

Intelligent Process Diagnosis Based on End-of-Line Electrical Test Data

Ruey-Shan Guo*, Cheng-Kai Tsai**, Jian-Huei Lee***, Shi-Chung Chang**

*Department of Industrial and Business Administration, National Taiwan University

**Department of Electrical Engineering, National Taiwan University

***Taiwan Semiconductor Manufacturing Corporation

Abstract

The goal of this research is to develop a fuzzy logic-based system for a first-cut end-of-line diagnosis function. Based on measured abnormal electrical test data, the system provides the engineers a list of prioritized causes (process steps) for further investigation. The intelligent diagnosis system consists of three major modules: fuzzy modeling, knowledge base and inference engine. Experienced engineers' diagnosis knowledge is captured in the knowledge base using fuzzy logic knowledge representation models. Each major processing step's fault possibility is calculated in the inference engine. The intelligent diagnosis system has been validated against 23 real fab cases. Results show that version 2.0 of the system identifies the real causes as the top three causes in 20 cases. Our analysis indicates that the inference engine is robust but the knowledge base is insufficient. Improvement strategy has been to periodically update the knowledge base by field engineers based on lessons learned from the case study.

1. Introduction

Semiconductor wafer fabrication involves the most complex manufacturing processes. Wafer lot abnormality may occur due to many causes in the fabrication processes such as a single step process error, process integration problems, equipment faults and miss operations. Diagnosis of such problems has been a difficult task in the semiconductor manufacturing, and many approaches have been proposed in the past, such as statistical approaches [1][2], rule-based approaches [3][4], fuzzy logic and neural network approaches [5]-[8].

In this paper, we propose an intelligent system for a first-cut end-of-line diagnosis. Based on abnormal end-of-line electrical test data and fuzzy logic modeling of experienced engineers' empirical knowledge, the system provides the engineers a list

of prioritized causes (processing steps) for further investigation. This approach is based on the modification of a current industry diagnosis practice and the systematic modeling of associated empirical knowledge.

In the next section we review one of the current industry end-of-line diagnosis practices. An intelligent diagnosis system and its architecture are proposed in section 3. Three important modules of the proposed system: fuzzy modeling, knowledge base and inference engine, are described in sections 4 and 5. In section 6, we present the validation results. Conclusions are made in section 7.

2. End-of-line Diagnosis Practices

In a modern semiconductor wafer fab, a very high volume of end-of-line electrical test data and empirical knowledge associated with the fabrication processes are available for the purpose of end-of-line diagnosis. End-of-line data mostly refer to the wafer electric test data after completing the whole fabrication processes. They provide important information regarding process integration status. When measured end-of-line data show abnormal values, integration engineers must find out the root causes as quickly as possible and feedback such information to process engineers to prevent further errors.

In general, integration engineers use their process physics knowledge and abnormal test data to diagnose the root causes. In order to facilitate the diagnosis process, experienced engineers write down their knowledge into a knowledge table called "key node table." Fig. 1 depicts one such table used by the industry, which empirically correlates the abnormal end-of-line test data (symptoms) to major process steps (root causes). Numerical values in the table represent the relative rank. When compared columnwisely, number one represents the most probable cause. For example, when EOL_2

shows abnormal values, step_2 is the most probable cause compared to step_4 and step_1.

Processing Step	End-of line data			
	EOL_1	EOL_2	EOL_3	EOL_4
Step_1		3		1
Step_2		1		2
Step_3	1			3
Step_4	2	2	1	

Fig. 1 Key node table used by the industry.

Historically, field engineers use the key node table approach to perform the first-cut diagnosis. Although useful in some situations, this approach is insufficient. Several problems arise with the use of the key node table:

- Key node table is empirical. There is great fuzziness in defining this table by field engineers.
- Key node table is incomplete. The accuracy is highly dependent on engineers' experience.
- Diagnosis process is empirical and incomplete. There is no systematic inference method to determine the root causes when there are

multiple abnormal end-of-line measurements.

- Diagnosis process is not robust. The diagnosis accuracy is highly dependent on the accuracy of the key node table and end-of-line data.

3. Intelligent Diagnosis System

Due to many limitations of the current approach, an intelligent system for the first-cut diagnosis is proposed with the following objectives:

- to enhance the current diagnosis accuracy,
- to automate diagnosis process,
- to train junior engineers, and
- to facilitate knowledge integration / expansion.

