Standard Least Squares Method for Identifying Problem Tools in a Manufacturing Environment

Edward L. Hutchins
Sustaining Engineering Manager, Materials Division
Cree Inc.

Abstract: We describe an analytical method to identify problem tools in a manufacturing environment. Using JMP, we visualized a yield problem, collected a relevant data set, and created a Standard Least Squares model to identify the tools contributing most to the problem. Once identified, further analysis in JMP gave insight into the underlying root cause and suggested the proper corrective action. On numerous occasions, we have found this strategy to be an effective and efficient method to identify problem tools.

From a sustaining engineering perspective, there are two types of problems that arise in manufacturing. Discrete, obvious, and high-rate problems are usually solved on the manufacturing floor. What does the defect look like? Where is the defect being caught? Answer those questions, apply some process insight, and the source can be quickly found. While these types of investigations must always happen, they can turn up empty when the problem occurs with low frequency and cannot be directly characterized or imaged. The problem may not even be apparent until some downstream operation or department. In situations such as these, we have found JMP to be a highly effective tool for visualizing the problem, identifying its source, and directing us to the best resolution.

Such was the situation when we were faced with the mysterious "Defect_A" in the summer of 2006. Of course, that is not the real name of the defect. As with other details of our process, I will use aliases to protect the details of our intellectual property, for the knowledge we gained during this problem, investigation, and solution remains relevant to our LED manufacturing process at CREE today. Suffice it to say, Defect _A was a bad defect to have. It was known to cause LED failures; indeed, when present, it caused losses of thousands of LED die per wafer. These losses, though, were only realized in downstream characterizations after several subsequent and costly processing steps had been performed. Once evident, the defect was found on a sizeable fraction of our production wafers, and its pattern of occurrence within a wafer strongly suggested that its origin was in our upstream department. Concerned that significant effort was being expended on these wafers only to reveal this defect, our customers in the downstream departments reported the problem to us and requested that we find a solution.

Our first challenge was to describe the extent to which Defect_A was observed. When we produced the affected wafers, our internal characterizations revealed nothing abnormal, so we had no direct process control data by which we could gauge the scale of the problem. Instead, we worked backwards from the characterization data collected by our downstream customers. On each wafer, they had been counting the incidence of Defect_A at several predefined locations and storing that information in a SQL database. It was then a simple matter to create an ODBC connection to this database and import their

characterization data into JMP. In JMP, we then summarized the count of Defect_A at the multiple locations to generate a sum of Defect_A per wafer. Further summarization by our downstream customer's processing date allowed us to create a u-chart of this defect and visualize its increase (Figure 1).

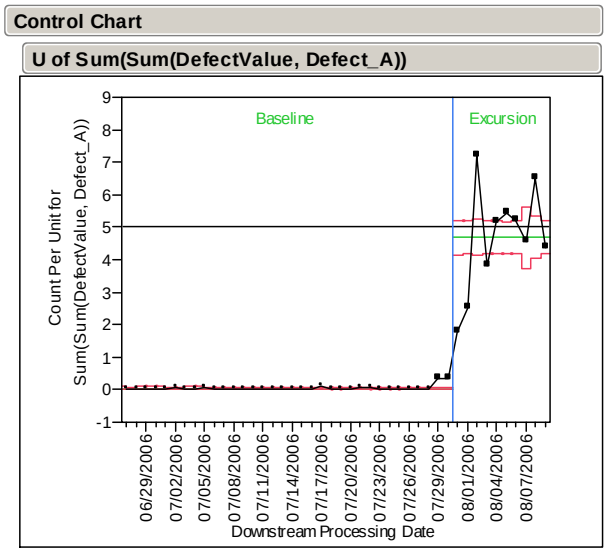


Figure 1: U-chart showing the increase in average Defect_A count per wafer by downstream processing date. The upper specification for this defect is 5 per wafer. Relative to the control limits calculated by JMP for our baseline process, out-of-control data points are identified (in bold).

The u-chart showed a clear and significant increase in the incidence of Defect_A, beginning when it exceeded the baseline control limits on July 29 and continuing to the point where the average count per wafer was at or above the upper specification limit of 5 per wafer. While our customers agreed with our determination of when the problem began, they did not report failure rates as high as the u-chart suggested. To dig into this inconsistency, we used JMP's lasso tool to select the out-of-control data points and created a subset of data from which we plotted the distribution of the sum of Defect_A per wafer (Figure 2).

