Health Assessment of Electronic Products using Mahalanobis Distance and Projection Pursuit Analysis

Sachin Kumar, Vasilis Sotiris, and Michael Pecht

Abstract—With increasing complexity in electronic systems there is a need for system level anomaly detection and fault isolation. Anomaly detection based on vector similarity to a training set is used in this paper through two approaches, one the preserves the original information, Mahalanobis Distance (MD), and the other that compresses the data into its principal components, Projection Pursuit Analysis. These methods have been used to detect deviations in system performance from normal operation and for critical parameter isolation in multivariate environments. The study evaluates the detection capability of each approach on a set of test data with known faults against a baseline set of data representative of such "healthy" systems..

Keywords—Mahalanobis distance, Principle components, Projection pursuit, Health assessment, Anomaly.

I. INTRODUCTION

PROGNOSTICS and health management (PHM) is a process of predicting the future reliability of the system by assessing the extent of deviation or degradation of a product from its expected normal operating conditions in a preemptive and opportunistic manner to the anticipation of failures. This can enable continuous, autonomous, real time monitoring of the health conditions of a system by means of embedded or attached sensors with minimum manual intervention to evaluate its actual life-cycle conditions, to determine the advent of failure, and to mitigate system risks. The term "diagnostics" refers to the detection and isolation of faults or failures and "prognostics" refers to the process of predicting a future state (of reliability) of the system based on its current and historic conditions. The aim of failure prognosis is intended to identify and estimate the advancement of fault conditions to system failure.

Quantification of degradation and the progression from faults to failure in electronic products is a challenging task. Gu et. al. [2] identifies six levels of prognostics

S. Kumar is with the Center for Advanced Life Cycle Engineering (CALCE), University of Maryland, College Park, MD 20742 USA (phone: 301-405-5324; fax: 301-314-9269; e-mail: skumar@ calce.umd.edu).

V. Sotiris with the Center for Advanced Life Cycle Engineering (CALCE), University of Maryland, College Park, MD 20742 USA (e-mail: vsotiris@ calce.umd.edu).

M. Pecht with the Center for Advanced Life Cycle Engineering (CALCE), University of Maryland, College Park, MD 20742 USA (e-mail: pecht@ calce.umd.edu).

implementation for electronics, from on–chip packaging to complete products of products. They provided an approach for prognostics implementation at the various levels of electronics, based on failure modes, mechanisms and effects analysis.

Zhang et. al. [3] proposed a model to assess intermittent as well as hard failures. The model is a fusion of two prediction algorithms based on life consumption monitoring and uncertainty adjusted prognostics.

Vichare et. al. [1][4][5] proposed methods for monitoring and recording in-situ temperature loads. This includes methods for embedding data reduction and load parameter extraction algorithms into the sensor modules to enable reduction in on-board storage space, low power consumption, and uninterrupted data collection.

Two approaches for detection and fault isolation based on classification theory are presented in this paper. Both are capable of system level anomaly detection in multivariate, data-rich environments. One methodology uses the Mahalanobis Distance (MD) and the other uses a projection pursuit analysis (PPA) to analyze on-line system data. Both approaches are used to monitor the health of the system and identify onsets and periods of abnormalities. Parameter contribution is performed by both approaches as a means of identifying dominant and potentially faulty parameters [8] [9] [12]. Experiments were performed on notebook computers to generate data and validate the analysis approaches. The experimental details, the algorithmic approach to anomaly detection, and a case study are discussed.

II. METHODOLOGY TO IDENTIFY ABNORMALITIES IN ELECTRONIC PRODUCTS

The Mahalanobis Distance (MD) methodology is a process of distinguishing data groups [6][10]. The MD measures distances in multi-dimensional spaces by considering correlations among parameters. The distance is sensitive to the correlation matrix of the healthy group. The MD values are used to construct a normal operating domain also known as Mahalanobis space to monitor the condition of a multi-dimensional system. Health of a system is defined by several performance parameters. These parameters are standardized and the MDs are calculated for the normal group. These MD values define the Mahalanobis space, which is used as a

reference set for the MD measurement scale.