Fig. 2 illustrates the architecture of the proposed intelligent system. It consists of six modules: fuzzy modeling, knowledge base, inference engine, history data base, I/O interface, and self-learning. As shown in Fig. 2, engineers provide abnormal end-of-line test data as the inputs to the system through the I/O interface. The intelligent diagnosis system then calculates each process step's fault possibility and provides the engineers a list of prioritized causes for further investigation. Experts' knowledge or key node table, either old or newly updated, can be also sent to the fuzzy modeling module for further processing and stored in the knowledge base.

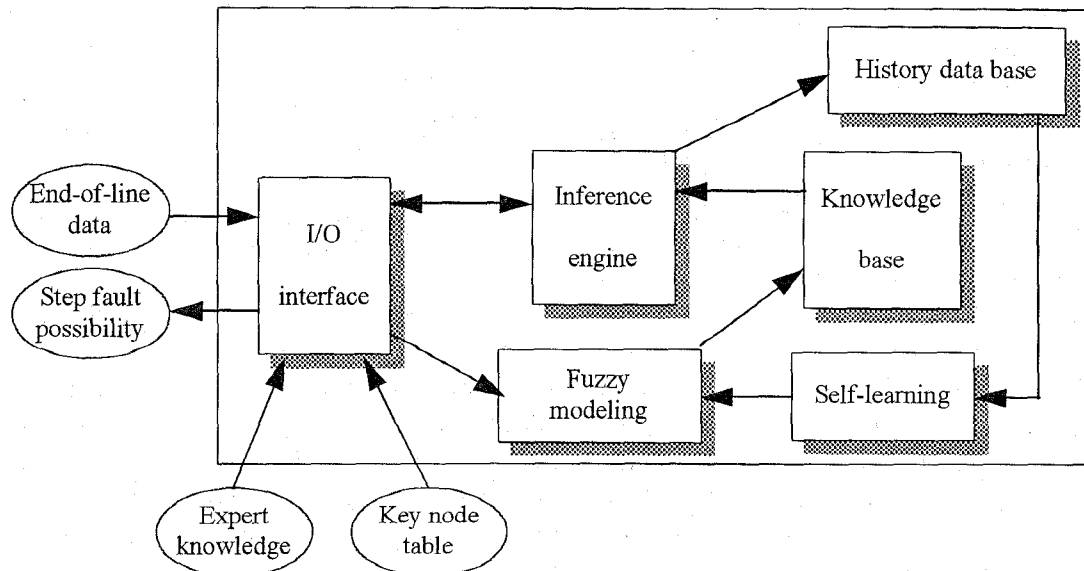


Fig. 2 Architecture of an intelligent diagnosis system.

In general, experts' diagnosis knowledge contains a lot of qualitative or uncertain information. Traditional probability concept cannot model such information very well. In the fuzzy modeling module, we use fuzzy logic's possibility concept to model vague information. The resulting information is then stored in the knowledge base.

With the information of abnormal end-of-line data and knowledge base, each major process step's fault possibility is calculated in the inference engine. The resulting information can be provided to the users through the I/O interface or sent to the history data base. The diagnosis results together with the real causes can be sent to the self-learning module for further knowledge training. This is useful when the initial knowledge base is insufficient or incomplete.

In this paper, three main modules, fuzzy modeling, knowledge base and inference engine, will be described in detail in the next two sections. The self-learning function is currently performed by experienced engineers by comparing the system's diagnosis results with the real causes.

4. Fuzzy Modeling / Knowledge Base

As mentioned earlier, several problems arise with the use of the key node table. In order to overcome these difficulties, several approaches are used to effectively extract knowledge.

The first approach is to redefine the key node table by many experienced field engineers. Through group discussion and information exchange, the resulting key node table represents the most complete key node table available. In addition, numerical number in each cell no longer represents the "rank" of the cause in which number one represents the most probable cause. Instead, it is redefined as the "correlation" between the abnormal end-of-line data and corresponding process steps. Here, a higher number represents a strong correlation and a lower number represents a weak correlation. This new definition has two advantages. First, each cell number can be defined individually. Second, the concept of strong, medium and weak correlation can be easily modeled using fuzzy logic theory.

