AN ABSTRACT OF THE THESIS OF

Eric T. Wolbrecht for the degree of Master of Science in Mechanical Engineering presented on June 25, 1998. Title: Monitoring and Diagnosis of a Multi-Stage Manufacturing Process using Bayesian Networks.

Redacted for Privacy

Abstract approved: ______________________________

Robert Paasch

This thesis describes the application of Bayesian networks for monitoring and diagnosis of a multi-stage manufacturing process, specifically a high speed production part at Hewlett Packard. Bayesian network "part models" were designed to represent individual parts in-process. These were combined to form a "process model", which is a Bayesian network model of the entire manufacturing process. An efficient procedure is designed for managing the "process network". Simulated data is used to test the validity of diagnosis made from this method. In addition, a critical analysis of this method is given, including computation speed concerns, accuracy of results, and ease of implementation. Finally, a discussion on future research in the area is given.

Monitoring and Diagnosis of a Multi-Stage Manufacturing Process using Bayesian Networks

by

Eric T. Wolbrecht

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

Presented June 25, 1998
Commencement June 1999

Master of Science thesis of Eric T. Wolbrecht presented on June 25, 1988.

APPROVED:

Redacted for Privacy

Major Professor, representing Mechanical Engineering

Redacted for Privacy

Chair of Department of Mechanical Engineering

Redacted for Privacy

Dean of the Graduate School

I understand that my thesis will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my thesis to any reader upon request.

Redacted for Privacy

Eric T. Wolbrecht, Author

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

LIST OF FIGURES

LIST OF TABLES

MONITORING AND DIAGNOSIS OF A MULTI-STAGE MANUFACTURING PROCESS USING BAYESIAN NETWORKS

1 INTRODUCTION

Hewlett Packard in Corvallis, Oregon manufactures several precision products on high speed, automated assembly lines. An essential process in the production of one of these products is the alignment of a cap to the base part. This process is performed in several automated stages with significant part queuing between stages. Performance of this process is dually important. First, the quality of the product depends on the positional accuracy of the cap. Second, minimization of the production line, including the yield loss of the alignment process, presents a significant opportunity to reduce manufacturing costs.

In order to improve performance of the alignment process a prototype of a real time monitoring and diagnosis system was developed. The purpose of this system is to expeditiously identify component failures. The potential advantages of this system include yield improvement, improved product quality, data reduction for process operators, and reduced labor requirements.

The system designed for monitoring and diagnosis of the alignment process is composed of Bayesian networks, a probabilistic modeling technique. Bayesian networks have several advantages over other diagnostic methods. First, Bayesian networks provide a complete probabilistic description of a domain without specifying the probabilities of all propositions. This solves the intractability problem of traditional probabilistic modeling while not sacrificing completeness. Second, Bayesian networks provide better resolution for variable representation than traditional deterministic methods. Finally, Bayesian networks utilize prior knowledge of the causal relationships between variables in the domain.

The primary goal of this research is to develop a general approach for monitoring and diagnosis of a multi-stage manufacturing process using Bayesian networks. Though the specific goal of this paper is to provide monitoring and diagnosis of the cap alignment process, the methods used in the approach are applicable to other multi-stage manufacturing processes. The approach should be scalable in both speed and memory requirements for significantly larger applications.

This research is unique because it applies Bayesian networks to a multi-stage process containing numerous parts. In addition, monitoring and diagnosis is performed on-line in real time with the intent of identifying problems as soon as possible and determining the most probable source. This differs from traditional Bayesain network applications where diagnosis is performed after a failure has occurred and the machine or system has been shut down.

This report describes the application of Bayesian networks in developing a system for monitoring and diagnosis of the cap alignment process. First, a general description of the alignment process is given, followed by a brief introduction to Bayesian networks. Next, the designs of Bayesian networks used to model both the cap and base part assembly and the alignment process are presented. This is followed by an outline of system implementation. Testing procedures and results are presented next. Finally, conclusions are discussed followed by recommendations for future work.

2 BACKGROUND

Bayesian networks have been used in numerous applications over the past several years. Some of these applications include traffic scene analysis (Huang, 1994), general equipment diagnosis for photolithographic sequences (Leang, 1997), manufacturing and process diagnosis (Agogino, 1986), and tracking and avoidance of objects for automated vehicles (Alag, 1995). Bayesian networks have also been applied recently at Hewlett Packard for integrated circuit tester diagnosis (Mittelstadt, 1995).

In addition to real world application, research has been performed over the past few years to extend the scope of traditional Bayesian network diagnosis. This research has included real-time diagnosis (D'Ambrosio 1995,1996), decision-theoretic troubleshooting (Breese, 1996), troubleshooting under uncertainty (Heckerman, 1994), and monitoring multi-stage manufacturing processes (Rao, 1995).

This research in this paper utilizes the previous work on Bayesian networks, including design considerations and inference algorithms. This research is different in that it attempts to provide diagnosis in real-time as parts are produced. This is achieved by designing a general Bayesian network to represent each part and connecting these networks to form one large process network. Inference of each part network is performed using existing methods. The algorithm for inference of the process network is unique to this paper and is the main contribution of this research.

3 THE CAP ALIGNMENT PROCESS

3.1 Basic Layout

The cap alignment process consists of four main stages: 1) cap alignment, 2) pre-join operations, 3) the join process, and 4) post-join operations. The alignment operation is performed in parallel by three separate aligners. An upstream process feeds base parts and cap material into the three aligners automatically. The aligned cap and base parts then flow out of the three aligners and into a single part stream. This part stream is fed to pre-join operations where inspection takes place. Next, the joining operation receives the single stream of parts from the pre-join operations. After the parts are joined they are fed from a single part stream to the post-join operations. The parts are then split into two part streams for post-join inspection, which is performed with two sensor systems. A simplified diagram of this process is shown in Figure 3.1.

3.2 Inspection Data

The existing systems used for control of the cap alignment automated assembly line provide data from six inspection points throughout the process: after alignment, after pre-join operations, and after post-join operations. Each of these inspection points is capable of rejecting parts except for pre-join inspection, which performs position measurements only. This difference does not effect the operation of the monitoring and diagnosis system, so all data acquisition will be referred to as inspection. The location of these data inspection points can be seen in Figure 3.1. The following three subsections describe the data received from these inspection points.

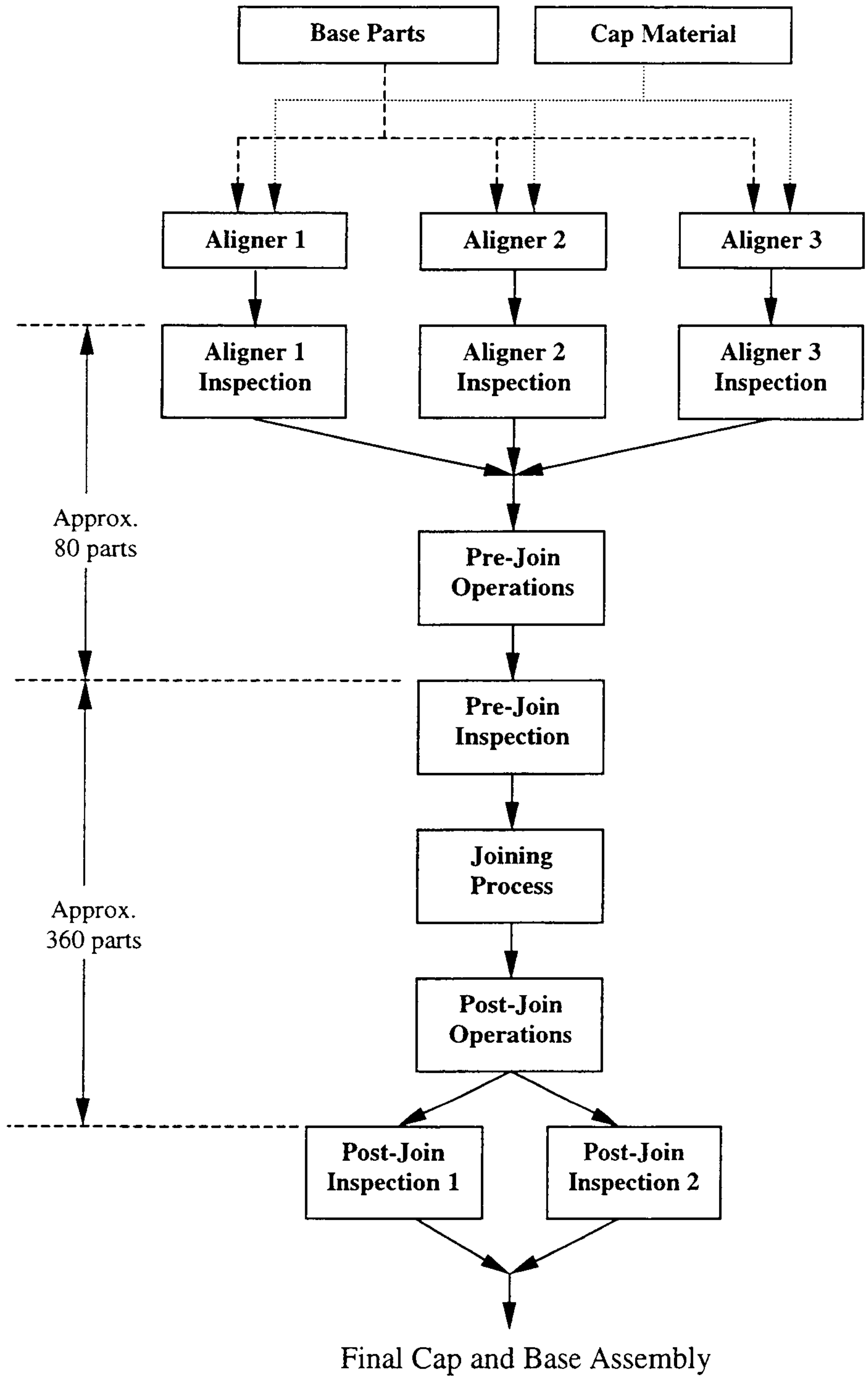


Figure 3.1: Diagram of the cap alignment process showing the process layout, part numbers, and data inspection points.