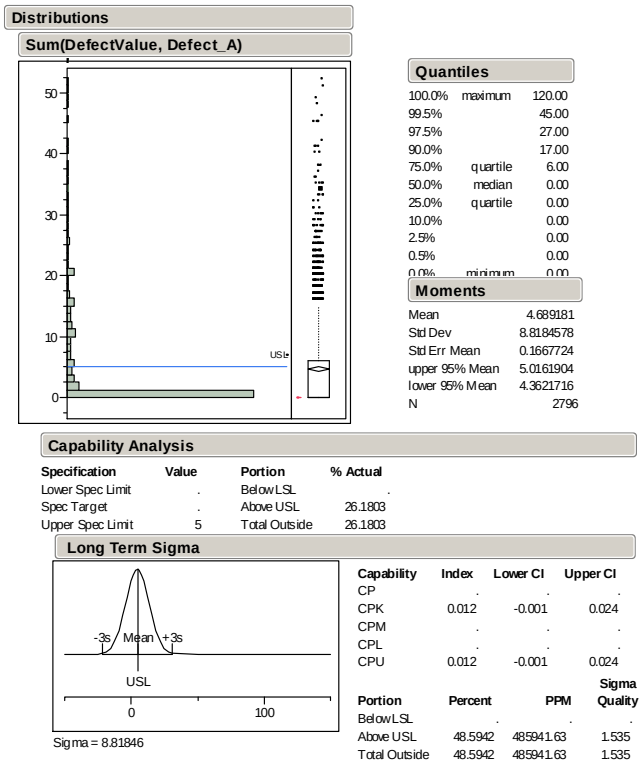
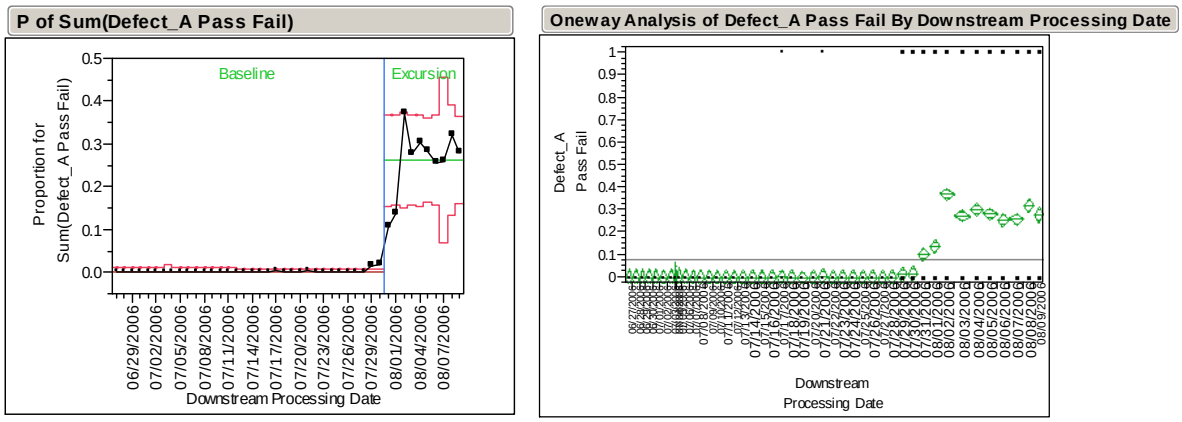


Figure 2: Distribution and statistics describing the sum of Defect_A observed on the wafers taken during the excursion period. The Capability Analysis section shows that 26% of wafers exceed the upper specification limit (USL) for this defect.

Clearly, this distribution is not normal, having few wafers with very high counts of Defect_A and the majority with little or none of it. The average value of Defect_A is therefore an inappropriate statistic for describing this distribution and the problem it represents. The capability analysis does a better job, showing that in this sample of wafers, 26% exceeded the upper specification limit. Since this is a quantification that has direct relevance to our downstream customers and to management, we began plotting this problem as a fraction of wafers that fail the specification.

