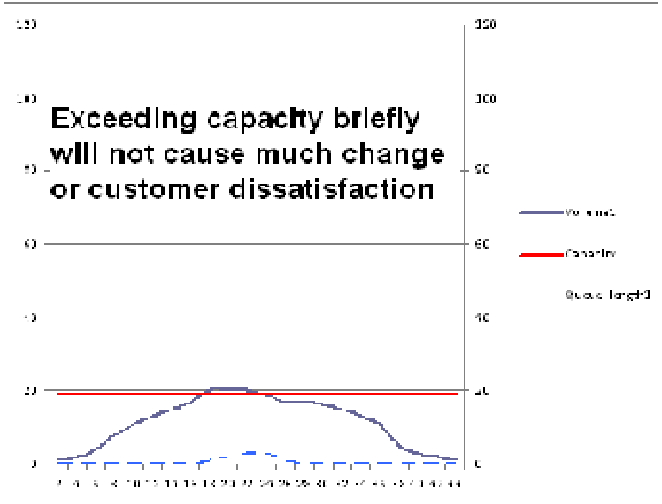
IEEE SW Test Workshop
Semiconductor Wafer Test Workshop

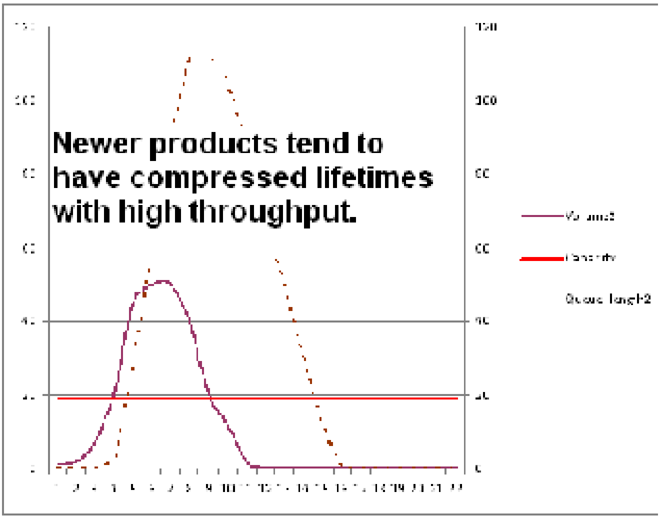
June 12 to 15, 2011
San Diego, CA

PATTERN SEARCHING FOR FAILING DIE

Richard Tuttle
Avago Technologies

Problem Statement


Exceeding capacity briefly will not cause much change or customer dissatisfaction


Newer products tend to have compressed lifetimes with high throughput.

- If throughput rarely exceeds test capacity no action is typically taken with little or no impact to shipment schedules.
- Recent history shows shorter product lifetimes with fast volume ramps to levels exceeding capacity.
- Without action, large queues develop and shipments are delayed.
- Actions must be taken to improve throughput still balanced with sufficient test coverage.

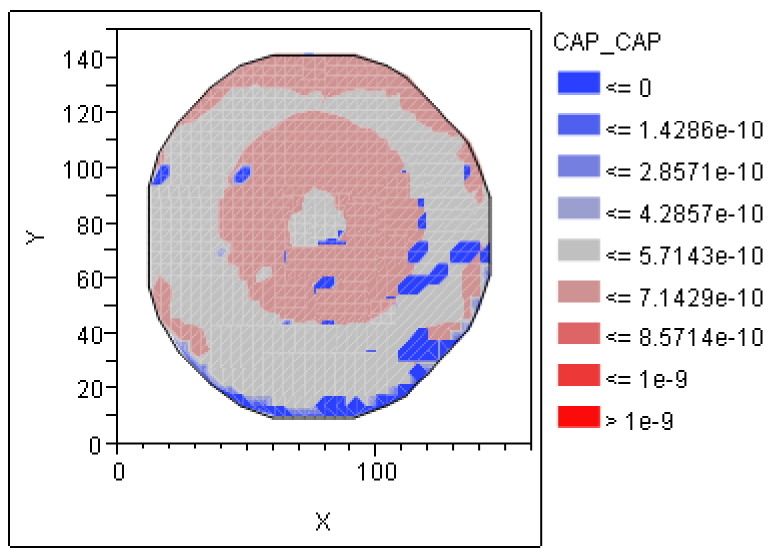
Standard Capacity Improvements

- **Improve Test Capacity**
    - Factors that degrade or overwhelm capacity
  - Frequent new product introductions
  - Fast ramp up in volumes
  - Retest, due to:
    - Unverified spec limits based on simulations
    - New process steps or recipes
- **Standard solutions are: buy more test platforms and add head count (long lead times)**
- **Reduce number of tests**
- **Reduce sampling size(100%→ sampling grid pattern)**

Design of Experiment

- **Phase 1-Implement search algorithm code and run against actual datasets to confirm robustness of pattern search**
  - Wide range of products , manufacturing processes
  - Compare resulting number of failing die from 100% original die test against pattern search failures and analyze/fix causes for missed die.
- **Phase 2-Add pattern search algorithm to direct prober stepping and collect pattern data subset.**
- **All data taken using HP4072 parametric tester with Electroglas 4090 probers.**