3.2.1 Aligner Inspection Data

Alignment inspection provides the following data fields: Aligner_Used, Date/Time, Part_ID, dX, dY, and dThZ. The first three identify the aligner used, the day and time of the measurements, and the part, respectively. The last three are continuous variables that measure a feature associated with the alignment process. This is not a measurement of the absolute position of the cap, which makes pre-join inspection necessary.

3.2.2 Pre-Join Inspection Data

Pre-join measurement provides the following data fields: Date/Time, Part_ID, dX, dY, and dThz. The first two identify the day and time of the measurements, and the part, respectively. The last three are continuous variables that measure the location of the cap with respect to the base part after the pre-join operations.

3.2.3 Post-Join Inspection Data

Post-join inspection is performed in parallel with two different sensor systems. Post-join inspection provides the following data fields: Sensor_Used, Date/Time, Part_ID, dX, dY, and dThZ. The first identifies which of two post-join sensors was used. The next two identify the day and time of the measurements, and the part, respectively. The last three are continuous variables that measure the location of the cap with respect to the base part after the post-join operations.

3.3 Production Rate and Part Flow

Cap alignment is a high speed, automated process. The speed of the total process requires a large number of parts to be queued between stages. There are about 26-27 parts in process between pre-join and each aligner, for a total of approximately 80 parts. There are about 360 parts between post-join and pre-join inspection. The number of parts on the assembly line is seen in Figure 3.1.

3.4 Component Failure Types

The cap alignment process consists of three fundamental component types: sensors, operations, and materials. Each of these is capable of failing, either isolated or in conjunction with other failures. A failure implies improper operation or improper characteristics of a component, but does not necessarily indicate that parts are being made out of specifications. The purpose of this system is to identify these failures before they produce parts out of specifications.

Sensor failures occur at the inspection points shown in Figure 3.1. There are six sensor systems: aligner 1 sensor system, aligner 2 sensor system, aligner 3 sensor system, pre-join sensor system, and post-join sensor systems 1 and 2. Sensor malfunctions cause local data errors but do not directly affect downstream processes and do not necessarily indicate the production of bad parts.

There are five separate sources of operation failures: aligner 1, aligner 2, aligner 3, pre-join, and post-join. An operation failure will affect the data from every future operation in the cap alignment process.

The only material failure source is the cap material. Cap material may have incorrect dimensions or features. This type of failure will affect all of the data received from the inspection points. Note that the thermal properties of the cap material may vary significantly batch to batch. This type of failure will only surface after the joining process in the data received from post-join, and is also a material related problem. However, for the purpose of this discussion it is treated as a post-join operation failure.

4 BAYESIAN NETWORKS

Bayesian networks are probabilistic models of a domain. A Bayesian network is a directed acyclic graph (DAG). The nodes in the graph represent variables within the domain of interest. The directed links between nodes represent the causal relationships between the variables. A directed link from one node to a second node indicates an influence of the first node, called the parent node, on the second node, called the child node. The degree of influence that a parent node has on a child node is represented by the conditional probabilities associated with the child node. In complete Bayesian networks, parent nodes may have several children and likewise child nodes may have several parents.

4.1 Bayes' Rule

The foundation for Bayesian networks is Bayes' Rule, seen in Equation 4.1. This rule governs the conditional probabilistic relationship between two Boolean variables.

Equation 4.1: Bayes' Rule.

$$P(B|A) = \frac{P(A|B) \cdot P(B)}{P(A)}$$

By designating the variables B and A in Equation 4.1 as a parent and a child node, respectively, Bayes' Rule allows the inference of the posterior probability, $P(B|A)$, of the parent node given the state of the child node.

4.2 State Space Variables

Nodes in a typical Bayesian Network represent state space variables. Each node has a conditional probability table that stipulates the probabilities of the node's states given the states of node's parents. For example, in the Bayesian network seen in Figure 4.1 the probability of variable *B* having state True may be more likely if the variable *A* has state True and the variable *C* has state False. This causal relationship is defined in the conditional probability table of node *B*, seen in Table 4.3. Nodes *A* and *C* have no parents, and therefore only prior probabilities are defined, as seen in Table 4.1, and Table 4.2, respectively.

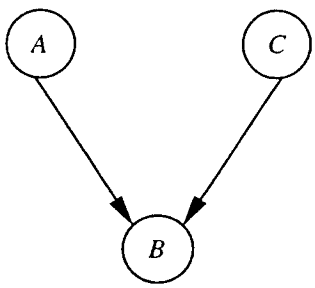


Figure 4.1: An example Bayesian network.

A	
P(True)	*P(False)*
0.80	0.20

Table 4.1: Prior probability table for node *A*.

C	
P(True)	*P(False)*
0.40	0.60

Table 4.2: Prior probability table for node *C*.

		B	
A	*C*	*P(True)*	*P(False)*
True	*True*	0.75	0.25
True	*False*	0.90	0.10
False	*True*	0.15	0.85
False	*False*	0.65	0.35

Table 4.3: Conditional probability table for node *B*.

This representation can be extended for Bayesian networks with a large number of nodes and variables with three or more states.

4.3 Observation and Inference

In diagnosis applications the typical goal is to determine the posterior probabilities of parent nodes given observations on the child nodes. For example, consider the Bayesian network of Figure 4.1. If the state of B is observed to be *True*, then the posterior probability of A=*True* can be determined using a modified version of Bayes' rule.

4.4 Continuous Variables

By defining distributions for each state of a variable, Bayesian networks can be used to represent continuous variables. Consider again the Bayesian network of Figure 4.1. Suppose when B=*True* a normal distribution is expected, and when B=*False* a uniform distribution is expected. Then when a value, x, is given for B, rather than observing B=*True* or B=*False*, observations are made to the relative likelihood of B=*True* and B=*False*. Calculating the relative heights of the normal and uniform distributions at x does this.

5 PART MODEL DESIGN

The system developed for monitoring and diagnosis of the cap alignment process is based on Bayesian network models of each cap and base part. These "part models" represent the probabilistic relationships between inspection data, alignment position, and the alignment process components. The part models are combined to form the process model, discussed in the next chapter. Diagnosis is then performed by Bayesian inference of the process model.

This chapter describes the design of the Bayesian network used to represent each cap and base part. The design is presented in four sections. The first section describes the nodes of the part model. The second section explains the causal relationships between the nodes. The next section details the conditional and prior probabilities associated with each node. The final section discusses the distributions defined to represent the states of each inspection field node.

5.1 Node Definitions

There are four basic node categories in the part model: position nodes, delta nodes, inspection nodes, and component nodes. Node descriptions of each of these are presented in the next four subsections.

5.1.1 Position Nodes

Position nodes represent the alignment position of the cap at the three basic inspection points. The name, states, and description of these three nodes are given in Table 5.1.

Node Name	States	Description
Apos	OK, Fault	Inkjet head position after the alignment process.
PreJPos	OK, Fault	Inkjet head position after the pre-join operations.
PostJPos	OK, Fault	Inkjet head position after the joining process and post-join operations.

Table 5.1: Nodes representing the cap position at the three basic inspection points.

5.1.2 Delta Nodes

Delta nodes represent a feature associated with the alignment process. Delta nodes are defined for each of the three aligners. This permits the part model to represent parts produced by any of the three aligners. These nodes are defined in Table 5.2.

Node Name	States	Description
A1Delta	OK, Fault	A feature associated with alignment from alinger 1.
A2Delta	OK, Fault	A feature associated with alignment from alinger 2.
A3Delta	OK, Fault	A feature associated with alignment from alinger 3.

Table 5.2: Nodes representing a feature associated with alignment.

5.1.3 Inspection Nodes

Inspection nodes represent the data observed at the inspection points. As mentioned in Section 3.2, there are three basic inspection points: aligner inspection, pre-join inspection, and post-join inspection. The following subsections describe inspection nodes representing these three inspection points.

5.1.3.1 Aligner Inspection Nodes

As mentioned in Section 3.2.1, aligner inspection produces three continuous variable fields that measure a feature associated with the alignment process: dX, dY, and dThZ. Nine inspection nodes are defined in Table 5.3 below to represent these continuous variable fields. A node is also defined which represents the aligner used.