The parameters collected from a system are denoted as X_i, where i = 1,2,…,m. The observation of the ith parameter on the jth instance is denoted by X_{ij}, where i =1, 2,…, m, and j = 1, 2,…,n. Thus the (m x 1) data vectors for the normal group are denoted by X_j, where j = 1, 2,…, n. Here m is the number of parameters and n is the number of observations. Each individual parameter in each data vector is standardized by subtracting the mean of the parameter and dividing it by the standard deviation. These mean and standard deviation are calculated from the data collected for normal or healthy system. Thus, the standardized values are:

$$Z_{ij} = \frac{\left(X_{ij} - \overline{X}_i\right)}{S_i}, \quad i = 1, 2\dots m, j=1, 2\dots n, \qquad (1)$$

where, $\overline{X}_i = \frac{1}{n}\sum_{j=1}^{n} X_{ij}$ and $S_i = \sqrt{\dfrac{\sum_{j=1}^{n}\left(X_{ij} - \overline{X}_i\right)^2}{(n-1)}}$

Next, the values of the MDs are calculated for the normal items using:

$$MD_j = \frac{1}{m} z_j^T C^{-1} z_j \qquad (2)$$

where, $z_j^T = [z_{1j}, z_{2j}, …, z_{mj}]$ is a vector comprises of z_{ij}, z_j is transpose of z_j^T and C is correlation matrix calculated as:

$$C = \frac{1}{(n-1)}\sum_{j=1}^{n} z_j z_j^T \qquad (3)$$

Next, the test system is considered to determine its health. The MDs for the test system are calculated after the parameters are standardized using the mean and standard deviation for normal-group. The resulting MD values from test system are compared with the MD values of the normal or healthy system to determine test system's health.

A. Significant Parameter Identification Using Mahalanobis Distance Analysis

Critical parameters using MD output can be achieved by identifying parameters that contribute more to the MD value. In other words, the parameters that have significant impact on the MD value should be identified. The effect of individual parameters or combination of parameters on MD output values can be observed through a leave-one-out approach where a reduced set of parameters is used to compute the MD values, in effort to understand the effect of the excluded parameter. The leave-one-out approach can be expressed by an orthogonal array (OA) and a signal-to-noise (S/N) ratio that can be used for quantify the impact of selected combinations. An OA is a design matrix that reflects the presence or absence of parameters involved in the leave-one-out approach for all experiments. Each parameter is assigned to a column and each row represents an experimental run for the leave-one-out approach. Initially all parameters are considered and later with each run one parameter is removed from consideration to observe the effect of that parameter. The MD values corresponding to each leave-one-out run are then used to calculate the S/N ratio values, which are the corresponding responses for each run.

Many different S/N ratios are used in Taguchi's design of experiment. One option mentioned is to use Taguchi's [9] larger-is better S/N ratio, defined as (4)

$$-10\log\left[(1/q)\sum_{j=1}^{q}\left(\frac{1}{MD_j}\right)^2\right] \qquad (4)$$

where, q is the number of observation for an leave-one-out run.

For a given parameter X_i, the average value of the S/N ratio is determined over all runs with that parameter present (S/N_p) and absent (S/N_a). If the difference between (S/N) ratios S/N_p (X_i) - S/N_a (X_i) is positive then this parameter has higher responses when it is part of the system and therefore the parameter X_i, is retained and considered critical.

B. Projection Pursuit Analysis

The Projection Pursuit Analysis uses a Principal Components Analysis (PCA), least squares optimization (LS) and a Singular Value Decomposition (SVD) treatment of the data. PCA is used in a wide array of applications to reduce a large data set to a smaller one while maintaining the majority of the variability present in the original data. It's also very useful in providing compact representation of temporal and spatial correlations in the fields of data being analyzed. PCA facilitates a multivariate statistical control to detect when abnormal processes exist and can isolate the source of the process abnormalities down to the component level.

Two statistical indices, the Hotelling Squared (T^2) and squared prediction error (SPE) are used in the PCA. The SPE statistic is related to the residuals of process variables that are not illustrated by the PCA statistical model, and is a reliable indicator to a change in the correlation structure of the process variables. The SPE physically tests the fit of new data to the established PCA models and is efficient at identifying outliers from the PCA model [7]. The Hotelling T^2 score measures the Mahalanobis distance from the projected sample data point to the origin in the signal space defined by the PCA model.