Method: Checking for Spatial Patterns

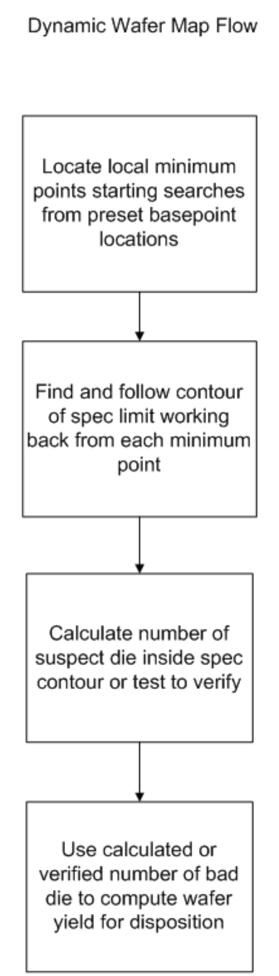


- Patterns are created by distinct process steps during wafer processing.
- When a lot fails the yield engineer will plot single or stacked wafer datasets visually checking for clues to the low yield.
- One parameter, or no more than a few parameters, tell the whole yield story.
- Analysis is done, by hand, after testing is completed.

Pattern Searching

- Pattern searches have evolved and been applied over the last 50 years.
- Heavily used in artificial intelligence applications to find the best solution of continuous functions. (mathematical functions instead of actual datasets are used in this presentation for demonstration).
- The Hooke-Jeeves algorithm is generally considered the simplest and most robust simplex searches
- This project uses a modified version of the H-J algorithm

Wafer level testing flow



Dynamic Wafer Map Flow

Locate local minimum points starting searches from preset basepoint locations

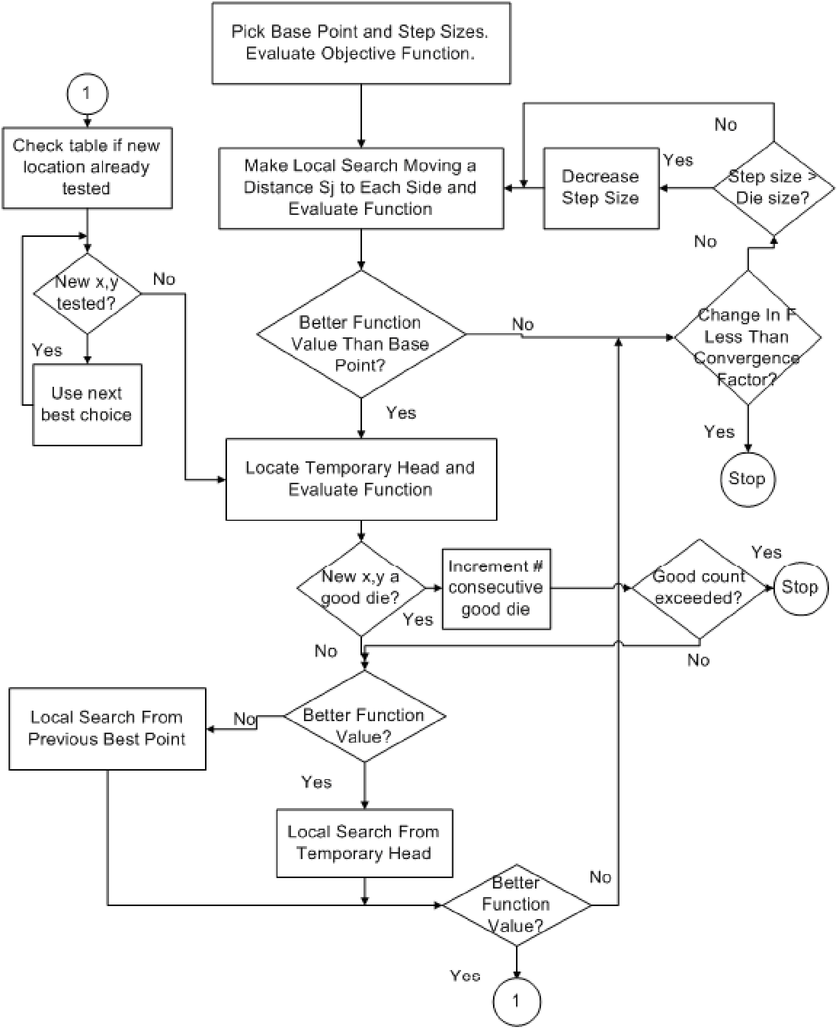
Find and follow contour of spec limit working back from each minimum point