Node Name	States	Description
A1DdX	OK, Fault	Data field dX of aligner 1 inspection.
A1DdY	OK, Fault	Data field dY of aligner 1 inspection.
A1DdThZ	OK, Fault	Data field dThZ of aligner 1 inspection.
A2DdX	OK, Fault	Data field dX of aligner 2 inspection.
A2DdY	OK, Fault	Data field dY of aligner 2 inspection.
A2DdThZ	OK, Fault	Data field dThZ of aligner 2 inspection.
A3DdX	OK, Fault	Data field dX of aligner 3 inspection.
A3DdY	OK, Fault	Data field dY of aligner 3 inspection.
A3DdThZ	OK, Fault	Data field dThZ of aligner 3 inspection.
AUsed	A1,A2,A3	Aligner used for the present part.

Table 5.3: Inspection nodes representing the continuous variable data fields associated with an alignment feature.

5.1.3.2 Pre-Join Inspection Nodes

As mentioned in Section 3.2.2, the pre-join inspection produces three continuous variable fields that measure the position of the cap on the base part: dX, dY, and dThZ. To represent these continuous variable fields from pre-join inspection three inspection nodes are defined in Table 5.4.

Node Name	States	Description
PreJdX	OK, Fault	Data field dX measuring alignment position at pre-join inspection.
PreJdY	OK, Fault	Data field dY measuring alignment position at pre-join inspection.
PreJdThZ	OK, Fault	Data field dThZ measuring alignment position at pre-join inspection.

Table 5.4: Inspection nodes representing the continuous variable data fields from pre-join inspection.

5.1.3.3 Post-Join Inspection Nodes

As mentioned in Section 3.2.3, post-join inspection produces three continuous variable fields that measure the position of the cap on the base part: dX, dY, and dThZ. To represent these continuous variable fields from post-join inspection three inspection nodes are defined in Table 5.5. An additional node is defined representing the post-join sensor used for inspection of the present part.

Node Name	States	Description
PreJdX	OK, Fault	Data field dX from pre-join inspection.
PreJdY	OK, Fault	Data field dY from pre-join inspection.
PreJdThZ	OK, Fault	Data field dThZ from pre-join inspection.
PJSUsed	S1, S2	Sensor used for the present part.

Table 5.5: Inspection nodes representing the continuous variable data fields from post-join inspection.

5.1.4 Component Nodes

Component nodes represent the basic components that constitute the alignment process. The following subsections describe the component nodes associated with the three basic assembly line processes: alignment, pre-join, and post-join.

5.1.4.1 Aligner Component Nodes

The aligner component nodes represent the basic components of the alignment process and the aligner inspection. Two nodes are defined for each aligner: one representing the aligner and one representing the aligner sensor. In addition, a node is defined to represent the cap material, which feeds all three aligners. These definitions of these seven nodes are given in Table 5.6.

Node Name	**States**	**Description**
A1	OK, Fault	Aligner 1.
A1Sens	OK, Fault	Sensor used for aligner 1 inspection.
A2	OK, Fault	Aligner 2.
A2Sens	OK, Fault	Sensor used for aligner 2 inspection.
A3	OK, Fault	Aligner 3.
A3Sens	OK, Fault	Sensor used for aligner 3 inspection.
Material	OK, Fault	Cap material.

Table 5.6: Component nodes associated with aligner 1, aligner 2, and aligner 3.

5.1.4.2 Pre-Join Component Nodes

The pre-join component nodes represent the basic components of the pre-join process and pre-join inspection. A single node is defined to represent both, as seen in Table 5.7.

Node Name	**States**	**Description**
PreJoin	OK, Fault	The condition of the pre-join process.
PreJSens	OK, Fault	The condition of the sensor used for pre-join inspection.

Table 5.7: Component nodes associated with the pre-join process and pre-join inspection.

5.1.4.3 Post-Join Component Nodes

The post-join component nodes represent the basic components of the joining process, the post-join process, and post-join inspection. A single node is defined to represent both the join and the post-join process. Two nodes are defined to represent the post-join sensors. These nodes are defined in Table 5.8.

Node Name	States	Description
PostJoin	OK, Fault	The condition of the join and post-join process.
PostJSens1	OK, Fault	The condition of sensor 1 used for post-join inspection.
PostJSens2	OK, Fault	The condition of sensor 2 used for post-join inspection.

Table 5.8: Component nodes representing the joining and post-join process, and post-join inspection.

5.2 Causal Relationships

In the cap alignment process, each operation is dependent upon the accuracy of the previous operation. For example, if the position of the cap is faulty after the alignment operation, then the position of the cap is expected to be faulty after pre-join and after post-join, regardless of the state of those two operations. Therefore, the node *APos* is a parent of the node *PreJPos*, which is a parent of the node *PostJPos*. These are the primary nodes in the part model.

There is the possibility that two successive operations are faulty but the data received from inspection after both operations is good. This would occur only if the faults were counter-acting. The probability of this is extremely low and therefore it is not considered in the part model as a possibility. This greatly reduces the hypothesis space of the part model without significantly affecting the accuracy of diagnosis.

The next three sections of this chapter discuss the causal relationships between these three nodes and the rest of the nodes in the part model. The completed part model is shown in Figure 5.1.

5.2.1 Aligner Nodes

The position of the cap after the alignment operation is dependent upon the mentioned feature associated with the alignment operation. If the feature is faulty, then the cap position will be faulty. Therefore delta nodes *A1Delta*, *A2Delta*, and *A3Delta* are parents of the position node *APos*. However, for a particular part, only one of the three delta nodes influences the position node *APos*. Thus the inspection node *AUsed* is also a parent of the node *APos*. The only parent of each delta node is its respective aligner component node, because if an aligner is not properly functioning the feature associated with the alignment process is expected to be faulty. The feature is measured at the aligner inspection. If the feature is faulty, then the aligner inspection data fields will be faulty. Therefore each delta node is the parent of its respective inspection nodes. The aligner inspection fields will also be faulty if the aligner sensor is faulty, and therefore the component nodes *A1Sens*, *A2Sens*, and *A3Sens* are parents of their respective aligner inspection nodes.

5.2.2 Pre-Join Nodes

The cap position after the pre-join operation is dependent upon the cap position after the alignment operation, as mentioned above, and the pre-join operation. If the pre-join operation is functioning improperly, then the cap position at pre-join inspection will be faulty. Therefore the component node *PreJoin* is a parent of the position node *PreJPos*. The cap position after the pre-join operation is measured at pre-join inspection. If the cap position is faulty, then the pre-join inspection data fields will be faulty. Therefore the position node *PreJPos* is a parent of the inspection nodes *PreJdX*, *PreJdY*, and *PreJdThZ*. These inspection nodes will also be faulty if the pre-join sensor is faulty, and thus are children of the component node *PreJSens*.

5.2.3 Post-Join Nodes

The cap position after join and post-join operations is dependent upon the cap position after the pre-join operations, as mentioned above, and the join and post-join operations. Therefore the component node *PostJoin* is a parent of the position node *PostJPos*. The cap position after join and post-join operations is measured at post-join inspection. If the cap position is faulty, then the inspection data fields are expected to be faulty. Thus the position node *PostJPos* is a parent of the inspection nodes *PostJdX*, *PostJdY*, and *PostJdThZ*. These inspection fields will also be faulty if the post-join inspection sensor used if faulty. Therefore the nodes *PostJSens1*, *PostCSens2*, and *PJSUsed* are also parents of the post-join inspection nodes.