The primary objectives of principal component analysis are data summarization, classification of variables, outlier detection, early warning of potential malfunctions and isolation of fault. PCA seeks to find a few linear combinations which can be used to summarize the data with a minimal loss of information. Let X = x_1, x_2, x_3,.., x_m be an m – dimensional data set describing the system variables. The first principal component is the linear combination of the columns of X, i.e. the variables, which describes the greatest variability in X, $t_1 = X_{p1}$ subject to $|p_1| = 1$. In the m-dimensional space p_1 defines the direction of greatest variability, and t_1 represents the projection of each sample data point onto p_1. The second principal component is the linear combination defined by $t_2 = X_{p2}$ which has the next greatest variance subject to $|p_2| = 1$ and subject to the condition that it is orthogonal to the first principal component, t_1 [11]. Essentially PCA decomposes the original signal X, as

$$X=TP^T = \sum_{i=1}^{m} t_i p_i^{\ T} \tag{5}$$

where p_i is chosen to be an eigenvector of the covariance matrix of X. P is defined as the principal component loading matrix and T is defined to be the matrix of principal component scores. The loadings provide information as to which variables contribute the most to individual principal components, and can help isolate the dominant faulty variables. In the approach used in this paper, each variable is separately scaled to zero mean and unit variance.

One important feature of the PCA model is that it gives information about the noise structure of the original data which means that it can tell something about variables that do not dominate on the variance level but are indeed degrading or faulty. Consequently, it is desirable to exclude less influential principal components from the signal space defined by the PCA model, which leads to the decomposition of X into the signal and residual subspaces. The signal subspace is intended to capture the variables that are contributing to any abnormal process variability and the residual subspace will complement this by examining the variables that are effectively over-shadowed by dominant variables in the signal subspace. It's important to note that faulty variables aren't always the ones that exhibit the greatest variability. An example of this phenomenon is presented in the data analysis and discussion section of this paper.

C. Principal Component Subspace Decomposition

Subspace decomposition into Principal Components can be accomplished using singular value decomposition of matrix X. The SVD of data matrix X, is expressed as $X=USV^T$, where $S=\text{diag}(s_1,\dots,s_m)\epsilon\ R^{n \times m}$, and $s_1>s_2>\dots>s_m$. The two orthogonal matrices U and V are called the left and right eigen-matrices of X. Based on the singular value decomposition, the subspace decomposition of X is expressed as:

$$X=X_s+X_r=U_sS_sV^T_s + U_rS_rV^T_r \tag{6}$$

The signal space S_s is defined by the PCA model and the residual subspace S_r is taken as the residual space [12]. The diagonal S_s are the singular values $\{s_1,\dots,s_k\}$, and $\{s_{k+1},\dots,s_m\}$ belong to the diagonals of S_r. The set of orthonormal vectors $U_s=[u_1,u_2,\dots,u_k]$ form the bases of signal space S_s. The projection matrix P_s onto the signal subspace is given by:

$$P_s=U_sU_s^{\ T} \tag{7}$$

The residual subspace is the orthogonal complement of the signal subspace and the projection of the original data onto it can be expressed as:

$$P_r=I-P_s \tag{8}$$

Any vector X can be represented by a summation of two projection vectors from subspaces S_s and S_r.

$$X=X_s+X_r=P_s\underline{x} + (I-P_s)\underline{x} \tag{9}$$

The subspace decomposition can also be accomplished by the eigen analysis of the correlation matrix of X, C, which is expressed as follows, where the columns of U are actually the eigenvectors of C, and the eigen values of C are the squared singular values of the diagonal matrix S. The eigen values provide a measure of the variance of each of the eigenvectors and determine the selection of the principal components and the number of principal components to choose.

$$C=(1/n)XX^T=(1/n)US^2U^T \tag{10}$$

D. Fault Detection Using Projection Pursuit Analysis

From the normal historical data one can derive the nominal normal system behavior statistics, mean, variance and from the above analysis the signal and residual subspaces. From the subspaces, we extract some statistics to describe the data distributions in two subspaces[13]. One is the Hotelling T^2 which measures the variation of each sample is the signal subspace. For a new sample vector x, it is expressed as:

$$T^2=\underline{x}^TU_sS^{-1}U_s^{\ T}\underline{x} \tag{11}$$

where S is the covariance of X, and is equal to U^TU. Another statistic, the squared prediction error (SPE), indicates how well each sample conforms to the PCA model, measured by the projection of the sample vector onto the residual space