Calculate number of suspect die inside spec contour or test to verify

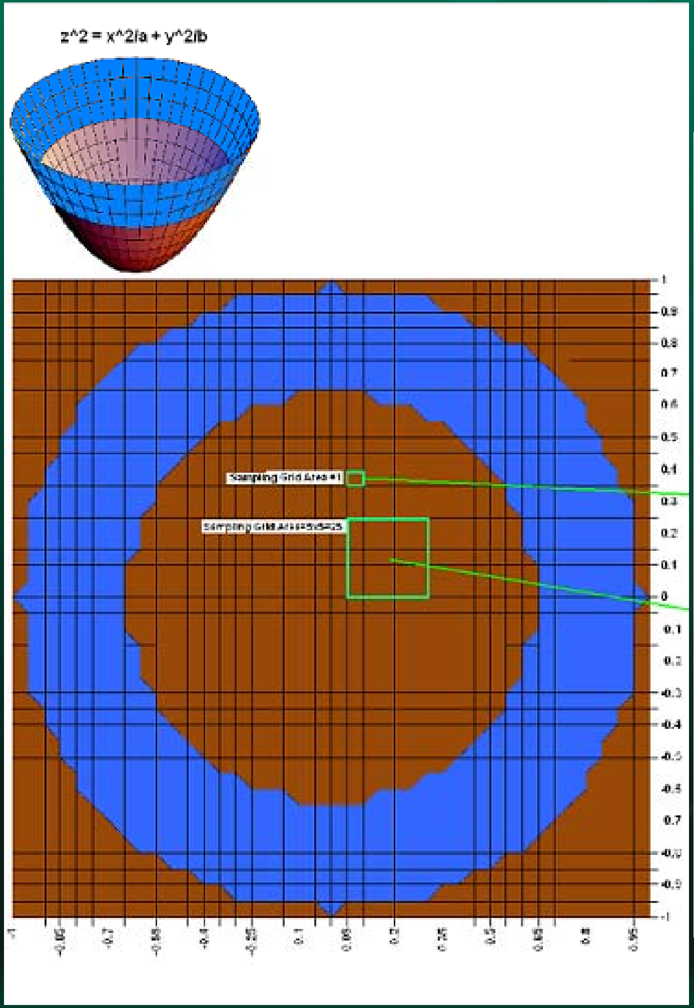
Use calculated or verified number of bad die to compute wafer yield for disposition

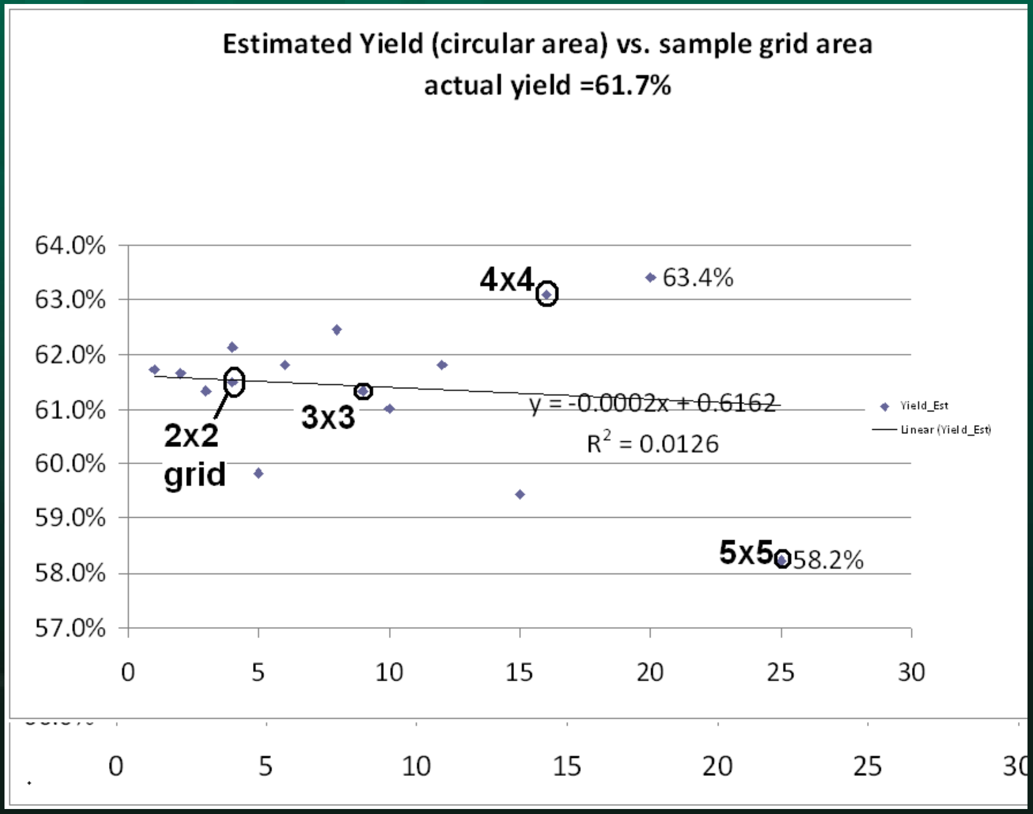
- A standard pattern search is launched from each static base point and the search continues until the local minimum reading is located. (This can also be a weighted combination of parameters).
- The search will start at a local minimum using the transform of y=(LSL-reading)^2 instead of the raw value to force the search to follow the contour of the spec limit surrounding that minima.
- With a sorted list of failing die as a boundary containing the suspect neighborhood, the algorithm may either finish testing die inside the boundary to verify these die fail, or simply count the number of untested die inside and report the combined bad and suspected bad die as a yield estimate.