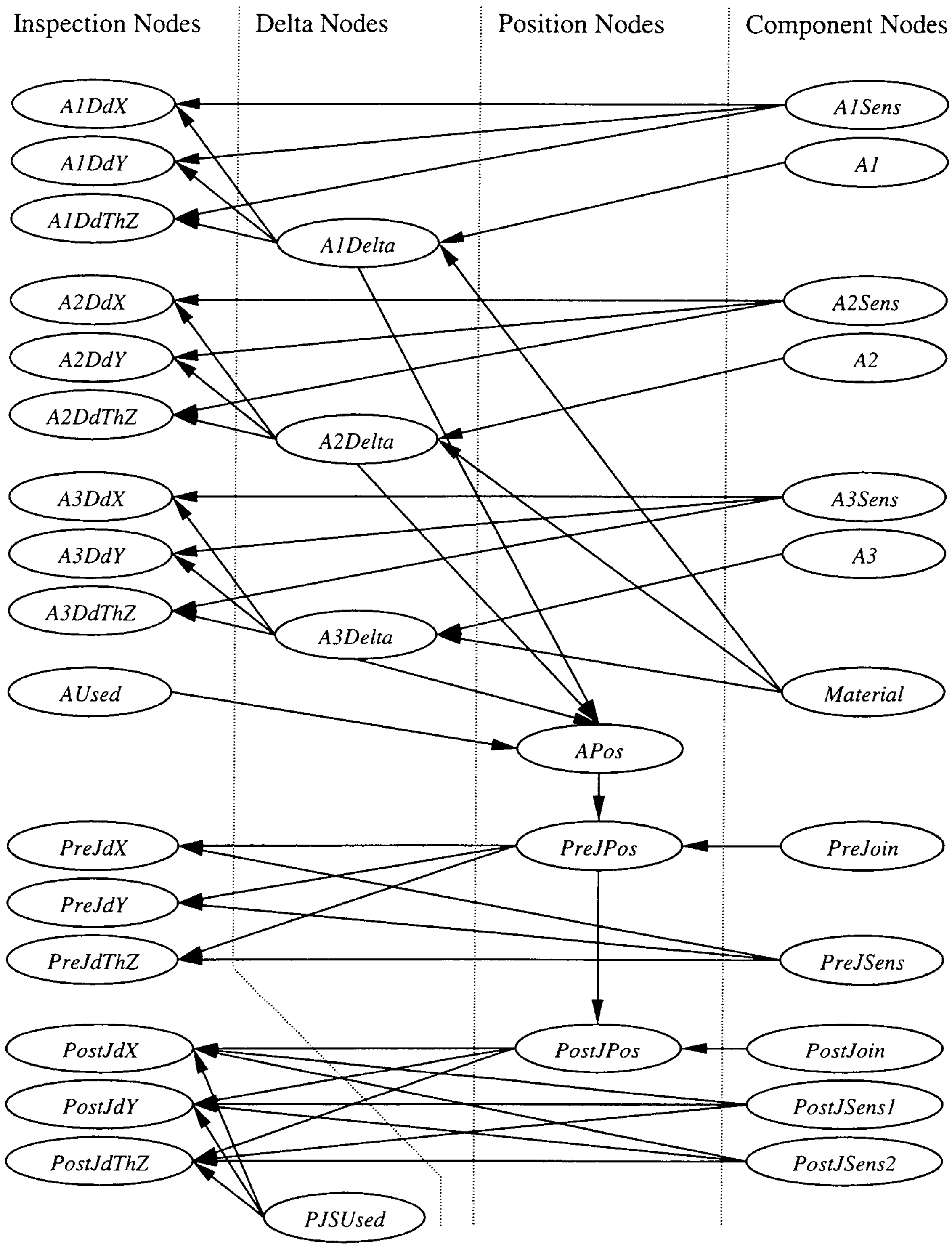


Figure 5.1: The Bayesian network part model.

5.3 Conditional and Prior Probabilities

The position of the cap at each of the three basic inspection points is dependent upon all preceding operations. If any one of the parent operations is faulty, then the cap position will be faulty. Likewise, the data received at the inspection points is dependent upon both the position of the cap and the inspection sensors. For these reasons, the probability of each child in the part model has value one for the union of its' appropriate parents, and zero for all other state combinations. For most nodes, the appropriate parents are all parents of that node. For nodes whose parents include the nodes *AUsed* or *PJSUsed* the appropriate parents are all parent nodes except for those not indicated by the node *AUsed* or *PJSUsed.* Table 5.9 shows the conditional probabilities for all child nodes.

Child node type	Parents	*P(Child=OK)*	*P(Child=Fault)*
Nodes where parents do not include *AUsed* or *PJSUsed*	$(P_1{=}OK \ldots \wedge P_n{=}OK)$	1	0
	$\neg(P_1{=}OK \ldots \wedge P_n{=}OK)$	0	1
Nodes where parents include *AUsed* or *PJSUsed*	$((Used{=}P_{Used} \wedge P_{Used}{=}OK) \wedge (P_1{=}OK \ldots \wedge P_n{=}OK))$	1	0
	$\neg((Used{=}P_{Used} \wedge P_{Used}{=}OK) \wedge (P_1{=}OK \ldots \wedge P_n{=}OK))$	0	1

Table 5.9: Conditional probabilities of all child nodes in the cap and base part model.

The prior probabilities of the component nodes in the part model are dependent upon the prior probabilities of future parts and the posterior probabilities of earlier parts. Therefore, the prior probabilities of the component nodes in the part model are assigned to be 0.5 for each state. This allows each part model to return relative likelihoods of the component nodes to the process model. The process model can then calculate the posterior probabilities of any component node at any location or time in the alignment process. This is discussed further in the next chapter.

5.4 Data Distributions

Distributions must be defined to represent each state of the continuous variable inspection nodes. Normal distributions are defined for the state *OK*. The mean and standard deviation of these distributions are defined by the system engineer and are based on historical data. The form of the distributions of erroneous data is not known, so uniform distributions are defined for the state *Fault*. This simplifies the calculations while still modeling the belief that data away from average is more likely faulty. The height of the uniform distribution is calculated from the difference limits established by the systems engineer. The difference limits represent the points away from the mean at which the data is considered faulty. The height is calculated so that it is equal to the height of the normal distribution at the difference limits, as shown in Figure 5.2. By doing this, data received at the difference limits is considered equally likely to be from the state *OK* as from the state *Fault*.

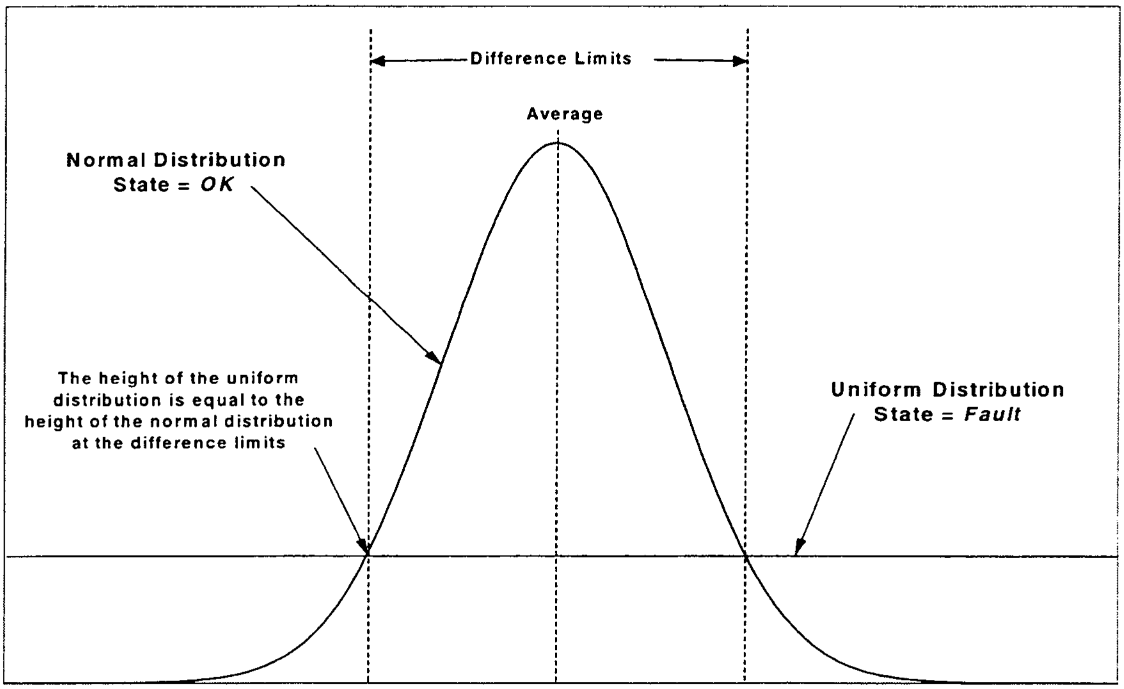


Figure 5.2: The relationship between the normal and uniform distributions.