$$SPE=||P_r\underline{x}||^2=r=||(I-P_s)\underline{x}||^2 \tag{12}$$

The process is considered normal if

$$SPE\leq\delta^2 \text{ and } T^2\leq\tau^2 \tag{13}$$

where δ^2 and τ^2 are the control limits for the SPE and T^2 statistics, respectively, given a 1-α confidence level. These limits assume that x follows a normal distribution and T^2 follows a χ^2 distribution with k degrees of freedom, where k is defined to be the cut off for the number of principal components used in the PCA model. Because SPE is a measure of the deviation in the residual space, it can be used to identify when the current operation deviates from the expected in terms of parameters that are not dominant but still abnormal. On the other hand, the T^2 will be more sensitive to the regular fluctuations that move the process away from normal based on the projections in the model subspace. The two statistics function independently in this analysis, although a combined index has been developed for process monitoring. Yue and Qin [14] proposed a convenient alternative for merging the information from SPE and T^2. For the purpose of this analysis though, each statistic will be examined separately.

E. Fault Isolation and Contribution Plots

After a fault has been detected, there are several methods that can be used to determine the critical system parameters. Contribution plots continue to be a widely used for fault diagnosis. During monitoring of the system each new observation that is projected on the model and residual subspaces will have a unique impact onto each subspace respectively as discussed earlier. The impact is quantified by

calculating the contributions to the SPE and T^2. The larger the contribution of an individual parameter, the more likely it is that the parameter is the reason for the changes or faults. The contribution of the mth parameter to the SPE is found by taking the squared residual associated with that parameter x_m by:

$$SPE_m = r_m^2 \qquad (14)$$

The contribution of all parameter to T^2 is given (in terms of the SVD) by:

$$T^2 = ||XUS^{-1/2}U^T|| \qquad (15)$$

III. EXPERIMENTAL SETUP

To demonstrate the feasibility of the proposed methodology, experiments were conducted to define a baseline for healthy products and to identify specific parameter behavior. Notebook computers were exposed to a set of environmental conditions representative of the extremes of their life cycle profiles. The performance parameters, the fan speed, CPU temperature, motherboard temperature, videocard temperature, %C2 state, %C3 state, %CPU usage, and %CPU throttle were monitored in-situ during the experiments. The baseline of healthy products was used to differentiate unhealthy products from healthy ones. The proposed anomaly detection methodology was verified by injecting an artificial fault into the system. Results from the study demonstrate the potential of the approach for system diagnostics and prognostics. Operational and environmental ranges and profiles that constitute a "healthy system" were used to replicate the real time usage of the notebook computer. Software was installed on the computer to be used. A set of user activities was defined and simulated using script file to run on notebook computers. An artificial fault was injected into the notebook computers to create and detect any change in system dynamics.

TABLE I
ENVIRONMENTAL CONDITIONS

Temperature- Humidity
5ºC with uncontrolled RH
25ºC with 55% RH (room)
25ºC with 93% RH
50ºC with 20% RH
50ºC with 55% RH
50ºC with 93% RH

TABLE II
EXPERIMENTS PERFORMED

Power Setting	Usage Level	Environmental Condition
AC adapter (when battery is fully charged)	1 - 4	1 - 6
AC adapter (when battery is initially fully discharged)	1 - 4	1 - 6
Battery only	1 - 4	1 - 6

Experiments were performed on ten identical notebook computers, representative of the state-of-the-art in (2007) notebook computer performance and battery life (nearly three and half an hours on a single battery). For the experiment, six different environmental conditions were considered as shown in Table I. For each temperature/humidity combination, four usage conditions and three power supply conditions were considered. Factorial experiment was designed to study the effect of each factor on the response variable, as well as the effects of interactions between factors on the response variable.

Table II shows the list of all 72 experiments. Each computer was turned on for 30 minutes before starting the experiment.

The software for in-situ monitoring was installed on the notebook computers, along with Windows XP Professional operating system, Microsoft Office, Front page, WinRunner, Spybot, Winamp, Real Player, Visual Studio, Java 5, Minitab, iTunes, Adobe Photoshop, MATLAB, Winzip and McAfee Antivirus. Selection of this software was based on the authors' discretion and experience. A script file was written using WinRunner software to simulate user activity. Antivirus application McAfee v8.0 was configured to run on the laptop all the time. A set of files (.doc, .mp3, .ppt, .pdf, .xls) was kept in a folder to be used during simulation. Notebook computers were kept at room temperature between each test condition. When the laptop was powered by the AC adapter (when the battery was fully charged), the test duration was 3.5 hours. When the laptop was powered by an AC adapter (when the battery was fully discharged), the test duration was determined by the time it took for the battery to fully charge. When the laptop was powered by its battery only, the test duration was determined by the time it took for the battery to fully discharge.