We created a formula in JMP to compare the sum of Defect_A per wafer to the specification and to assign a pass (0) or fail (1) value to each wafer. Then, summarizing by downstream processing date, we generated a p-chart that accurately showed the fraction of wafers that failed the specification per day (Figure 3a). The Central Limit Theorem allows for a simpler analysis without the need to summarize by date: on account of the large number of wafers we process each day, a oneway analysis of the individual pass-fail values and the addition of Mean Diamonds allowed us to see statistically significant changes in failure rate from one day to the next (Figure 3b).



Figures 3a (left) and 3b (right): The failure rate of Defect_A versus processing date, depicted both as a p-chart and as a oneway analysis of the average failure rate per day (0 = pass, 1 = fail). The addition of Means Diamonds to the oneway analysis shows significant changes from one day to the next.

After using these tools to accurately define the scale of the problem, our next step was to figure out what caused its increase. One of our first observations was that Defect_A was at a zero level for a long period of time prior to the onset of the excursion. For a problem that eventually affected such a large fraction of wafers, it seemed odd that Defect_A was hardly seen before. Suspecting a gauge issue, we talked to the inspectors in the downstream department about Defect_A, how they characterize it, and how long they had been seeing it. These conversations turned up an important detail. Inspectors were re-trained to look for this defect in late July, after the first few wafers with high levels of Defect_A were found. With renewed focus, more wafers with Defect_A were found in subsequent days. However, production wafers from previous days had already moved on, so it was not possible to retroactively apply the same inspection rigor to them. Thus, we could not be sure if the sharp onset of the problem was due to a specific production event or to the more rigorous inspection protocol.

Since this uncertainty would cloud any analysis that included data from older wafers, we decided instead to limit the investigation to data collected with the tightened inspection protocol. These wafers were all inspected to the same standard, so comparisons between them would be relevant. For each of these wafers, we retrieved into JMP via ODBC their full processing history through our department. This included such details as which production tool each wafer went through at each processing step, the time stamp for that processing, and parametric data for the conditions under which the wafer was processed. Wafers that were processed through low-capacity or infrequently-used sister tools at a given step were excluded, both because the small wafer volumes through these tools could not explain the scale of the problem and because we did not want the error inherent to small sample sizes to distort the result of the analysis.

Our experience was that production problems typically don't occur across sister tools at the same processing step as long as those sister tools are truly unique; for instance, not sharing any common supplies of chemicals. It would be a tedious affair, though, to examine the large number of p-charts or oneway analyses to look for differences between the sister tools and find the one tool that changed: for each step, for each sister tool, by each processing date and condition would easily be hundreds of charts. This method could also lead to inaccurate or incomplete conclusions, as it would not capture the effect of the interactions between tools at different production steps. A better method is to analyze this

data as a model in JMP. Each production step can be treated as a factor, with the sister tools used at each step acting as the values for that factor. Critically, the model would also allow us to examine the effect of interactions between tools at different production steps.

We fit a Standard Least Squares model to the sample of Defect_A observed with the improved inspection protocol, with the emphasis on Effect Screening (Figure 4a). At this point, we looked only at main factors, not interactions, because this historical data set could not support the huge possible number of all interactions between all factors. The RSq of this model was poor, only 0.31, largely because of the substantial non-normality of the data set. To improve the fit, we calculated a Box-Cox Transformation, with the best chosen by JMP, and then modeled the transformed data (Figure 4c). (Note that we had to add a constant, 1, to our observed levels of Defect_A to apply this transformation since some wafers had no observed Defect_A.) The Rsq increased to 0.37. Even though the model predicted less than 60% of the Defect_A variation in our data set, it was a good enough fit for our purposes, which was to differentiate the factors that were significant from those that were not. JMP shows this in the Effect Tests section of the Regression Report (Figure 5a). The processing tools of Steps 01, 02, 04, 08, and 10 had high P-values, so they were excluded and the model ran again. Doing so reduced the RSq value slightly but improved the resolution to the remaining factors. This process was then repeated until only the significant factors with P-values less than 0.05 remained (Figure 5b).