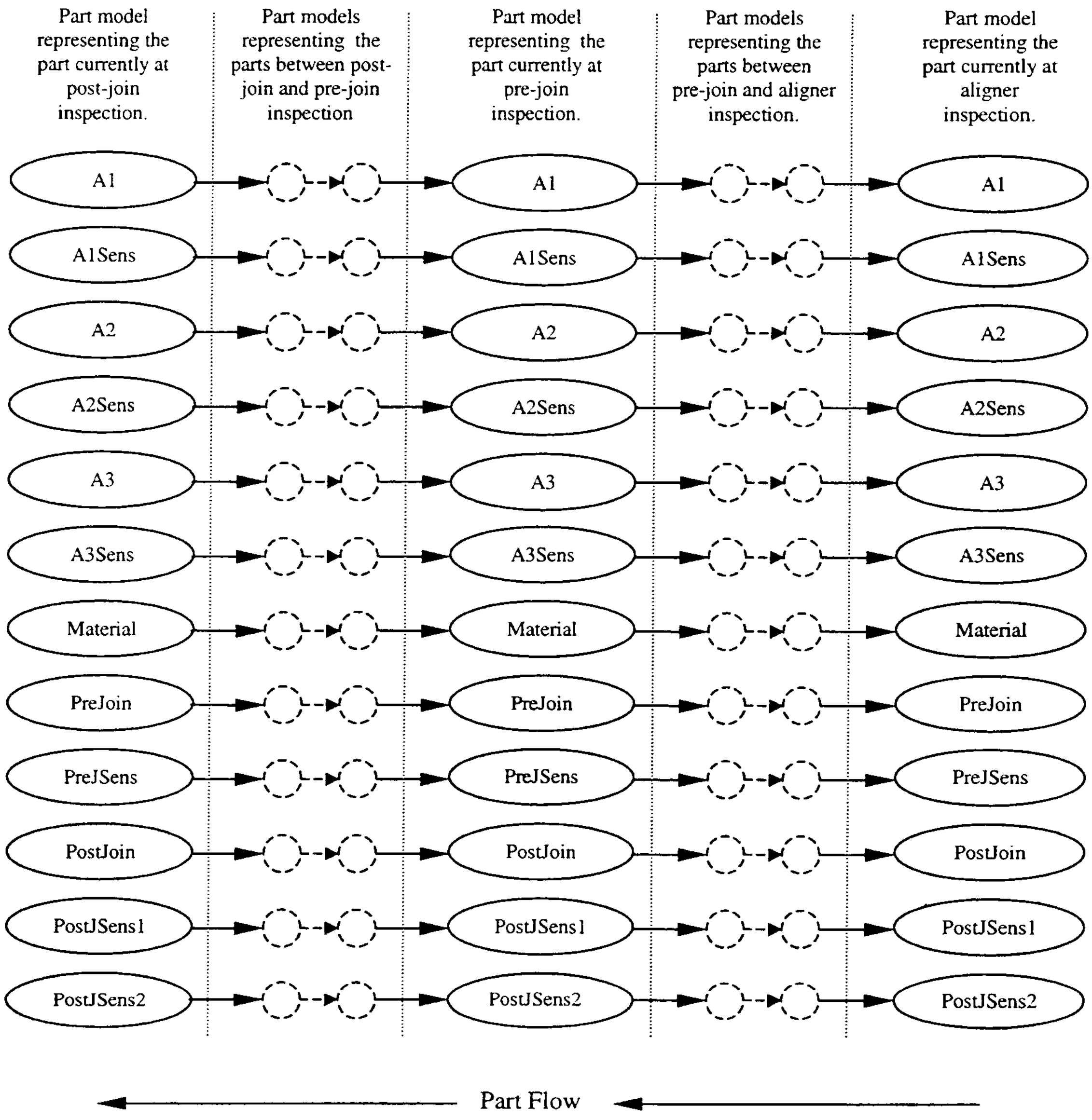
6 PROCESS MODEL DESIGN

The "process model" combines the "part models" from the previous section into a single Bayesian network representing all parts currently in the alignment process. The process model is used to determine the posterior probabilities of the alignment process components given the data observed from basic inspection points. This section presents the design of this process model. A description of the network structure is given first, followed by an explanation of the posterior probability-updating algorithm.

6.1 Network Description

The process model is constructed by connecting multiple part models. Each component node in the part model is the child of the corresponding component node in the previous part model and the parent of the corresponding component node in the next part model. This represents the causal relationship between consecutive parts in the alignment process. For example, if a component node has the state *OK* for one part, then the state of the corresponding component node of the next part has a high probability of also being *OK*. The process model is shown in Figure 6.1. The first row in the part model represents the prior probabilities of the components given no information.

The posterior probabilities of a component node for particular part model in the process model represents the posterior probability of that component at the time corresponding to the particular part. For example, consider a part currently at post-join inspection. The posterior probability of the component node, *PreJoin,* represents the posterior probability of the pre-join operation at the time the part passed through pre-join, which was many parts earlier. Therefore the current posterior probabilities of the components in the process model are calculated at the part model representing the most recent part through the corresponding component.



Each column of nodes represents a part model*. The gray nodes represent the nodes where the posterior probability represents the most current component belief. *note-only component nodes are shown for the part models.

Figure 6.1: The process model.

6.2 Posterior-Updating Algorithm

The function of the process model is to determine component posterior probabilities based on observed data from the inspection points. Figure 6.2 shows the basic algorithm developed for this purpose.

Inspection data is read in time order from five different input files corresponding to the five inspection points. The part identification number is used to check the data against the part already represented in the system. If a matching part is found, then the new data is observed on the existing part model. If a matching part is not found, then the system locates a place for a new part model. If the system is full, the part models are indexed and the oldest part model is removed to make place for the new part model. Once a place is found, a new part model is loaded and the data is observed.

Once the new data is properly observed the current part model is queried and the component joint probabilities are returned. These are used to update the posterior probabilities of the current part model, the previous part models, and the subsequent part models. When the joint likelihoods are multiplied, a small transition value is added to each entry. This allows the posterior probabilities to slowly change from part to part. In essence, this is the same as assigning a small probability of state change for component nodes in successive part models in the process model. Furthermore, posterior updating can be performed only after a specific number of data entries are received. This allows the run-time of the program to decrease without significant loss of system responsiveness. This is important for high-speed assembly lines such as the cap alignment process.

After the posterior probabilities are updated, the algorithm outputs the appropriate component beliefs from the current part model. The appropriate component beliefs are those for which data has been observed. For example, the component beliefs outputted from pre-join inspection include the nodes *A1*, *A1Sens*, *A2*, *A2Sens*, *A3*, *A3Sens*, *Material*, *PreJoin*, and *PreJSens*. This provides useful information about the state of the alignment process at the current time and at the time when the part went through upstream processes.

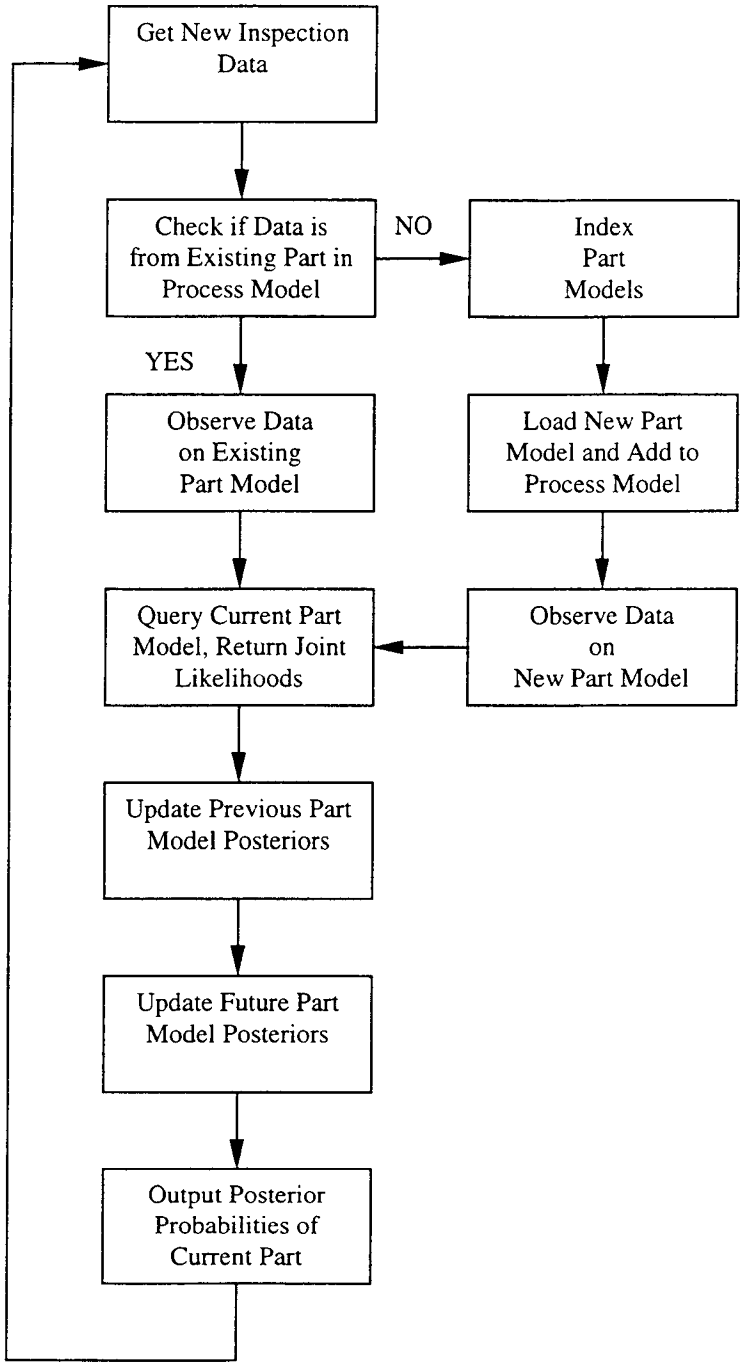


Figure 6.2: The posterior-updating algorithm.

7 SYSTEM IMPLEMENTATION

The part model was designed and constructed using Strategist, a Bayesian network modeling software application from Prevision. Once the part model is completed, it is saved as a .spi file, the standard file format for Bayesian networks. The process model was implemented in Visual C++ from Microsoft. Part models are loaded using routines from BMR.lib, a Bayesian modeling and reasoning library available from Prevision. BMR.lib was also used for making observations on the part models and for querying the part models. The posterior updating algorithm uses the results from the part model queries to calculate the posteriors of all the parts in the alignment process. These computations are implemented in Visual C++.