Same usage conditions were applied on all notebook computers to achieve time synchronization between computer and software application responses. The notebook computer's power mode was always set to ON. The screen saver and hibernation option were disabled to prevent these functions from occurring during an experiment. The wireless capability of notebook computer was disabled due to the limited wireless connectivity inside the temperature-humidity chambers. Four level of notebook computer usage were chosen:

1. Idle system - In this category the operating system is loaded, all windows are closed, user input from the keyboard or mouse, optical drive are disabled. USB or Firewire peripherals are not attached.

2. Office productivity - In this category, the usage condition is designed to simulate an office work environment. The simulator work is designed to read a word document as well as prepare a new word document. The simulator opens the file explorer and locates a file to be opened. The simulator opens a "technology benchmark report" word document of 88 pages and size of 2.6MB. The simulator reads through the document and uses arrow keys to move page up, page down and selects

a paragraph to copy. The simulator opens a new document from the word toolbar and pastes thee copied section to a new document. The simulator resizes both documents to make it easy to toggle between the two documents. The simulator switches to the original document and reads through pages and copies additional paragraphs and pastes again into the new document as new paragraphs. The simulator also types a new paragraph into the new document. With these activities, the simulator creates a five-page summary and saves it by pressing the save button in the word toolbar. Then it saves the file through invoking the save as file explorer and providing a file name for the new document. The simulator does a cleanup by resizing and closing all opened document. The simulator removes the new files from the desktop and pastes into another folder. Finally, the simulator closes all opened file explorer windows.

3. Media center – In this category, the usage condition is designed to simulate an entertainment need. Winamp (v5.24) media player started from the start menu. The file explorer window is opened by pressing the open button in Winamp. MP3 music files are stored on the hard drive and selected to play in Winamp. Music is stopped after 4 minutes followed by shutting down the Winamp player window. Real media player (v10.5) is started from the start menu. The file explorer window is opened to select video files by pressing the open button in Real player. Video files from a DVD are selected by maneuvering through the file explorer window and then played in Real player. Movie screens are resized to full screen. The movie is turned off after 90 minutes and Real player closed.

4. Game mode – In this category, the usage condition is designed to simulate gamming. Quake Arena II was started from the start menu and single player option is selected to start the game. After an hour of play, the game is stopped and exited.

The following section provides discussion on the data analysis and provides results on the data collected during these experiments. Data analysis is performed by the methodology discussed in the previous section.

IV. DATA ANALYSIS AND DISCUSSION

The experiments were performed on ten new notebook computers with an assumption that these products are representative of normal/healthy products. The MD value obtained from these datasets called Mahalanobis space is used to identify anomalies present in the product. Five thousand data points are selected from the experiments performed at CALCE. An effort was made to demonstrate the capability of the MD method to detect anomalies present in a test notebook computer and characterization of this computer model based on the baseline experiments. In Fig. 1, MD values are plotted to graphically present the performance of the test computer in comparison to the CALCE baseline. From Fig. 1, the test computer shows problems from the beginning, a fact verified by observing the actual data file for the test computer in which

the fan was not operating and the three monitored temperatures were unusually high. The drop in MD value for the test computer at the 2700th and 3600th observations are an indication that the computer were shutdown and then restarted, which caused a temporary drop in these temperatures. This can be verified by looking at the actual data file.

Based on our results, we have seen that a test computer can be characterized using experimental data representative of "healthy" computers. It is observed that the MD values for the test computer are different from the MD values corresponding to the baseline. Metrics corresponding to the analysis are given in Table III. The table gives the statistics of MD values for the test computer and the baseline. The test computer shows higher MD values as compared to the baseline.
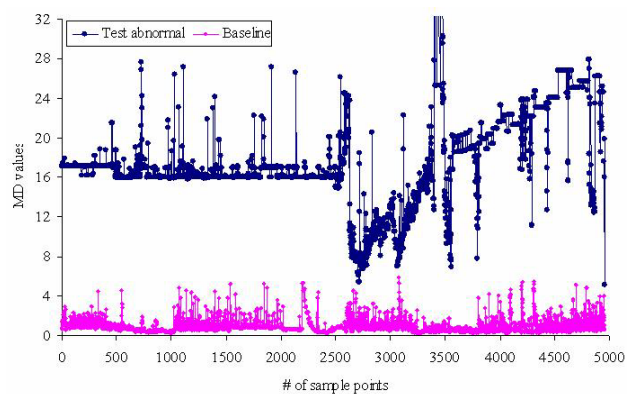