The second approach is to use two correlation

numbers in each cell of the key node table. As shown in Fig. 3, the left-hand side number represents the correlation for a "small" abnormal end-of-line measurement, while the right-hand side number represents the correlation for a "large" abnormal end-of-line measurement. This new definition uses the fact that the correlation is usually higher when an end-of-line measurement shows a larger abnormality. An empirical rule to decide if an end-of-line measurement is a small error or a large error is to check if its value is beyond the internal spec or customer spec, as shown in Fig. 4.

Processing Step	End-of line data			
	EOL_1	EOL_2	EOL_3	EOL_4
Step_1		1 2		3 3
Step_2		3 3		2 3
Step_3	3 3			1 2
Step_4	2 3	2 2	3 3	

Fig. 3 Modified key node table.

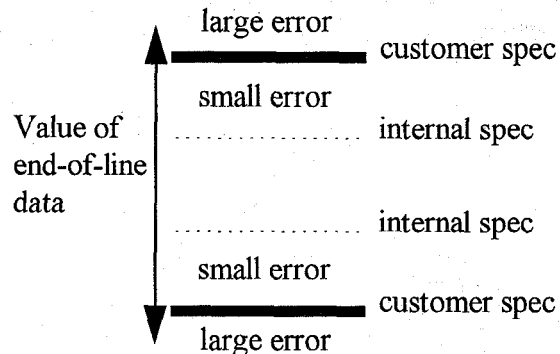


Fig. 4 A rule to decide small and large errors.

The third approach is to include the error direction information into the key node table. Often, the deviation of an end-of-line measurement from its target value may depend on the deviation of a process step's parameter from its target setting. If the correlation between the two deviations can be derived based on process physics, it can be used in the key node table as shown in Fig. 5. Here a

positive sign represents a positive correlation and a minus sign represents a negative correlation. This information also provides the inter-dependence relationship among the end-of-line measurements for a given miss-operated process step. For example, if step_4 has a positive error, then EOL_1 and EOL_3 will have positive errors while EOL_2 has a negative error.

Processing Step	End-of line data			
	EOL_1	EOL_2	EOL_3	EOL_4
Step_1		+		+
Step_2		-		-
Step_3	-			+
Step_4	+	-	+	

Fig. 5 Correlation between end-of-line and process step's deviations.

Although the key node table is redefined and more knowledge is extracted based on the first three approaches, the knowledge is still too empirical for further manipulation. The fourth approach is then to transform the correlation numbers into fuzzy logic's possibility using *S* membership function [9]. The *S* membership function is defined below and is also shown in Fig. 6.

$$\begin{aligned}
 S(x) &= 0 & x &\leq a \\
 &= 2\left(\frac{x-a}{b-a}\right)^2 & a < x &\leq \frac{a+b}{2} \\
 &= 1 - 2\left(\frac{x-b}{b-a}\right)^2 & \frac{a+b}{2} < x &\leq b \\
 &= 1 & b < x &
 \end{aligned} \tag{1}$$

Here, *x* is the correlation value as defined in the key node table, and *a* = 0, *b* = 3. The complete knowledge table after fuzzy modeling is shown in Fig. 7.

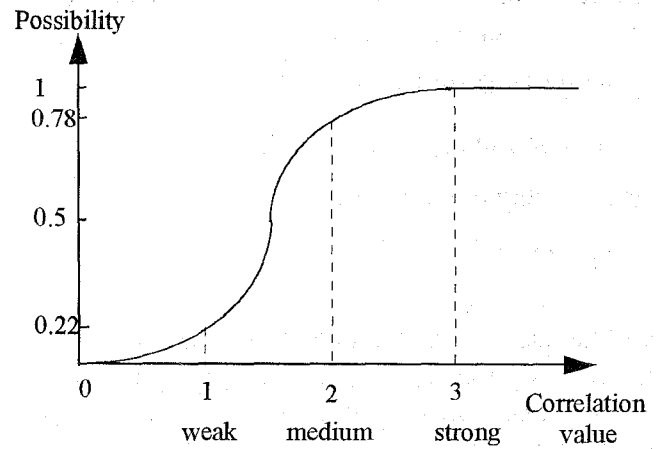


Fig. 6 S membership function.