Modified H-J Search Algorithm

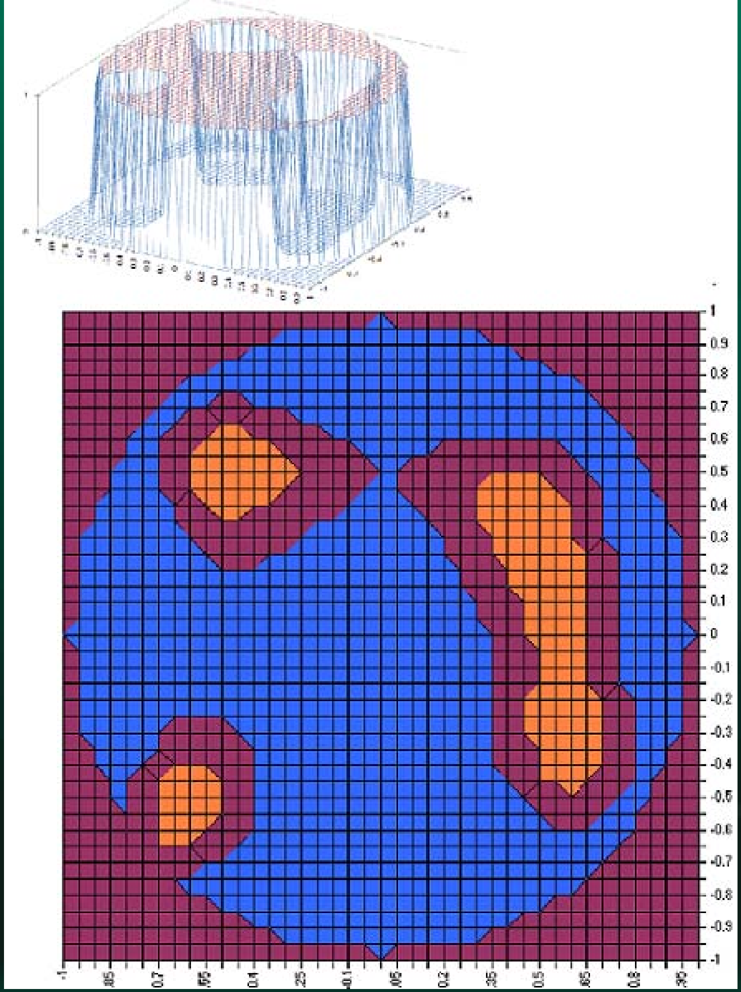


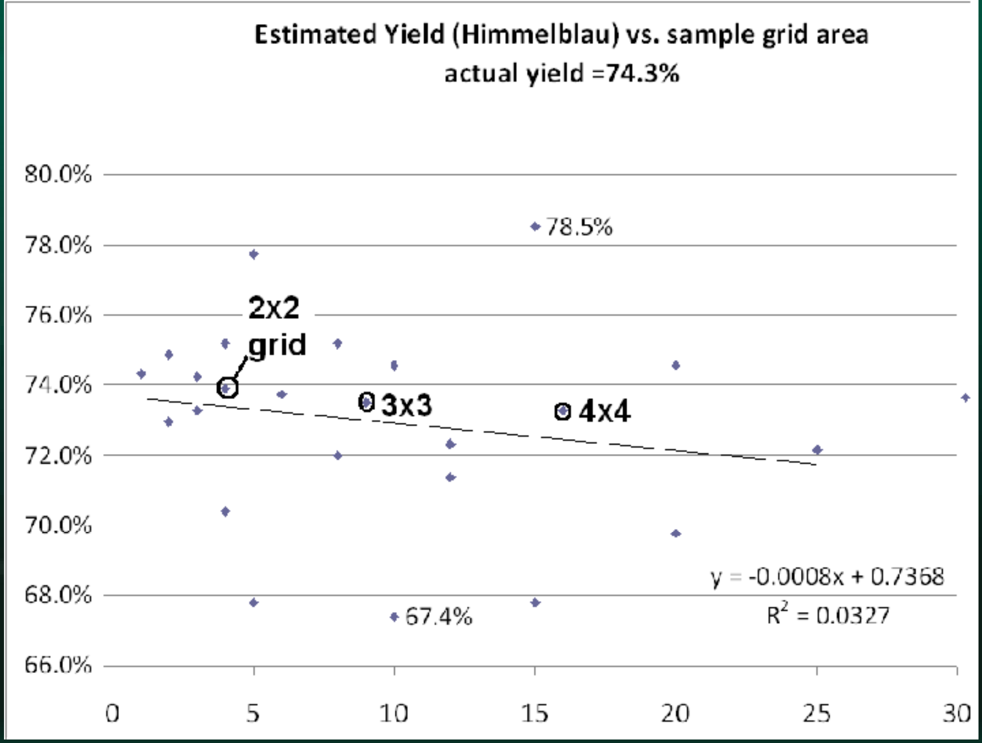
Static Wafer Map w/ symmetric distribution