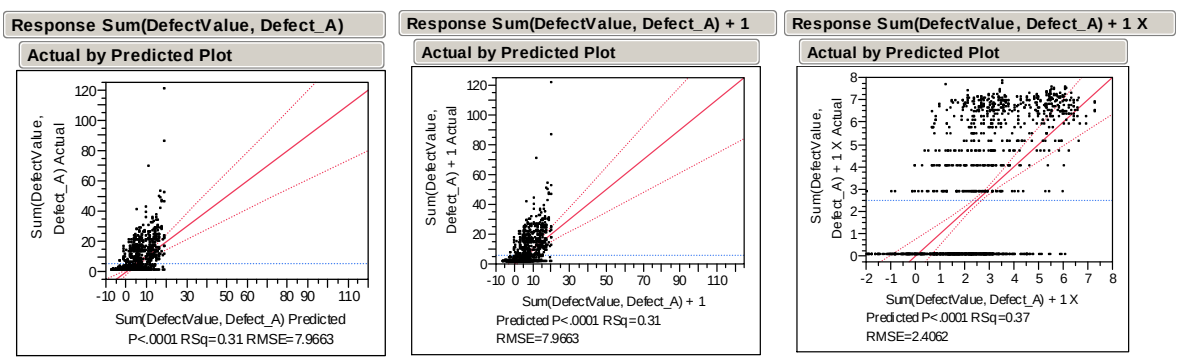


Figure 4a (left), 4b (center), and 4c (right): Models fit to the original (left) and transformed (right) Defect_A data. In order to perform the Box-Cox Transformation, a constant of 1 had to be added to the observed Defect_A values. The center plot demonstrates that this addition has no impact on the results of the least-squares fit. These first models include all factors (all processing steps) and all levels (sister tools) within each factor, but no interactions.

Effect Tests

Source	Nparm	DF	Sum of Squares	F Ratio	Prob > F
Tool - Step 01	2	2	34.5839	2.9867	0.0508
Tool - Step 02	3	3	37.7489	2.1733	0.0894
Tool - Step 03	4	4	74.5402	3.2187	0.0122 *
Tool - Step 04	2	2	33.7398	2.9138	0.0546
Tool - Step 05	1	1	37.4135	6.4621	0.0111 *
Tool - Step 06	5	5	144.2595	4.9833	0.0002 *
Tool - Step 07	3	3	98.5552	5.6742	0.0007 *
Tool - Step 08	1	1	17.0000	2.9362	0.0868
Tool - Step 09	2	2	2055.8088	177.5398	<.0001 *
Tool - Step 10	1	1	6.6355	1.1461	0.2846
Tool - Step 11	31	31	1892.4269	10.5439	<.0001 *

Effect Tests

Source	Nparm	DF	Sum of Squares	F Ratio	Prob > F
Tool - Step 03	4	4	248.2096	10.4597	<.0001 *
Tool - Step 05	1	1	29.0869	4.9030	0.0269 *
Tool - Step 07	3	3	67.2130	3.7765	0.0102 *
Tool - Step 09	2	2	3407.6509	287.2013	<.0001 *
Tool - Step 11	31	31	2619.2446	14.2422	<.0001 *

Figure 5a (left) and 5b (right): The Effect Screening emphasis of the Standard Least Squares model tabulates the significance of each factor in the model. Factors that are statistically significant to the model, with P-values less than 0.05, are identified with asterisks (*). By successively removing the insignificant factors from the original model (left) and re-running the model, we distilled it to only the significant factors (right).

The remaining factors all had statistical significance to the model, but they differ by the magnitude of the effect they had on the result. Tools in process steps that contributed in large and significant ways were probably where the problem laid, while tools in process steps that contributed in small but significant ways were probably not the cause. The Prediction Profiler provided a quick way to understand this behavior and identify the tools that have the greatest contribution to the result (Figure 7).