Processing Step	End-of line data			
	EOL_1	EOL_2	EOL_3	EOL_4
Step_1		0.22, 0.78 +		1, 1 +
Step_2		1, 1 -		0.78, 1 -
Step_3	1, 1 -			0.22, 0.78 +
Step_4	0.78, 1 +	0.78, 0.78 -	1, 1 +	

Fig. 7 Complete knowledge table after fuzzy modeling.

5. Inference Engine

The inference engine computes, by using fuzzy logic operations, the possibility of a process step causing the observed abnormal end-of-line measurements. The inference algorithm is based on comparing the observed abnormal patterns with patterns in the knowledge base (KB). For a given process step, a better match between the two patterns, a higher possibility is assigned to the step.

For a process step *i*, the possibility of a positive fault is given by:

$$P_i^+ = \omega_i A_i = \frac{\sum_{j=1}^n \lambda_{ij}^+ S_{ij}}{\sum_{j=1}^n S_{ij}} \times \left[\frac{1}{n} \sum_{j=1}^n \lambda_{ij}^+ S_{ij} \right] \tag{2}$$

n = number of abnormal end-of-line measurements
 N_i = number of end-of-line items for step i in the KB
 λ_{ij}^+ = cell ij matching factor
 S_{ij} = cell ij possibility in the KB
 A_i = step i weighted sum of possibility
 ω_i = step i matching factor

Notice that the “+” sign represents a positive fault, i.e., the step’s parameter is “higher” than the target value. We can also use equation similar to (2) to calculate the possibility of a step’s negative fault (by using λ^-). The calculation steps are described in detail below and illustrated in Fig. 8.

Step 1: Calculate the cell matching factor λ^+

Assume the step i is positively (+) deviated from its target value, then we define:

$\lambda_{ij}^+ = 1$ if KB cell ij has the same sign
 (+,-) as the sign of the observed
 end-of-line measurement j

$\lambda_{ij}^+ = 0$ other cases

Step 2: Calculate the weighted sum of possibility A

$$A_i = \frac{1}{n} \sum_{j=1}^n \lambda_{ij}^+ S_{ij} \quad (3)$$

Here, S_{ij} must be chosen based on whether end-of-line measurement j is a small error or a large error.

Step 3: Calculate the step matching factor ω

$$\omega_i = \frac{\sum_{j=1}^n \lambda_{ij}^+ S_{ij}}{N_i \sum_{j=1} S_{ij}} \quad (4)$$

Step 4: Calculate each step’s fault possibility P^+

Step 5: Repeat for P^-

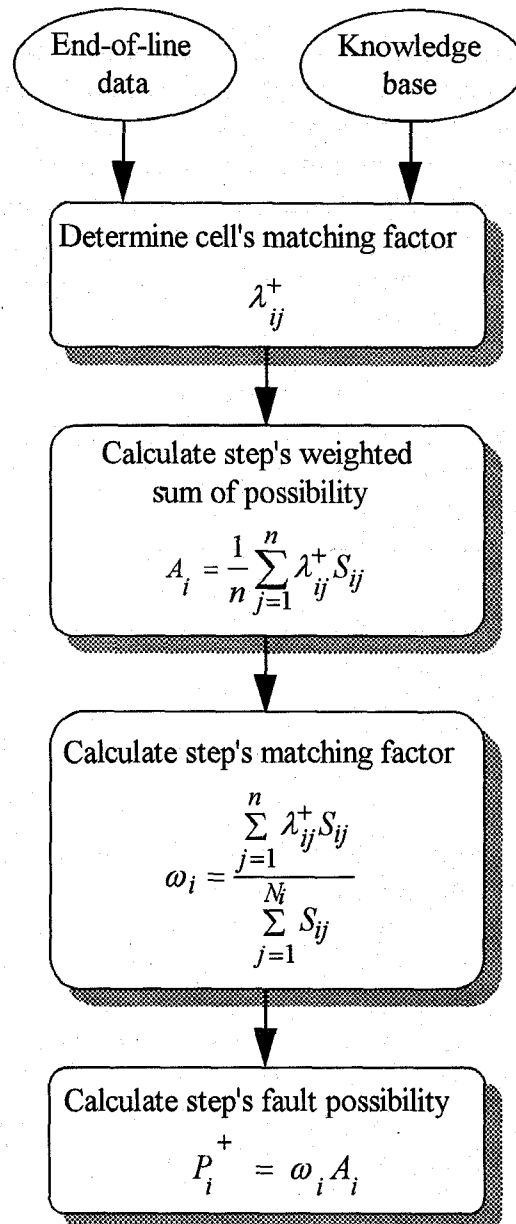


Fig. 8 Inference engine calculation steps.