Static Wafer Map w/ Discontinous distribution

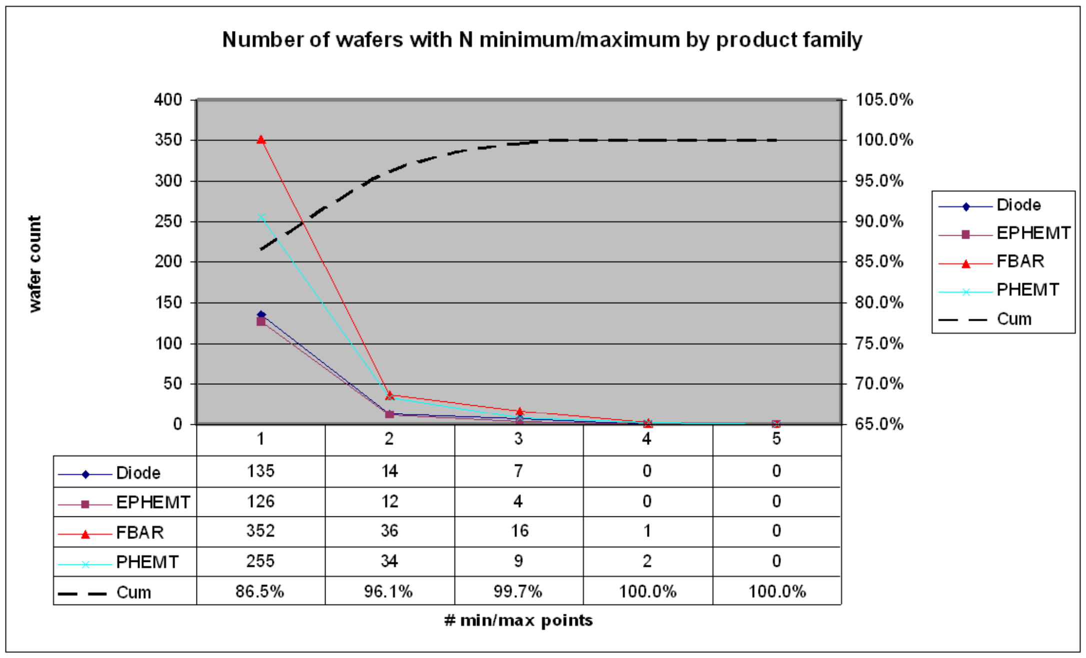




Sample Grid Size Impact on Yield for Two Distributions

- **Simple hyperboloid case with failing die below lower spec**
  - 100% sampling = precise yield (61.7%) but takes most time
  - Using a standard sample grid takes less time, but yield estimate can vary (58.2-63.4%) depending on grid size)
- **Himmelblau function, w/ 4 local minima (two merged together on right side).**
  - 100% sampling = precise yield (74.3%)
  - Using a sample grid takes less time, but yield estimates can vary (67.4-78.5%)
  - Underestimating yield may cause wafer to be scrapped unnecessarily

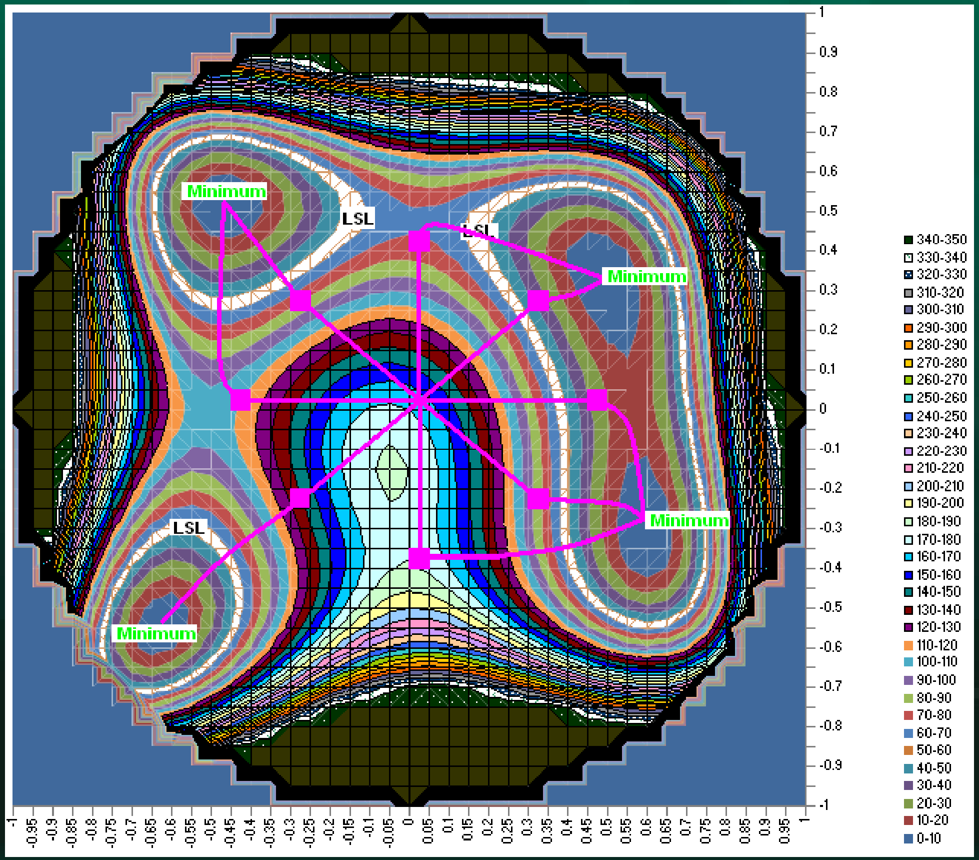
Optimum number of Searches



	1	2	3	4	5
Diode	135	14	7	0	0
EPHEMT	126	12	4	0	0
FBAR	352	36	16	1	0
PHEMT	255	34	9	2	0
Cum	86.5%	96.1%	99.7%	100.0%	100.0%

- **A review of datasets across four product families (diode, caps, PHEMT,HBT confirms:**
  - that most wafers have only one local extreme for a given parameter (87%).
  - Only 9.5% of all wafers have two extremes.
  - Only 3.5% of all wafers have three extremes.
  - Leaving 0.3% with four extremes.
- **Using additional searches beyond 6-8 does not find additional extremes and wastes time.**

Step 1: Locate all Minimum Points



- Place the 6-8 base points at 1/√2 * radius from center to create equal segment sizes includes within that radius as without (base points shown inboard of that radius for clarity).
- The expected search path from each base point to a local minima shows that often more than one search results in locating the same extreme.
- Searches can move in multiples of die steps while locating these minima