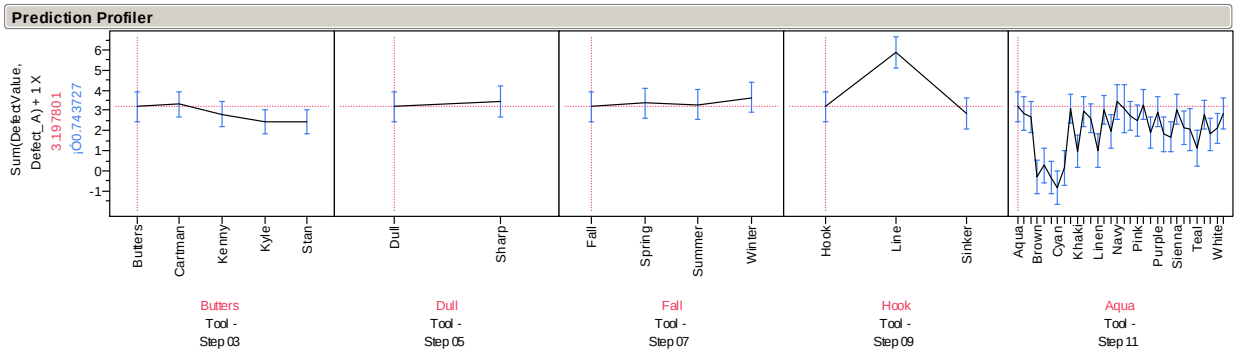


Figure 6: Prediction Profiler for the model that includes only the significant factors, processing steps 03, 05, 07, 09, and 11. For each processing step, the names of the sister tools are given. The profiler makes it clear which tools contribute most to the problem.

While the differences in sister tools at Steps 03, 05, and 07 may be statistically significant, the profiler shows that those differences are small in comparison to those that exist between the sister tools in Steps 09 and 11. Suspecting that the root of the problem was in these two steps, we removed the factors of Steps 03, 05, and 07 from the model. Admittedly, this sacrificed some of the descriptive power of the model, but it greatly reduced the possible number of interactions between the remaining factors. With greater assurance that our data set would include the possible interactions, we added them to the model. Doing so improved the model beyond the original result, increasing the RSq value from 0.37 to 0.42, and gave us much greater insight to the problem. This was again visualized using the Prediction Profiler (Figure 8). When wafers were processed though tool “Sinker” in Step 09, all tools in Step 11 showed very low levels of Defect_A. However, the tools in Step 11 behaved differently on wafers that had been processed through tool “Line” in Step 09. Six of Step 11 tools showed no increase in Defect_A when processing a wafer that had also gone through tool “Line”, but the majority showed a dramatic increase. Once aware of this interaction, we knew two questions we had to answer. My engineering group was tasked with understanding and fixing what was wrong with tool “Line”. A second engineering group was tasked with identifying the robust process conditions in the unaffected Step 11 tools and propagating them to the other sister tools at that step.

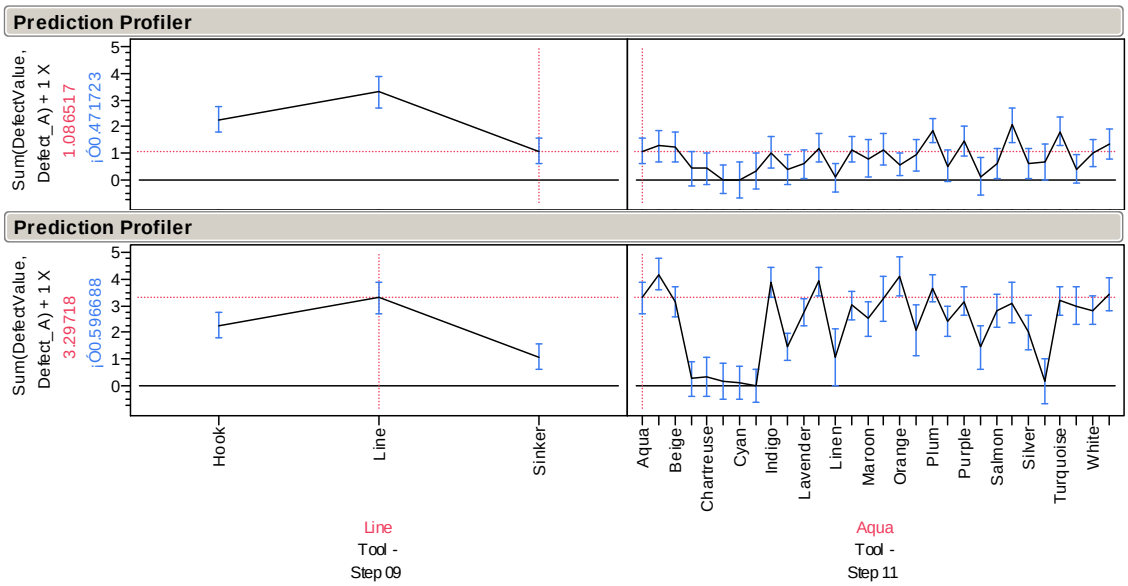


Figure 7: Prediction Profiler for the refined model that includes only the tools in Steps 09 and 11 and the interactions between the tools in those two steps. The top chart shows the predicted incidence of Defect_A on wafers processed through tool "Sinker" in Step 09, while the bottom chart shows the predicted incidence of Defect_A on wafers processed through tool "Line" in Step 09. Note that all but six of the tools in Step 11 are affected by the choice of tools in Step 09.