The purpose of cell matching factor and step matching factor is to evaluate the consistency between the knowledge base (prior information) and the observed end-of-line data.

The cell matching factor evaluates the error direction consistency. The step matching factor evaluates the consistency of the overall error pattern. If the diagnosis is based only on emphasizing measurements that deviate but ignoring those that do not, it might bias the decision.

End-of line data

Processing Step	Vt_NL		Vt_N		Isat_N		BV_N		Rs_N+		Rc_N+	
	1	1	1	1	1	1			1	1	1	1
P+ S/D Photo (Blanket)	+		+		-				+		+	
NLDD-Implant (Dosage)	0.22 0.78		0.22 0.78		0.78 1		0.22 0.78					
N+ S/D Implant (Dosage)	0.22 0.22		0.22 0.22		0.22 0.78				1 1		1 1	
	-		-		+		-		-		-	

Fig. 9 Knowledge base used in the example.

Example

Five end-of-line measurements show abnormal values, which are:

- Vt_N (-) small error
- Isat_N (-) small error
- Rs_N+ (+) large error
- Rc_N+ (+) large error
- Rs_NW (+) small error

Part of the knowledge base is shown in Fig. 9. Based on the information, we want to know which process steps are the causing steps.

Step 1: Calculate the cell matching factor λ^+

	Vt_N	Isat_N	Rs_N+	Rc_N+	Rs_NW
P+ S/D photo :					
$\lambda^+ =$	0	1	1	1	0
NLDD-implant :					
$\lambda^+ =$	1	0	0	0	0
N+ S/D implant :					
$\lambda^+ =$	1	0	0	0	0

Step 2: Calculate the weighted sum of possibility A

P+ S/D photo : $A_1 = (0 \times 1 + 1 \times 1 + 1 \times 1 + 1 \times 1 + 0 \times 0) / 5 = 0.6$
 NLDD-implant : $A_2 = (1 \times 0.22) / 5 = 0.044$
 N+ S/D implant : $A_3 = (1 \times 0.22) / 5 = 0.044$

Step 3: Calculate the step matching factor ω

P+ S/D photo : $\omega_1 = (1+1+1) / (1+1+1+1+1) = 0.6$
 NLDD-implant : $\omega_2 = (0.22) / (0.22+0.22+0.78+0.22) = 0.153$
 N+ S/D implant : $\omega_3 = (0.22) / (0.22+0.22+0.22+1+1) = 0.08$

Step 4: Calculate each step's fault possibility P^+

P+ S/D photo : $P^+_1 = 0.6 \times 0.6 = 0.36$
 NLDD-implant : $P^+_2 = 0.044 \times 0.153 = 0.006$
 N+ S/D implant : $P^+_3 = 0.044 \times 0.08 = 0.004$

Step 5: Calculate each step's fault possibility P^-

P+ S/D photo : $P^-_1 = 0.2 \times 0.2 = 0.04$
 NLDD-implant : $P^-_2 = 0.156 \times 0.54 = 0.08$
 N+ S/D implant : $P^-_3 = 0.444 \times 0.84 = 0.37$

So the three most possible steps are:

- N+ S/D implant (negative error, $P^-_3 = 0.37$)
- P+ S/D photo (positive error, $P^+_1 = 0.36$)
- NLDD-implant (negative error, $P^-_2 = 0.08$)

6. Data Validation

Twenty three test cases are carefully selected from historical lot data by experienced field engineers to validate the proposed diagnosis system and to explore its properties. The metric for validation is defined as follows. If the top (second, third) possible cause of a case identified by the diagnosis system matches the real cause, a score of one (0.5, 0.25) is given to the test case. Our validation focuses on the inference engine and knowledge base. Improvements in these two modules are carried out along with the validation by individual cases.