Fig. 1 Comparison of MD values of abnormal test computer with baseline

TABLE III
STATISTICS OF MD VALUE BASED ON EXPERIMENTAL DATA

System/Stats	Mean	Std Dev	1st Quartile	3rd Quartile
Baseline	0.91	0.65	0.53	1.03
Test Abnormal	17.65	5.14	16.01	19.87

Orthogonal array analysis is used to identify the significant parameters out of the eight original parameters. The leave-one-out experimental runs are shown in Table IV. The parameters are listed in the columns and the leave-one-out runs in the rows. An entry of '1' in the cell indicates that the parameter is included and '2' indicates that it's excluded. In total, nine leave-one-out runs were conducted, one with all parameters present and then the remaining eight excluding one parameter respectively each time to investigate the effect of that parameter on the MD output values. The S/N ratio is used as a measure of performance for each leave-one-out run and calculated using equation 4.
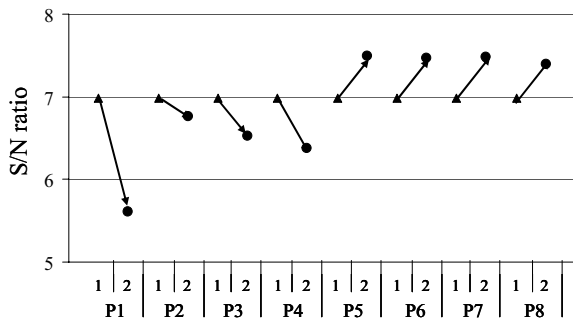
Fig. 2: Graphs of factorial effects

In Fig. 2, the difference in the (S/N) ratio is represented in Fig. 2 as a vector, and highlights parameters that have a vectors with a negative slope. These parameters are considered significant ones. The level of importance for each parameter is then further defined by the magnitude of their vector slopes. Parameters that show positive slopes are not considered significant because elimination of these parameters does not result into information loss. Since removal of these parameters result into higher S/N ratio these parameters are not identified as significant once, since larger the better S/N ratio is the criteria for parameter selection. This analysis identifies four important parameters: the fan is the most critical parameter out of these four, and the three temperature parameters are also shown to be important. They are affected by the failure of the fan and temperature increase of ~10 degrees centigrade as experienced by the temperature components. The following paragraphs discuss the Projection Pursuit analysis approach.

TABLE IV
ORTHOGONAL ARRAY

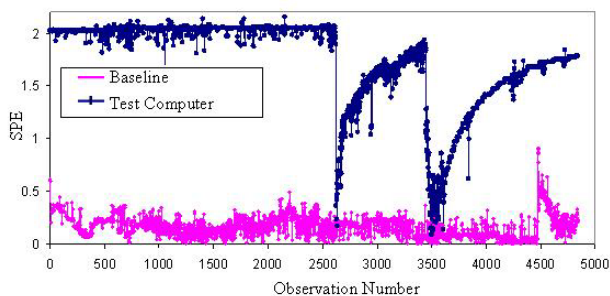| No.	Performance Parameters								S/N ratio
	Fan Speed	CPU Temp	Mother board Temp	Video card Temp	%C2 State	%C3 State	%CPU Usage	%CPU Throttle	
1	1	1	1	1	1	1	1	1	6.99
2	2	1	1	1	1	1	1	1	5.60
3	1	2	1	1	1	1	1	1	6.76
4	1	1	2	1	1	1	1	1	6.52
5	1	1	1	2	1	1	1	1	6.37
6	1	1	1	1	2	1	1	1	7.49
7	1	1	1	1	1	2	1	1	7.47
8	1	1	1	1	1	1	2	1	7.48
9	1	1	1	1	1	1	1	2	7.39

Fig. 3 Comparison of SPE scores of abnormal test computer with baseline

The goal of the Projection Pursuit analysis was to use Hotelling T^2 and SPE statistics from a healthy computer and successfully classify and detect faults in a new computer of the same model. Two "healthy" baseline sets of T^2 and SPE statistics were derived from two sources: one from a CALCE baseline set, based on 10 healthy computers of a different model, and the other based on one computer of the same model. From the analysis, the known faulty computer was identified as abnormal based on both comparisons. Fig. 3 and Fig. 4 are used as example plots to show the analysis results by plotting the SPE and Hotelling T^2 statistics for the above two scenarios versus the number of sample points. The lower pink line indicates baseline "healthy" values for each statistic and the upper blue line indicates the test values for each statistic for the abnormal test computer. It is clear from the plots that the abnormal test computer statistics are different from the baseline computer for both scenarios. The first five principal components were used to form the model space and the remaining three for the residual space.