My engineering group began our investigation into tool "Line" with an examination of the parametric data collected at this step, specifically the position of the wafer in the tool during the process and its relationship to the observed levels of Defect_A. We found considerable variation, and interestingly, different patterns of variation between the three tools (Figure 9). Ideally, our next step would have been to measure the local process conditions at each position for each tool and correlate those measurements to the observed levels of Defect_A. Although the nature of the process and the layout of the tools themselves made such direct, *in-situ*, measurements impossible, we could infer the local conditions by studying wafer attributes we already understood. When we found patterns of Attribute_B that matched up with the patterns of Defect_A failures, and combined that with the knowledge that Attribute_B is driven by subtle concentration variations between the positions in the tool, we knew that these same concentration variations were driving the rate of Defect_A failures. Moreover, we could explain not only why tool "Line" was worse than the other sister tools at Step 09, but why tools "Hook" and "Sinker" showed some Defect_A failures as well.

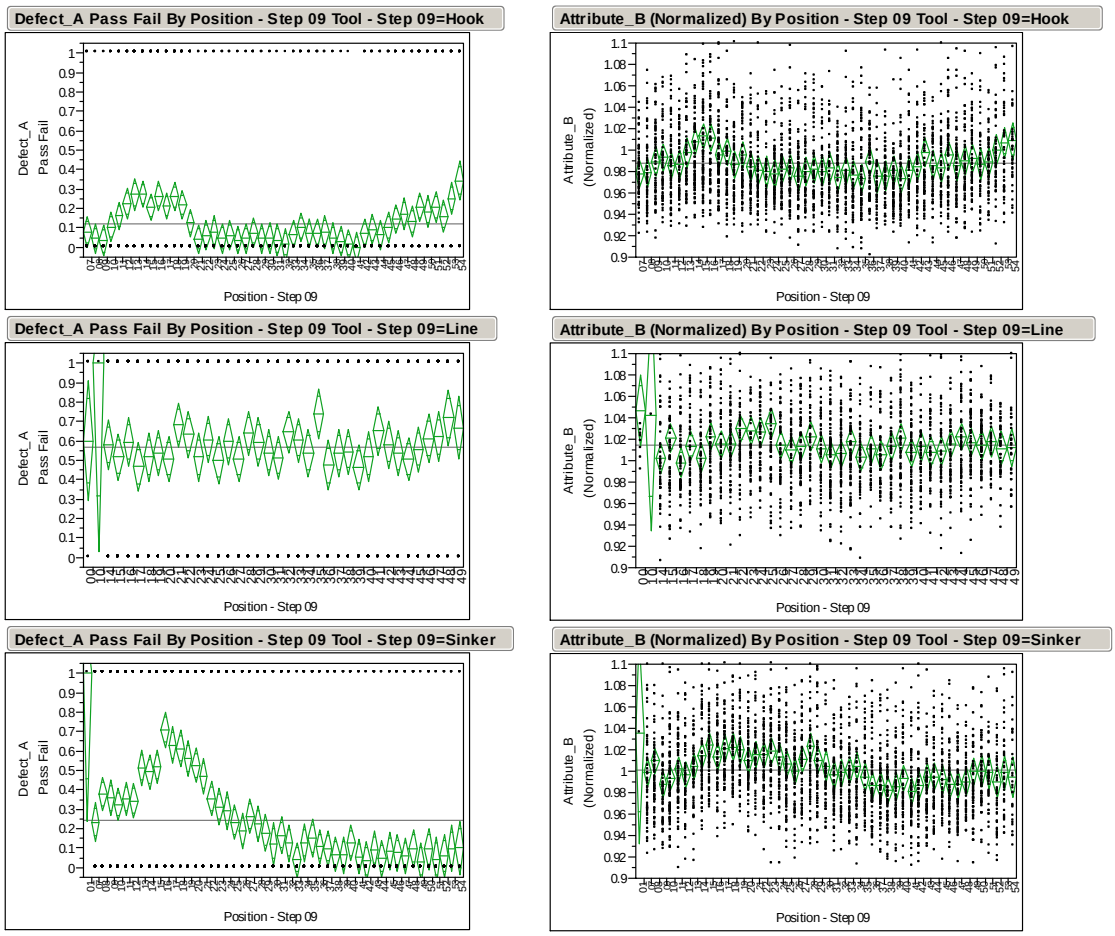


Figure 8: Oneway analyses showing the failure rate of Defect_A (left) and the measured observation of Attribute_B (right) by position within each tool at Step 09. The spatial agreement of these two trends across wafer position in these three tools suggested they had the same underlying root cause.

With the insight that concentration variations were driving the rate of Defect_A failures just as they drove the changes in Attribute_B, we began work on improving our concentration uniformity and control in the tools of Step 09. Concurrently, the other engineering group examined the six tools of Step 11 that showed no interaction to the choice of tool used in Step 09, identified their robust process conditions, and propagated them to the other sister tools at that step. It took time to both understand the problem and make these changes, but by the middle of September, the rate of Defect_A failures had returned to zero (Figure 10).