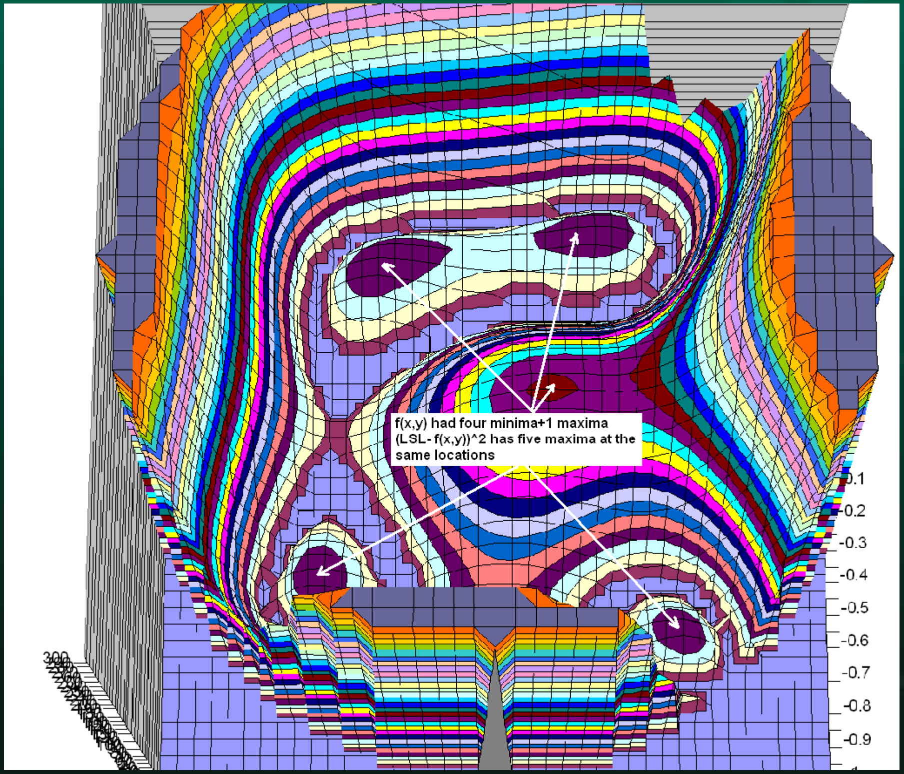
Step 1: Verifying Local Points

- **At this point, a reviewer asked:**
  - "How do you know that you have found an extreme and not wandering around with a contaminated probe?"
- **At the completion of a search, the algorithm returns to a tested die, typically the base point die, and verifies no significant difference in readings. The test aborts and provides an error message to either clean or replace the probe card.**

Step 1: Terminating the Search

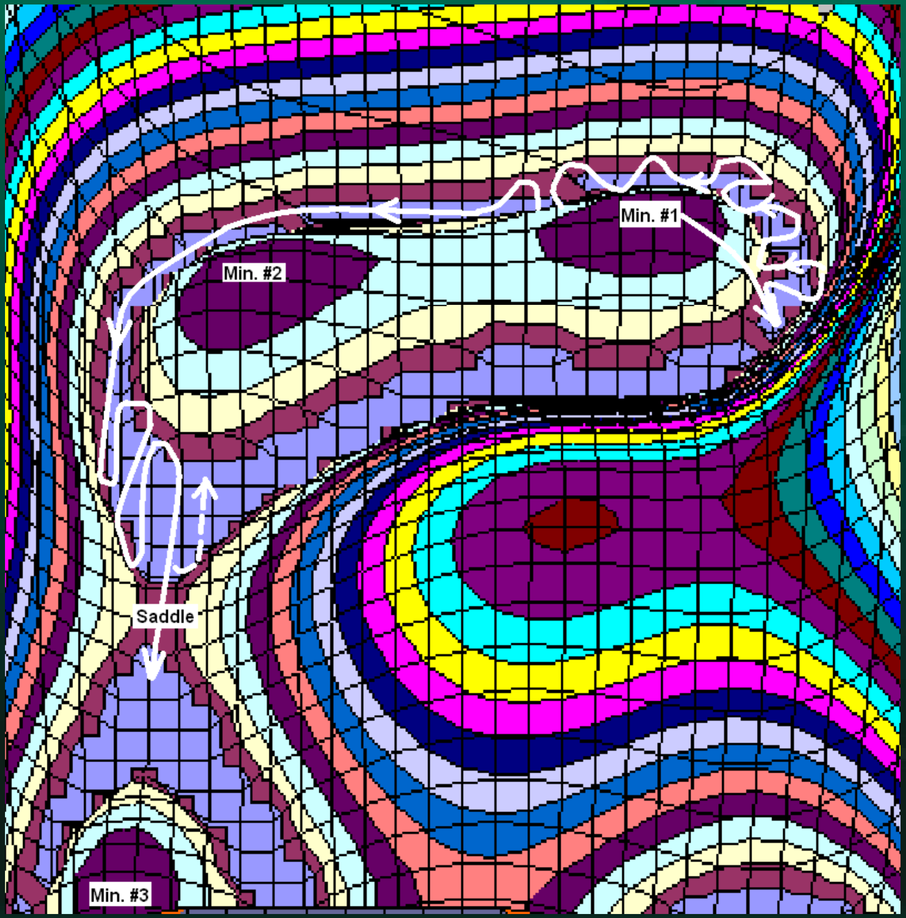
- **A traditional pattern search decreases the step size until the step size becomes smaller than a given tolerance value.**
  - The modified H-J uses: x tol = die x-step and y tol=y-step.
- **A search can loop endlessly in a very flat region while using very small step sizes**
- **Many search methods avoid looping by using a lookup “Tabu” table that records the last n locations and disallows repeating that location until the tested location drops off the list.**
- **This algorithm uses the Tabu table to choose the next best path in Step 2: contour plotting.**