8 TESTING AND RESULTS

As mentioned in the previous section, the system designed for monitoring and diagnosis of the cap alignment process reads data from five input files that correspond to inspection data from the three aligners, pre-join, and post-join. These files can be obtained in real-time from the actual alignment process. However, before the system can be applied to actual production it was first validated using simulated data with known characteristics. By using simulated data, simplified typical faults of a known origin can be tested. The results of such tests help to better define the capabilities of the system and therefore should provided a better understanding of the results obtained when the system is applied to the actual production process.

This chapter describes the testing performed on simulated data. Fortunately, Hewlett Packard already has a process emulator that can generate simulated data in the form of the five data files previously discussed. This emulator was used to generate simulated fault scenarios. Single fault scenarios were tested first followed by some typical multi-fault scenarios.

Testing and results are presented in four sections. The first section describes testing on a typical simulated data set. This data set covers several single fault scenarios. The data is presented graphically along with an analysis of the results, which are also presented graphically. The second section discusses the results from additional testing on single fault scenarios. The final two sections discuss the results from testing on multiple faults and process drift.

8.1 A Typical Simulated Data Set

This section describes the data and the results from a typical simulated data set. The simulated data set is composed of six single faults occurring over several hundred parts in the alignment process. The data consists of inspection data from the five input files corresponding to aligner inspection, pre-join inspection, and post-join inspection.

The simulated data has two significant characteristics: Local deviation and long term process drift. Three graphs were generated showing the simulated data as well as the sytsem output from each of the three basic inspection points.

The first three faults in this simulated data set are aligner failures. Each aligner produces data that is offset from the mean for approximately 300 parts. This is a simplified representation of an aligner failure. The entire process is cumulative, so when the parts corresponding to the faulty data are simulated at pre-join and post-join inspection, the same offset will be apparent. Had these been aligner sensor failures then the data would be faulty at aligner inspection only. Figure 8.1 shows the inspection data from the aligners for this simulated data set.

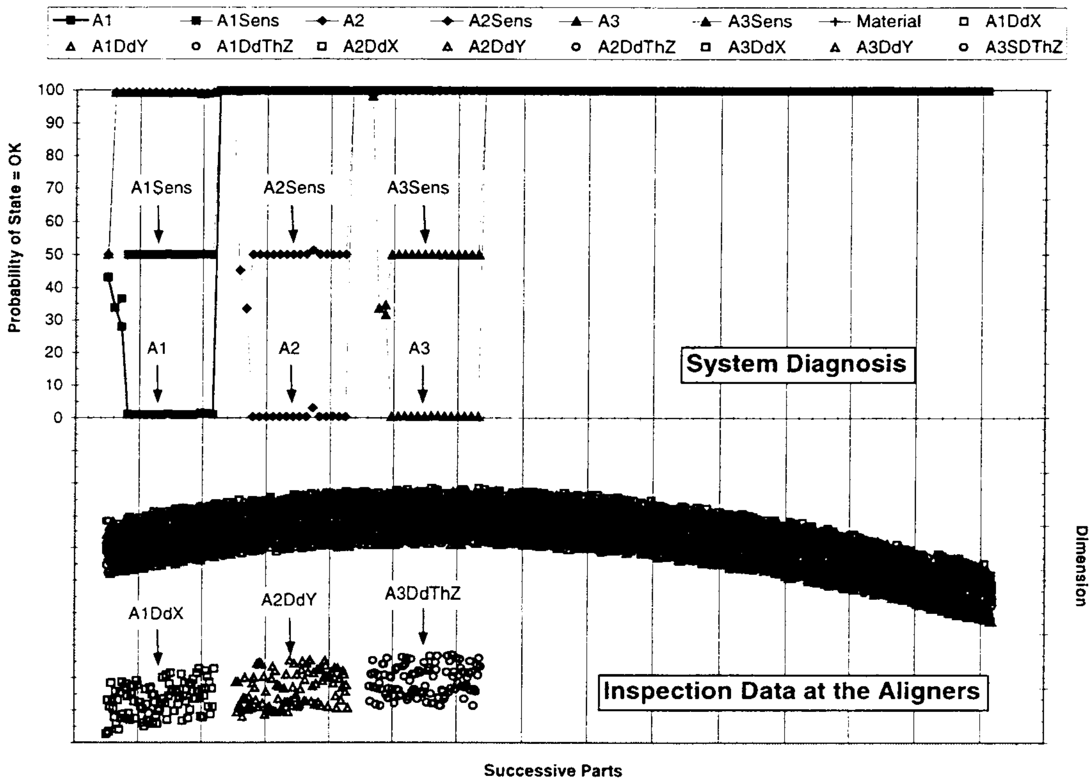


Figure 8.1: System diagnosis and inspection data at the aligners for a typical data set.

When faulty data is received at one of the aligners the system initially diagnosis that both the aligner and the aligner sensor are faulty. The system recognizes that three hypothesis exist to explain the faulty data: 1) both the aligner and the aligner sensor are faulty, 2) the aligner is faulty, and 3) the aligner sensor are faulty. Because both the aligner and the aligner sensor appear in two out of these three hypotheses, the posterior probability of both initially fall to 33%, which represents the probability that *State=OK*. This is shown in Figure 8.1.

When the faulty parts associated with these three aligner failures are inspected at pre-join, the system is able to propagate the information gained from pre-join inspection back to the system diagnosis at aligner inspection. This information propagation is a function of distance between inspection points. Aligner inspection and pre-join inspection are relatively close, so this information propagation from pre-join inspection is significant to the aligner diagnosis. Once the new information reaches the aligner diagnosis, the system recognizes that the aligner must be faulty, and thus eliminates the hypothesis that the aligner sensor is faulty. Because the aligner appears in both of the remaining hypotheses, it's posterior probability drops to close to 0%. The posterior probability of the aligner sensor rises slightly to 50% because it exists only in one of the remaining hypotheses.

The system has an easier time arriving at this diagnosis at pre-join and post-join inspection. This is because the system has all upstream inspection data available. This can be seen in Figure 8.2 and Figure 8.3. The diagnosis of these aligner failures at pre-join and post-join inspection is actually a diagnosis of what the state of the process was at the time the current part passed through the aligners. For example, when the post-join diagnosis indicates an aligner failure, it actually refers to the state of the aligner when the alignment operation was performed on the part. This is useful as a process history, but is not as useful for real-time monitoring.

The fourth failure in the simulated data set is a pre-join sensor failure. The inspection data at pre-join is offset from the mean for a period of approximately 300 parts. The offset, however, is not present when the same parts are inspected at post-join. This represents a simplified pre-join sensor failure where the sensor calibration is off but parts at pre-join inspection are actually within the set difference limits. The data received at pre-join inspection is shown in Figure 8.2. This data shows both the pre-join sensor failure and the aligner failures discussed in the preceding paragraphs.

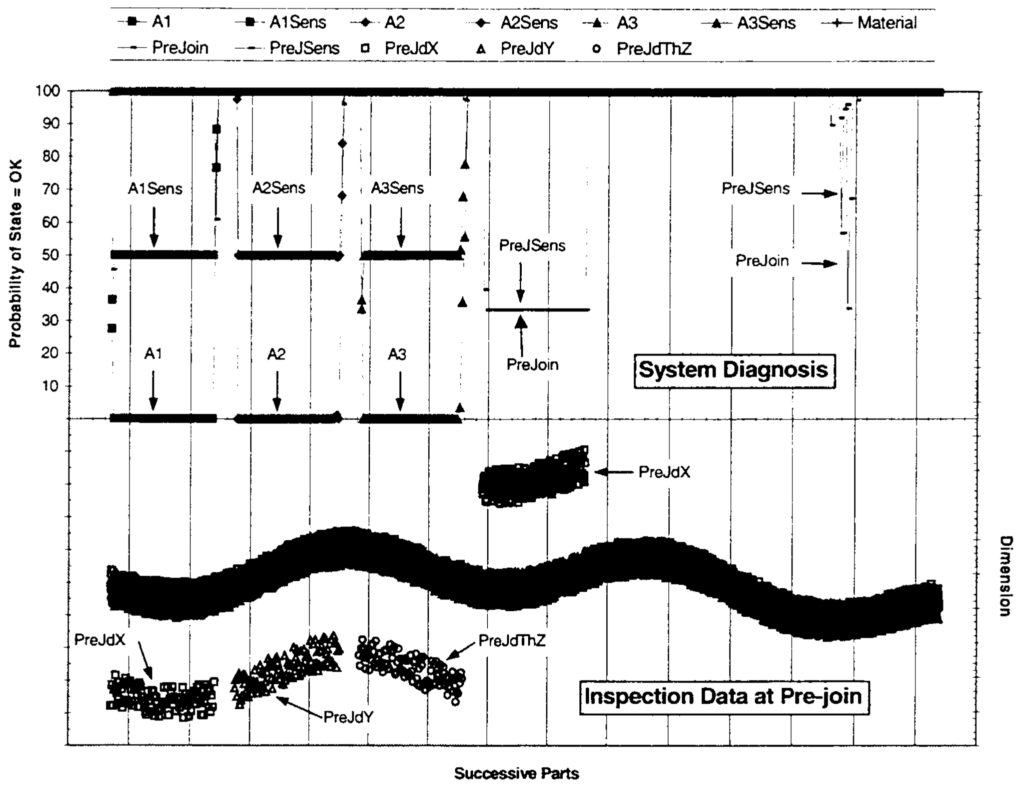


Figure 8.2: System diagnosis and inspection data at pre-join for a typical data set.