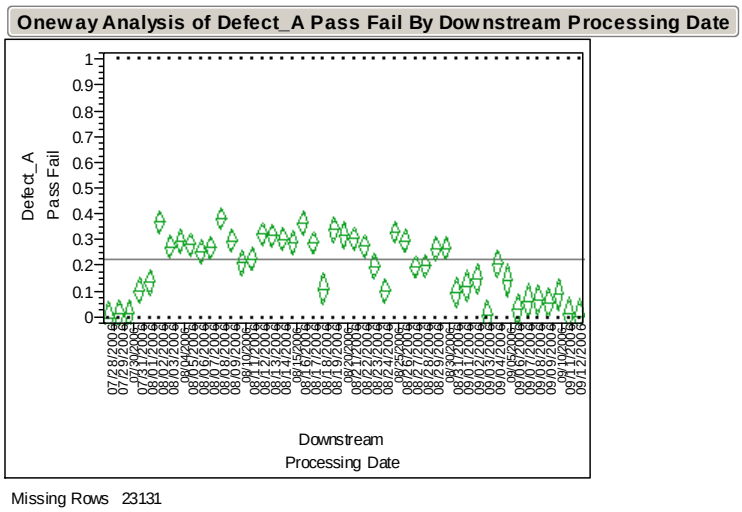


Figure 9: Defect_A failure rate versus downstream processing date. The crisis that began in late July was fully resolved by mid-September.

Several years later, as I reflect upon this excursion and the methods we applied to attack it, I'm shocked by how long it took us to understand and solve this problem. However, I now take this least squares method for granted. At the time of the Defect_A crisis in 2006, we had neither this method nor some of the basic data visibility in place. Indeed, it was this crisis that drove us to develop this method and data infrastructure. Our previous methods were not working; only when we used JMP in this manner did we identify the root cause. We have since been able to control the levels of Defect_A by controlling the concentration uniformity in the tools of Step 09. Moreover, we have been able to apply this least squares method to many other different problems. Each time, it has proven to be an effective and efficient method to identify problem tools.

About the Author: Ed Hutchins is a Sustaining Engineering Manager in the Wafer Operations Division of Cree in Research Triangle Park, NC. Cree is an innovator and manufacturer of semiconductors that enhance the value of LED solid-state lighting, power and communications products by significantly increasing their energy performance. Hutchins focuses on the manufacture of the common element in these diverse products, the silicon carbide (SiC) wafer substrate. Earlier, he worked as a Process Engineer and a Sustaining Engineering Manager in Cree's Advanced Device Epitaxy Group. Before joining Cree, Hutchins worked as a Process Engineer for bulk and epitaxial gallium nitride (GaN) development at ATMI in Danbury, CT. While at ATMI, he became certified in Six Sigma and Design for Six Sigma by Air Academy Associates. He uses JMP extensively, especially for scripting the analyses used in statistical process control. Hutchins is a founding member of the Triangle JMP Users Group. He graduated from Cornell University in 1999 with a bachelor's degree in materials science and engineering.