Step 2: Follow Spec Limit Contour



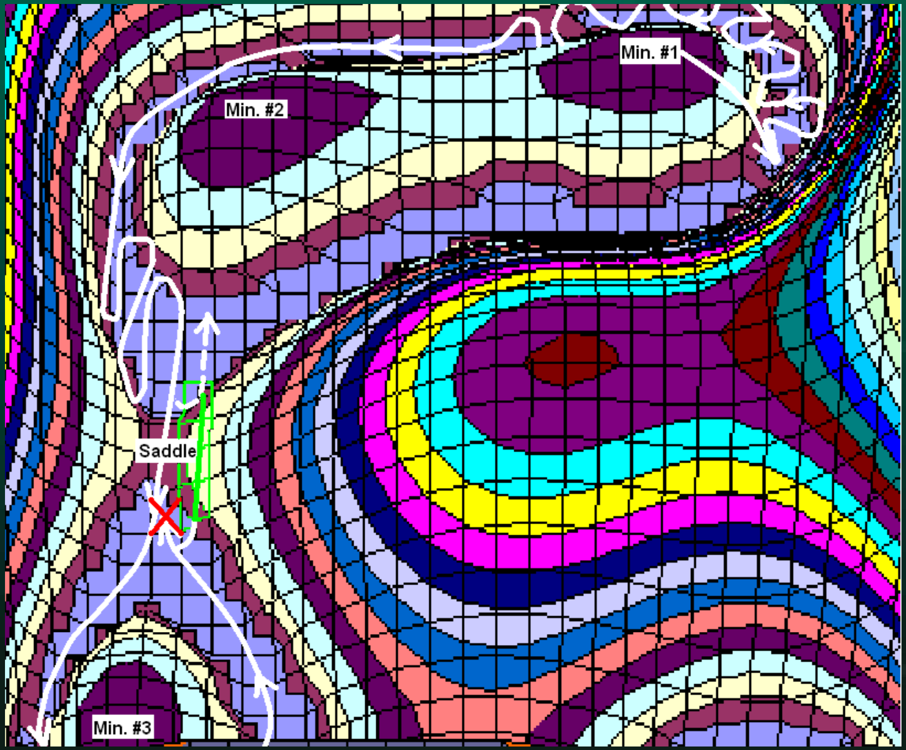
- Contour following using same H-J algorithm on f(x) = (LSL – f(x,y))^2 in place of f(x,y)
- The areas surrounding the four local minima are now transformed into local maximums.
- The lowest regions correspond to the graph floor and outline the spec limit boundaries.

Step 2: Follow Spec Limit Contour



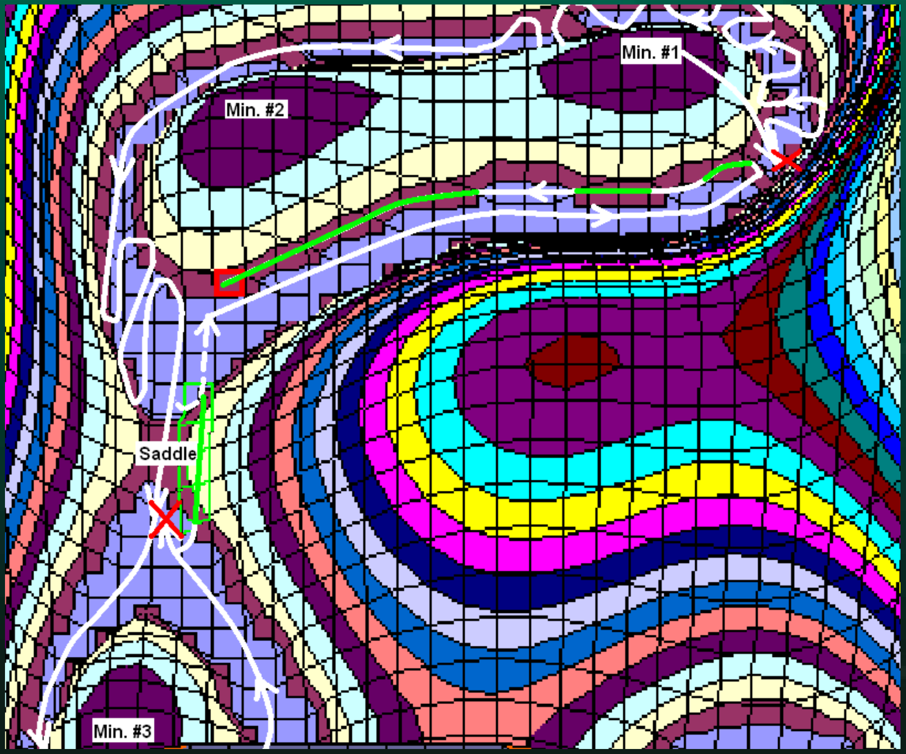
- The H-J algorithm will find the next point minimum in the immediate neighborhood of contiguous die.
- The next minimum may be slightly higher than the present temporary head, but the tabu table will block moving backwards and force the second-best solution.
- The walls of the plot force the search along the valley floor and the tabu table forces the search to move forward causing the search to circle the area inside the LSL boundary in a CCW direction in this example.
- A potential bottleneck due to the saddle is shown on the next slide