When the faulty data is received from pre-join inspection, the system recognizes that three hypotheses exist as a pre-join diagnosis, similar to the response from the aligner diagnosis for the aligner failures. These three hypotheses are: 1) pre-join failure and pre-join sensor failure, 2) pre-join failure, and 3) pre-join sensor failure. Because both pre-join and pre-join sensor appear in both of these hypotheses, the posterior probabilities of both drop to 33%. Because pre-join inspection and post-join inspection are separated by over 300 queued parts, the data from this pre-join sensor failure never appears simultaneously at both pre-join and post-join inspection. Therefore the diagnosis at pre-join is unable to differentiate between these three hypotheses and the posterior probabilities remain at 33%, as shown in Figure 8.2. If the failure continued longer the post-join diagnosis, which correctly diagnoses the pre-join sensor as the only fault would be able to propagate information back to the pre-join diagnosis. The diagnosis at pre-join would then increase the posterior probability of pre-join back to 100% and reduce the posterior probability of the pre-join sensor to 0%.

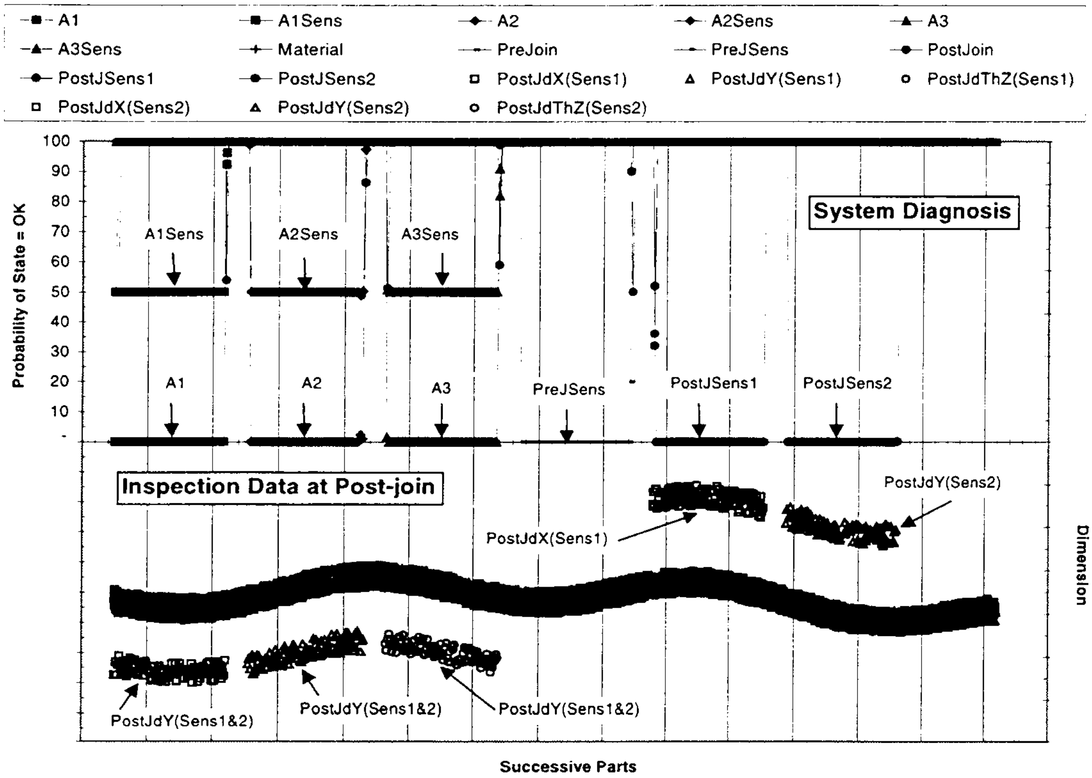


Figure 8.3: System diagnosis and inspection data at post-join for a typical data set.

In addition to the pre-join diagnosis of the aligner failures and pre-join sensor failures there exists a significant spike at close to the last 300 parts. This spike shows both pre-join and the pre-join sensor posterior probabilities dropping as far as 33%. This occurs because the process at pre-join drifts away from the mean and begins to pass the set difference limits.

The final two failures in this simulated data set are, in order, post-join sensor 1 failure and post-join sensor 2 failure. The inspection data at post-join is offset for each sensor on two separate occasions. This represents a post-join sensor failure where one sensor is calibrated incorrectly, producing faulty data, while the other sensor works correctly, producing normal data. The data received at post-join inspection is shown in Figure 8.3. Also evident in this data is the offset from the earlier aligner failures, which is correctly diagnosed. Noticeably missing is any offset from the pre-join sensor failure. This allows post-join inspection to correctly diagnose the pre-join sensor failure.
Unlike the previous failures, the post-join diagnosis can immediately determine that the only feasible hypothesis is a post-join sensor failure. This is because while one sensor is failing, the other is properly functioning and thus producing normal data. This normal data indicates that post-join is operating correctly, and therefore cannot be the source of the fault. The post-join diagnoses this correctly, and thus the posterior probability of each sensor is reduced to 0% when the sensors fail independently.

8.2 Additional Single Faults

There are four additional single fault scenarios not discussed in the previous section: 1) aligner sensor failure, 2) cap material failure, 3) pre-join failure, and 4) post-join failure. Each of these failures were simulated and tested, and the results are given in the following paragraphs. The diagnostic results are presented in the form: component node(probability of state=OK). Only those nodes with posterior probabilities less than 100% are given.

When an aligner sensor fails, the data read at the respective aligner inspection is offset from the mean. However, this offset is not evident at pre-join and post-join inspection. This is because an aligner sensor failure indicates only faulty measurements, not faulty position. When the faulty data is initially read the system cannot differentiate between an aligner failure and an aligner sensor failure, resulting in a diagnosis at the aligners of: aligner(33%), and aligner sensor(33%). Once the faulty data reaches pre-join inspection, the system recognizes that the alignment position is good, and thus the diagnosis at pre-join is: aligner sensor(0%). Because the system already knows the alignment position is good, the diagnosis at post-join is the same as the diagnosis at pre-join.

A cap material failure produces faulty data for all three aligners and all down stream inspections. The system diagnosis at aligner inspection is: aligner 1(45%), aligner 1 sensor(45%), aligner 2(45%), aligner 2 sensor(45%), aligner 3(45%), aligner 3 sensor(45%), and material(30%). Here the system recognizes the material is the most probable source of the failure, but because the data is faulty at all inspections the system cannot eliminate other failure hypotheses. When a material failure reaches pre-join inspection, the pre-join diagnosis improves slightly to: aligner 1,2, & 3(44%), aligner 1,2, & 3 sensor(46%), material(25%), pre-join(47%), and pre-join sensor failure(47%). The post-join diagnosis is the same as the pre-join diagnosis for the aligner components and material. The post-join diagnosis slightly alters the pre-join diagnosis of the pre-join components to pre-join(46%) and pre-join sensor(48%), and also diagnoses the failure possibilities of the post-join components: post-join(49%), post-join sensor 1 (49%), and post-join sensor 2(49%).

The data from a pre-join failure is faulty at pre-join inspection and at both post-join inspections. When the faulty data is initially received at pre-join inspection, the pre-join diagnosis cannot distinguish between a pre-join failure and a post-join failure. The initial diagnosis at pre-join is: pre-join(33%) and pre-join sensor(33%). When the faulty data is received at post-join, the post-join diagnosis is able to conclude that the most-likely failure is pre-join(23%). The post-join diagnosis also recognizes other possible failure hypotheses: pre-join sensor(38%), post-join(43%), post-join sensor 1(48%), and post-join sensor 2(48%).

A post-join failure affects the data at both post-join inspections. The system recognizes that the faulty data from a post-join failure indicates that the best failure hypothesis is post-join(20%), but also recognizes the possibility of one or both of the post-join sensors failing, and thus produces the diagnosis: post-join sensor 1(40%) and post-join sensor 2(40%).

8.3 Multiple Faults

Most multiple faults have the same diagnosis as one of the single fault scenarios discussed in the previous two sections. For example, if an aligner failure and pre-join failure occur at the same time, the system will produce the same diagnosis as a single aligner failure. This is because an aligner failure affects all downstream inspections, so the data will be faulty at pre-join inspection regardless of whether pre-join is functioning properly. This line of reasoning applies to all multiple fault scenarios where one fault affects the data for another downstream fault.

Some multiple fault scenarios have a unique diagnosis. In all cases the system determines all possible fault hypotheses and produces a diagnosis based on how many of those fault hypotheses a particular component appears in. The more fault hypothesis a component appears in, the more likely it is to be faulty, and the lower the probability of *State=OK* will be.