From these results we see that the variability of the process in both the PCA model and residual subspaces can be used to capture abnormal system behavior. The detection is based on the geometry of the problem whose dimensions are established by the PCA model and residual subspaces. The subspaces as discussed are constructed from the "healthy" data and represent a fixed frame of reference used to compare incoming new observations. New observations are taken as a point in the multi-dimensional space and are projected onto the PCA model and residual subspaces respectively.

With the projection the new observation is reduced from its original dimension $R^{1 \times m}$ to the lower dimension of the PCA model, $R^{1 \times k}$, where k is the number of principal components used to form the model subspace. If the projection of the observation falls within the statistical control limits τ^2 and δ^2, of the model and residual subspaces respectively, then it is taken as normal or "healthy", otherwise it's treated as abnormal or "un-healthy". The PCA model can mask faults. This can occur for example when the new computer starts to exhibit abnormal behavior yet the variability of test data in both the model and residual subspaces fall within the "healthy" control limits for the system.
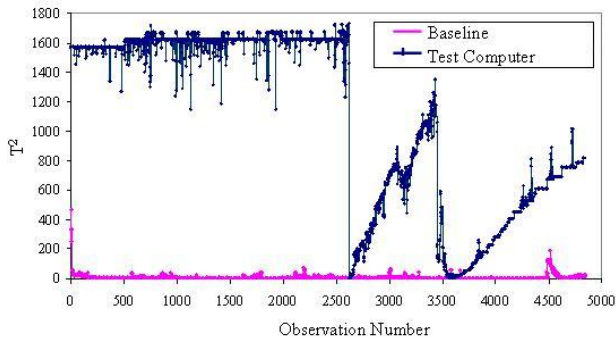
Fig. 4: Comparison of Hotelling T^2 scores of abnormal test computer with baseline

For the baseline, the statistics are modeled with patterns that are similar for both comparisons. One of the explanations for this is that the baseline for the computers captures the necessary range and variability of normal operating conditions of such computer models. Without the use of the control limits this analysis is left to identify the presence of abnormalities between test and training data and also to identify the critical system parameters.

A. Fault Isolation – Dominant Variables Using Projection Pursuit Analysis

The model space is designed to capture the data that varies the most, whereas the residual space is designed to capture the data that does not vary but contributes to a faulty state. The residual space can therefore detect changes in the distribution from variables that are degrading or have faults and are not effecting the variance. Below are the principal components for the entire subspace S. Each principal component is composed of the eight parameters with a particular weighting as shown in Table IV. The model/signal subspace is composed of the first four principal components. This was chosen based on iterative experimentation to best capture the faults. The decision of how many principal components are chosen to represent the model/signal space is based on experience and understanding of the data at hand. There are computational/statistical techniques that can provide estimates for the selection of the number of PCs to optimized results. The remaining columns span the residual subspace.

TABLE V

PRINCIPAL COMPONENT OF SUBSPACE S AND PARAMETER CONTRIBUTION

[S]	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8
Fan Speed	0.999	0.019	0.001	0.001	0.001	0.002	0.004	0.001
CPU Temp	0.005	0.076	0.042	0.048	0.000	0.484	0.856	0.153
Motherboard Temp	0.002	0.035	0.077	0.079	0.004	0.523	0.158	0.830
Videocard Temp	0.000	0.048	0.107	0.093	0.002	0.670	0.490	0.537
%C2 State	0.001	0.060	0.018	0.091	0.994	0.001	0.002	0.001
%C3 State	0.015	0.730	0.384	0.554	0.102	0.005	0.025	0.011
%CPU Usage	0.013	0.661	0.271	0.690	0.028	0.114	0.019	0.000
%CPU Throttle	0.001	0.130	0.872	0.439	0.033	0.166	0.038	0.005

