algorithm in DESIGN/CPN is shown in Figure 4. The fusion place "threshold" is the same as represented in Figure 3. In the transition "BackPort", we defined all the necessary variables and calculations of the back-propagation algorithm, and specified 50,000 training cycles. Then, the respective threshold values can be obtained.

5. SUMMARY & FUTURE WORK

The above modeling and analyzing techniques have been used to analyze a PCBA manufacturing process in a factory located in China. By varying the allocation of different resources, and perform the simulation experiments on the model, the best solution obtained has a gain of 40% of overall productivity and 36% reduction of the breakeven time compared to the initial setting of the factory. Bottleneck processes have also been identified, and decision-makers have been given appropriate support through these experiments. Our future work includes extension of our methodology to solve due-date scheduling problems as well as the modeling of hybrid case-based expert systems for manufacturing systems

6. REFERENCES

[1] Cao, T., & Sanderson, A.C. (1995). "Task Sequence Planning Using Fuzzy Petri Nets," *IEEE Trans. on SMC*, Vol. 25, No. 5, pp. 755-768.

[2] Chan, T.S. (1998). Modeling of Printed Circuit Board Assembly manufacturing System using Colored Petri Nets, *MSc Dissertation*, The HK Polytechnic University.

[3] Cossins, R. & Ferreira, P. (1992). "CELERITAS: A Colored Petri Net Approach to Simulation & Control of Flexible Manufacturing Systems," *Int. J. of Production Research*, Vol. 30, No. 8, pp. 925-1956.

[4] Ezpeleta Joaquin, Colom Jose Manuel, & Martinez, Javier, (1995). "A Petri Net Based Deadlock Prevention Policy for Flexible Manufacturing Systems," *IEEE Trans. on R&A*, Vol. 11, No. 2, pp. 173-184.

[5] Hatono, I., Yamagata K. & Tamura H., (1991). "Modeling & On-Line Scheduling of FMS Using Stochastic Petri Nets," *IEEE Trans. on SE*, Vol. 17, No. 2, pp. 126-132.

[6] Jensen, K. (1995). *Colored Petri Nets: Basic Concepts, Analysis Methods & Practical Use* Volume 1,2&3, Springer-Verlag, Berlin, Heidelberg.

[7] Lau, K.O. (1998). Modeling & Analyzing of PCB manufacturing by Fuzzy Petri Nets, *MSc dissertation*, The HK Polytechnic University.

[8] Lee Doo Yong & DiCesare Frank, (1992). "FMS Scheduling Using Petri Nets & Heuristic Search," in *Proceedings of the 1992 IEEE Int. Conf. on R&A*, pp. 1057-1062.

[9] Scarpelli, H. & Gomide, F., (1996). "A Reasoning Algorithm for High-Level Fuzzy Petri Nets," *IEEE Trans. on FS*, Vol. 4, No. 3, pp. 282-294.

[10] Scarpelli, H., & Gomide, F. (1993). "Fuzzy Reasoning & Fuzzy Petri Nets in Manufacturing Systems Modeling," *J. of Intelligent & Fuzzy Systems*, Vol. 1, pp. 225-242.

[11] Silva, M., & Valetta, R. (1990). "Petri Nets & Flexible Manufacturing," *Advances in Petri Net '89*, Springer-Verlag, pp. 375-417.

[12] Tsang, E.C.C., & Yeung, D.S. (1997). Modeling Fuzzy Production Rules with Fuzzy Expert Networks. *Expert Systems with Applications*. Vol.13. No.3., pp.169-178, 1997.

[13] Viswanadham, N., Narahari Y. & Johnson, T.L., (1990). "Deadlock Prevention & Deadlock Avoidance in Flexible Manufacturing Systems Using Petri Net Models," *IEEE Trans. on R&A*, Vol. 6, No. 6, pp. 713-723.

[14] Xu, Y., Paul, R.P., & Shum, H.Y. (1991). Fuzzy Control of Robot & Compliant Wrist System, Conference Record of the 1991 *IEEE Industry Applications Society Annual Meeting*, pp. 1431-1437.

[15] Yeung, D., Liu, J., Shiu, S. & G. Fung, (1996). "Fuzzy Colored Petri Nets in Modeling Flexible Manufacturing Systems," in *Proceedings of the 9th Int. Symposium on AI, ISAI/IFIS*, November 12-15, Cancun, Mexico, 1996, pp100-107.

[16] Yeung, D.S. & Tsang, E.C.C. (1998), :"A Multi-Level Weighted Fuzzy Reasoning Algorithm for Expert Systems," *IEEE Trans. on SMC part A*, Vol.28., No.2., March, 1998, pp.149-158.

[17] Yeung, D.S., Shiu, S.C.K., Tsang, E.C.C., Modeling Flexible Manufacturing Systems Using Weighted Fuzzy Colored Petri Nets, to appear in *Journal of Intelligent & Fuzzy Systems*.

[18] Zhou, M., & Leu, M.C. (1991). Modeling & Performance Analysis of a Flexible PCB Assembly Station Using Petri Nets, *Transactions of the ASME: Journal of Electronic Packaging*, pp. 410-416.