One interesting multiple fault scenerio is two aligner failures. If aligner 1 and aligner 2 fail at the same time, the aligner diagnosis is: aligner 1 & 2(33%) and aligner sensor 1 & 2(44%). Unlike a material failure, the data from a dual-aligner failure will only affect two thirds of the data received at pre-join inspection. The remaining one third will be good data produced from aligner 3. This good data allows the system to conclude that both pre-join and pre-join sensor are *OK*. In addition, the faulty data received at pre-join inspection indicates that the failures could not have been only aligner sensor failures. This improves the aligner diagnosis to give the following pre-join and post-join diagnosis: aligner 1 & 2(0%), and aligner sensor 1 & 2(50%).

8.4 Process Drift

The failures discussed in the previous testing section have been modeled as step-failures. This implies that a dynamic event occurred producing a level shift in the inspection data from within the normal distribution to somewhere outside the normal distribution. Failures of this kind may occur for many reasons, including, for instance, when a machine breaks or a new batch of material is introduced which has improper dimensions.

Process drift is a different type of failure. Process drift occurs when inspection data slowly deviates from the mean over a significant period of time. As long as the drift away from the expected mean is large enough, the resulting diagnosis will be the same as those discussed in the previous sections. The major difference is how rapidly the diagnosis is made. In the previous sections, the diagnosis followed quickly after the failure events themselves. In a process drift failure, the diagnosis of each potential source component will be more gradual. This is more of a system feature than a system limitation, because it provides information about not only the most likely source of the failure but also indicates the rate at which the failure may have occurred.

9 CONCLUSIONS

The system presented in this report can provide correct diagnosis of process failures in real-time. The system serves as a process monitor that can detect a failure within 10 bad parts. Diagnosis accuracy improves when the bad parts generated from the failure are received at downstream inspection points.

In a real application of this system the failed component can be shut down when a failure is detected. Then when more evidence is received from the bad parts already in process the system can provide probabilistic diagnosis. For example, the first three failures from the first simulated data set were all aligners failures. Once the failures are detected, the appropriate aligner can be shut down and serviced. Then when the bad parts reach pre-join inspection the system can report the probable failures in order of most likely. In this case the most likely failure was the aligner itself (0%), followed by the aligner sensor(50%).

The large number of parts between pre-join inspection and post-join inspection makes diagnosis of pre-join component failures at pre-join inspection difficult. However, the system still recognizes that there is a failure, even though it has difficult differentiating between a pre-join failure and a pre-join sensor failure. This difficulty is due mostly to the queuing in the alignment process, and is not necessarily an indication of system limitations.

The accuracy of the system is dependent upon the accuracy of the configuration parameters. The configuration parameters include the mean, standard deviation, and difference limits representing each of the alignment processes. The diagnosis system can only provide useful information if these parameters are properly set. The system considers data within the difference limits as good and data outside of the difference limits as faulty. If these parameters do not correctly characterize the process then the system will view some good data as faulty and likewise some faulty data as good.

Bayesian networks applied in the manner presented in this report can provide a good model of the probabilistic relationships between multiple parts and multiple components in a multi-stage manufacturing process. The results from this research indicate that a system developed in this manner has the capability to represent a complicated process by modeling each part separately and then connecting the multiple part models to form one process model. This assumes the following three things. First, if a good part is produced, there is high probability that the next part produced will be good, and vice versa. Second, data from part inspections is correlated to both the state of the part and the state of the operations used to produce the part. And finally, the configuration parameters of the inspection data are well known.

By remembering the likelihoods from each individual part model and updating these individually when new data is received a large Bayesian network can be represented without having to query every part model when the posterior probabilities are determined. This significantly reduces the processing time needed to compute the posterior probabilities without effecting the accuracy of the results.

The posterior probability-updating algorithm can be modified to only update the posteriors after a certain amount of inspection data has been received. This can significantly reduce the processing time while not greatly effecting the response time of the system to faults. This was evident in the output results from the first simulated data set, where the posteriors were updated only after five parts passed through each inspection point. The system output still indicated failures within 10 parts, or approximately 15 seconds.

In general, this system may be applied to similar manufacturing processes where parts are produced in sequential manner and inspection is performed at several points throughout the process. The current algorithm used by the system to update the posterior probabilities is order n in the number of parts in the system and order n^2 in the number of component nodes. This means the system is not completely scalable to larger processes. Solutions to these problems are discussed in the next chapter.

10 CONTINUATION

The system developed for monitoring and diagnosis presented in this report currently outputs the posterior probabilities of each component in the alignment process. In the future this information can be combined with action cost information to develop a decision model for the alignment process. For example, the system was unable to differentiate between a pre-join failure and a pre-join sensor failure in the first simulated data set. In this case the best action is to attempt to repair the component with the smallest ratio of repair cost to prior probability of failure.

A developed application of this system could include many user interface features. Possible features include graphs of the posterior probabilities in real-time, decision trees used to determine optimal repair sequences, and process history reports. A completed system would include a monitoring mode, a repair mode, and a process history mode.

The system could also be implemented with a learning mode, in which data is recorded from good parts currently being produced. This data could then be used to determine the configuration parameters of the system.

More detailed part models could be developed and used to provide greater information to the operator. The detailed models could be analyzed only when a failure has been indicated using the simpler part models, thus avoiding speed problems. For example, each component node has two states: OK and Fault. This could be replaced by three component nodes representing the states of the component for the field dX, dY, and dThZ.

The current posterior updating algorithm records the joint likelihoods of each component in the part model. In this case, the part model had 12 components, and therefore the joint likelihoods contain 4096 numbers. To properly update the posteriors 4 arrays of length 360 are needed, with each element containing 4096 numbers. This presents a significant memory allocation problem. However, this problem can be handle by implementing a sparse matrix to represent these joint likelihoods. In a sparse matrix, only the significant numbers and their locations are recorded, and the other are all assumed to be some small delta value.

The current posterior updating alogorithm performs multiplication of the joint probabilities between every part model in the system model. This presents a significant speed problem. This problem too can be avoided with more efficient programming. The posterior probabilities are only important at each inspection point, and therefore need not be computed at every part. The total of the multiplications between two inspection points can be saved and then updated when new data is received. This would require only one set of multiplications between inspection points rather than a set of multiplications for every part between inspection points. This method is currently being researched.

BIBLIOGRAPHY

1. Agogino, Alice M, and A. Rege. *Sensor-Integrated Expert System for Manufacturing and Process Diagnostics.* Knowledge-Based Experts Systems for Manufacturing, ed. By S. C.-Y. Lu and R. Komanduri, ASME PED- vol. 24, 1986, pp. 67-83

2. Alag, Satnam, Kai Goebel, and Alice M. Agogino. *A Methodology for Intelligent Sensor Validation and Fusion used in Tracking and Avoidance of Objects for Automated Vehicles.* Proceedings of the American Control Conference, Seattle, WA, June 1995, pp. 3647-3653.

3. Breese, John S., and David Heckerman. *Decision-Theoretic Troubleshooting: A Framework for Repair and Experiment.* Technical Report, MSR-TR-96-06, Revised May 1996, Microsoft Research, Advanced Technology Division, Microsoft Corporation, One Microsoft Way, Redmond, WA 98052. Also appears in the Proceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence, August 1996.

4. D'Ambrosio, Bruce. *Real-time Value-driven Diagnosis.* Technical Report, CS dept., Oregon State University, 1995.

5. D'Ambrosio, Bruce, and Scott Burgess. *Some Experiments with Real-time Decision Algorithms.* Technical Report, CS dept., Oregon State University, 1996.

6. Heckerman. *Similarity Networks for the Construction of Multiple-Fault Belief Networks.* Proceedings of the Sixth Conference on Uncertainty in AI 1990., 1990.

7. Heckerman, David, John S. Breese, and Koos Rommelse. *Troubleshooting under Uncertainty.* Technical Report, MSR-TR-94-07, Revised September 1994, Microsoft Research, Advanced Technology Division, Microsoft Corporation, One Microsoft Way, Redmond, WA 98052. Portions appear in the Proceedings of the Fifth International Workshop on Principles of Diagnosis, New Paltz, NY.

8. Huang, T., Koller, D., Malik, J., Ogasawara, G., Rao, B., Russell, S., and Wber, J., *Automated Symbolic Traffic Scene Analysis Using Belief Networks.* Proceedings of National Conference on Artificial Intelligence, Morgan Kaufmann, San Mateo, CA, 1994.

9. Leang, Sovarong and Costas J. Spanos. *A General Equipment Diagnostic System and its Application on Photolithographic Sequences.* IEEE Transactions on Semiconductor Manufacturing, Vol. 10, No. 3, August 1997.

10. Mitchell, Tom M. 1997. *Machine Learning*. New York, NY.: MIT Press and The McGraw-Hill Companies, Inc.

11. Mittelstadt, D., R. Paasch, and B. D'Ambrosiso, *Application of a Bayesian Network to Integrated Circuit Tester Design*. Artificial Intelligence for Engineering, Design and Manufacture, Vol. 9, No. 1, pp. 51-65, 1995.

12. Rao, Suraj, Andrzej Strojwas, John Lehoczky, Mark Schervish. *Monitoring Multi-Stage Sequential Manufacturing Processes: A Bayesian Approach.* International Symposium on Semiconductor Manufacturing, 1995.

13. Russell, Stuart, and Peter Norvig. 1995. *Artificial Intelligence*. Englewood Cliffs, N.J.: Prentice Hall.