The validation starts with the baseline diagnosis system, where the inference engine does not include consideration of step matching factor (ω_i in Eq. 4) and the knowledge base of version 1.0 is used. Validation result of the first 14 test cases is depicted

by the solid line in Fig. 10, where the system diagnoses the actual causes as top possible causes in half of the cases and the average score is 0.55. Analysis of the misdiagnosed cases (cases 6, 7, 9, 11, 12 and 14) indicate two types of deficiency in the baseline system, one suggesting inclusion of the step matching factor into the inference engine and the other suggesting enrichments of the knowledge base. Details are described as follows.

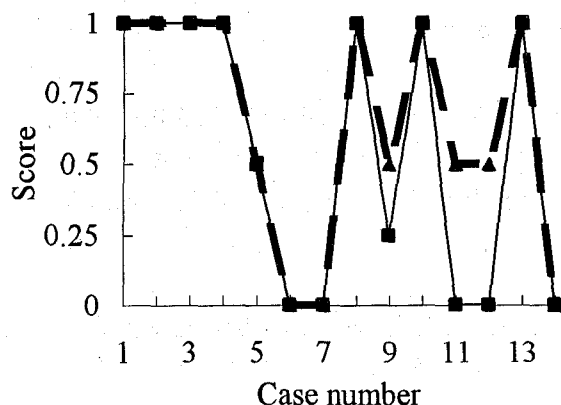


Fig. 10 Validation results for knowledge base version 1.0 and inference engine without step matching factor (solid line) and with step matching factor (dash line).

Inference engine

In the original inference engine, the possibility of a process step being the cause of fault is calculated using the weighted sum of possibility (A_i in Eq. 3).

With this method, the step's fault possibility is based only on the end-of-line measurements that deviate from their respective target values. The prior information about the error pattern, i.e., which measurements should deviate in what way when a process step goes wrong, is not utilized. In other words, the diagnosis is biased by overemphasizing on measurements that deviate but ignoring those that do not. Misdiagnosis in cases 9, 11, and 12 is due to such a deficiency.

Specifically, in case 12, the $Isat_N$ value is seriously higher than target (positive and large error) and Vt_N is slightly low. By merely looking at the deviations and using weighted sum of possibility A , we deduce that Vt_imp is the most possible cause. The true cause $Pwell_imp$, however, has a lower A than that of Vt_imp .

The calculated values are:

$$Vt_imp: A = 1.0$$

$$Pwell_imp: A = 0.75$$

However, if Vt_imp is really the cause, we should observe abnormal values for BV_N , Vt_P , $Isat_P$ and BV_P (Fig. 11). These motivate the addition of the step matching factor into consideration as described in section 5. The newly calculated possibilities are:

$$Pwell_imp: P = \omega A = 0.56 \times 0.75 = 0.42$$

$$Vt_imp: P = \omega A = 0.36 \times 1.0 = 0.36$$

Diagnosis results of the revised inference engine are depicted by the dash line in Fig. 10, where improvements in cases 9, 11 and 12 are obvious and there is no loss of diagnosability in other cases; the average score goes up to 0.64.

Knowledge base

Misdiagnosis of cases 6, 7 and 14 does not improve with the revised inference engine because of insufficient or inaccurate knowledge base. For example, the $P+_blanket$ step is the true cause of case 6 but it is not in the knowledge base of version 1.0.

To overcome the above deficiencies, more detailed process steps are added to the knowledge base, which increases from 34 to 132 in the validating process. Misdiagnosis due to missing steps in the knowledge base is therefore largely eliminated. Diagnosis result obtained by using the revised inference engine and knowledge base version 2.0 over 23 cases are given in Fig. 12, where the first 14 cases are the same cases as those in Fig. 10. Significant improvements in cases 6, 7 and a small improvement in case 14 can be observed while at the price of deterioration in cases 5, 8, 9, 11 and 12. The average score of the first 14 cases slightly goes up from 0.64 to 0.66 and is about 0.68 overall.

To sum up, in 13 cases the real cause is identified as the top cause, in 4 cases as the second and in 3 cases as the third. Out of the 23 studied cases, version 2.0 of the system identifies the real causes as the top three causes in 20 cases.

Processing step	End-of-line data						
	Vt_N	Isat_N	BV_N	VTF_PoN	Vt_P	Isat_P	BV_P
Pwell_imp	+	-	+	+			
Vt_imp	+	-	+		-	+	-

Fig. 11 Knowledge table for case 12.