Step 2: Follow Spec Limit Contour



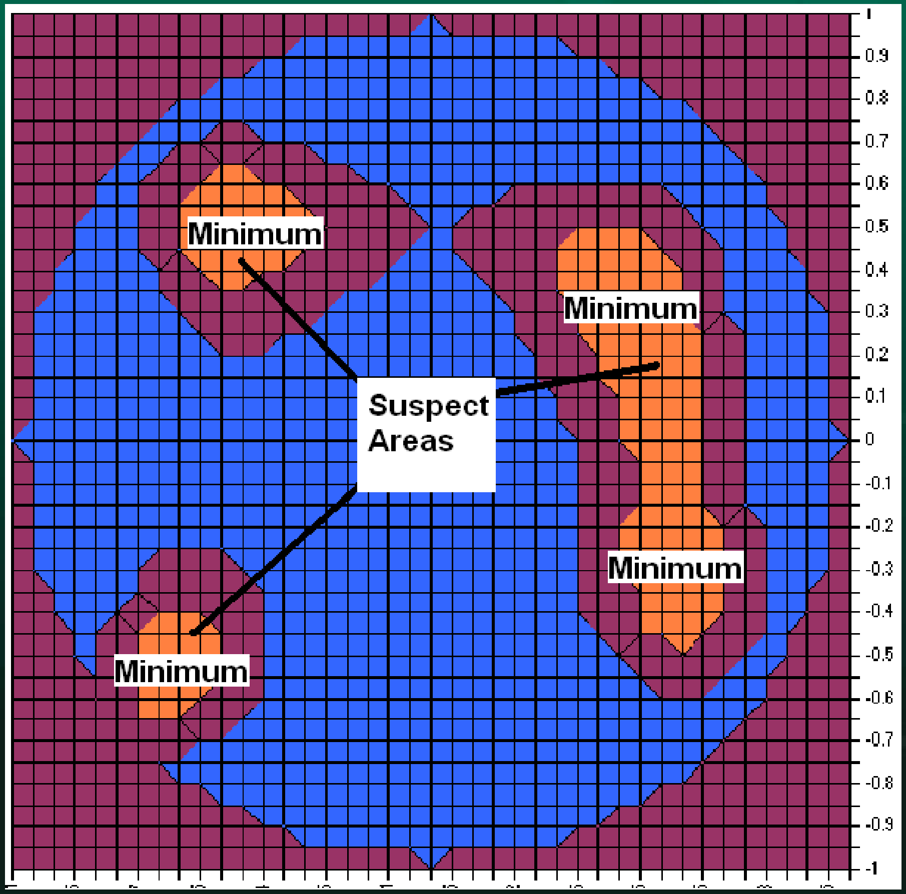
- After the search crosses the saddle point and circumnavigates the LSL contour around Min. #3, it approaches the saddle region but from the opposite direction.
- If the next chosen die coincides with the die indicated by the red cross, the tabu lookup table will disallow this next move The algorithm must briefly follow this second-best path outside the LSL contour.

Step 2: Terminating the Contour Test



- **Once the die on the valley floor have been mapped the algorithm will have to search for the next best choice further up the walls of the function and away from the floor. When this change in elevation exceeds a preset value a counter starts to increment.**
- **The search does not terminate for short excursions as through the saddle, but does terminate on an extended climb up a wall (shown at the termination square.**

Step 3:Processing the Suspect Die



- **The algorithm tested 141 failing die that surround three suspect regions containing an additional 182 die**
- **These 182 die lie between the minimum points and the spec contours and likely also fail.**
- **A separate algorithm can direct probing of the suspect die by filling in the missing ranges.**
- **Or, if time is critical, the suspect die can be assigned as failures.**

Comparison of Methods

- **100% Wafer Test time testing over 1200 die compared to the 141 tested to define the boundaries + 30 tests to locate the minimum points ≈ 175 tests.**
- **The pattern search provides a close yield estimate, especially is the suspect die are confirmed by testing.**
- **Using the same number of tests distributed in a grid arrangement:**
  - tends to miss highlighting the underlying topography and
  - can affect the sample yield by several percent.
- **These results were based on a wafer with only 1250 die maximum used for clarity in demonstrating the pattern search**
- **The ratio of enclosed die to the tested perimeter die increases quickly with wafers containing tens of thousands of die and test time reduction improves.**

Comparison of Methods:Future Work

- The question of “random” defects comes up on review.
- The random ratio sometimes used in yield analysis still shows spatial patterns usually in the form of low yields based on radius or quadrants or swirls. These are still recognizeable to the human eye.
- Even when the defects are truly randomly distributed the pattern search does no worse than a grid test pattern.

Collateral Improvements

- Fewer touchdowns also translates into slower build up of contaminants on probe tips and less in-line cleaning time.
- Probe tip wear out and damage reduced.
- If the worst case die still pass, there is no need to continue testing the remaining die!
- Quickly decide to scrap the wafer if the calculated yield loss combining confirmed and suspected die exceeds the acceptable limit.

Future Work

- Check for patterns that correspond to stepper offsets create by mask defects.
- Adjust search base points on subsequent wafers based on results of the first wafer.

Summary

- Existing pattern search algorithms can be modified slightly to locate the worst spatial features of a wafer quickly.
- The same algorithm can define the boundaries between pass/fail regions and concentrate on verifying the suspect areas.
- Mapping the tested locations will highlight the topography of the failing parameter compared to a grid pattern approach
- Test time reduced while still identifying problem areas.