The deterioration is mainly due to the inaccurate parameters in the knowledge base. For example, in case 8 step Poly_photo/etch (small poly gate length) can be identified as the cause of symptom pattern of low Vt_N and large Isat_N by applying two physical rules,

- $Vt \propto L$ (L is poly gate length)
- $Isat \propto 1/L$

Our diagnosis system only ranks Poly_photo/etch as 2nd possible because the weighting is inaccurate which makes step NLDD the top possible and is inconsistent with the reality. Parameters of future knowledge base should be adjusted to better reflect engineers' knowledge about the basic physical relationship.

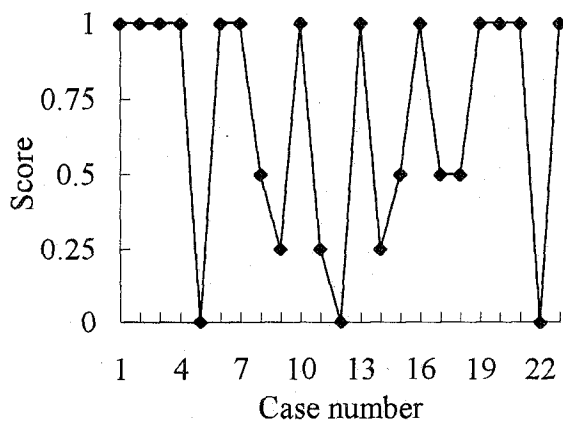


Fig. 12 Validation results for knowledge base version 2.0 and inference engine with step matching factor.

7. Conclusions

In this paper, we describe an intelligent diagnosis system for a first-cut end-of-line diagnosis function. The system uses information of abnormal end-of-line data and fuzzy logic-based knowledge base to calculate each major step's fault possibility.

The intelligent diagnosis system has been validated against 23 real fab cases. Results show that version

2.0 of the system identifies the real causes as the top three causes in 20 cases. Our analysis indicates that the inference engine is robust but the knowledge base is insufficient. A short-term improvement strategy is to periodically update the knowledge base by field engineers based on lessons learned from the case study. Future long-term improvement plan might use neural network to perform the self-learning.

References

- [1] K. Saito, M. Sakaue, T. Okubo, and K. Minegishi, "Application of Statistical Analysis to Determine the Priority for Improving LSI Technology," IEEE Transactions on Semiconductor Manufacturing, vol. 5, no. 1, February 1992.
- [2] S. Saxena and A. Unruh, "Diagnosis of Semiconductor Manufacturing Equipment and Processes," IEEE Transactions on Semiconductor Manufacturing, vol. 7, no. 2, May 1994.
- [3] K. Funakoshi and K. Mizuno, "A Rule-Based Process Flow Validation System with Macroscopic Process Simulation," IEEE Transactions on Semiconductor Manufacturing, vol. 3, no. 4, February 1990.
- [4] S. B. Dolins, A. Srivastava and B. E. Flinchbaugh, "Monitoring and Diagnosis of Plasma Etch Processes," IEEE Transactions on Semiconductor Manufacturing, vol. 1, no. 1, February 1988.
- [5] R.J. Kuo, "Intelligent Diagnosis for Turbine Blade Faults Using Artificial Neural Networks and Fuzzy Logic," Engineering Applications of Artificial Intelligence, vol. 8, no. 1, 1995.
- [6] N. Kartan, I. Flood, and T. Tongthong, "Integrating Knowledge-based Systems and Artificial Neural Networks for Engineering," Artificial Intelligence for Engineering Design and Manufacturing, vol. 9, 1995.
- [7] H.C. Fu and J.J. Shann, "A Fuzzy Neural Network for Knowledge Learning," International Journal of Neural Systems, vol. 5, no. 1, March 1994.
- [8] R.K. Ramamurthi, "Self-Learning Fuzzy Logic System for *In Situ*, In-Process Diagnostics of Mass Flow Controller," IEEE Transactions on Semiconductor Manufacturing, vol. 7, no. 1, February 1994.
- [9] D. Driankov, H. Hellendoorn, M. Reinfrank, *An Introduction to Fuzzy Control*, Springer, 1993.