Each variable is represented by each respective row of matrix [S]. The first row shows the contributions of the fan speed, and the rest show the CPU temperature, motherboard temperature, video card temperature, %C2 state, %C3 state, %CPU usage and %CPU throttle from top to bottom in matrix [S]. From the decomposition of [S] we can see that the model space variations should be dominated by the fan speed followed by %C3 state, %CPU throttle and usage. In the residual subspace the temperature components are dominant. We expect that the temperature variables to be highly dominant. Changes in the temperature are expected in turn to be less obvious to changes in system variance and should contribute to the shape of the multivariate data distribution. Such a distribution can be modeled as Gaussian mixtures, but in general a hard task. Intuitively, if the fan speed is not functioning, we expect that the temperature of the system will rise and become abnormally high. This is at first hand validated by the dominance of the temperatures components as observed in the residual subspace in [S]. Mathematically, this is also validated through the parameter contribution plots to the T^2 and SPE respectively as illustrated in the contribution plots shown in Fig. 5 and Fig. 6. The contribution plots tell us which parameter is contributing the most to the projection onto each subspace.

It is shown that on the model space the fan speed is highly dominant and varies the most in terms of standard deviation. This phenomenon masks the effect on parameters that are also exhibiting abnormalities but are overpowered by dominant parameters such as the fan speed. The residual space statistic, SPE, captures the inverse information and identifies the parameters that are indeed abnormal but are not dominating in terms of variance. Also interesting is the fact that the mathematics validates our intuition that because the fan is not functioning properly, the temperature sensors would be experiencing unusual readings. Note that these results are based on picking the model space using k = 4, that is the first four PCs in matrix [S]. The selection of more PCs for the model space and consequently fewer PCs for the residual space will change the results slightly. If all eight PCs are used to construct the model space then the SPE will be rendered ineffective although the results for the Hotelling T^2 will improve. Even though the results from the Hotelling T^2 improve with the selection of more PCs the information available through the SPE is lost. There are ways to select the optimum number of PCs necessary to optimize the information captured from both subspaces, often the selection

is purely based on experience or experimentation, although there are statistical methods such as the maximum likelihood estimator (MLE) which can estimate the optimum number of PCs to use.
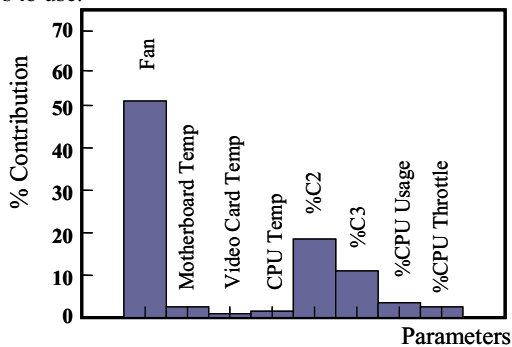

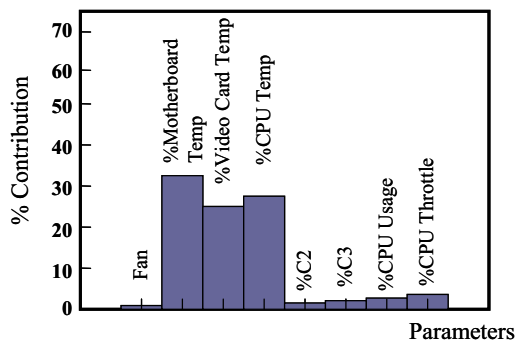Fig. 5 Contribution plot of each parameter towards T^2

Fig. 6 Contribution plot of each parameter towards SPE

V. Conclusions

A set of experiments in different usages and environmental conditions were conducted to establish the baseline "healthy or normal" operation on a set of notebook computers. A test computer was then subjected to the field use condition and it was evaluated using Mahalanobis Distance (MD), and Projection Pursuit analysis (PPA) techniques. In this study, PPA and MD were independently used to identify the similarity of new observations to healthy data, detect system anomalies and identify critical components. PPA performed this analysis in a reduced dimension based on an optimization criterion (maximum variance). The strength of PPA lies in the ability to decompose the signal and extract additional information not originally available, used to identify faults in the system. PPA overcomes masking effects when working with highly correlated data. The strength of the MD method is that it preserves all the information available because it does not reduce the original dimensionality of the data but it is susceptible to masking effects. Using an S/N ratio analysis based on Taguchi's technique the MD analysis was also used to identify the critical components.

Four critical parameters were identified through both methods: the fan speed and the three temperature components (CPU temperature, motherboard temperature, and videocard temperature). The fan speed is identified as the most

dominant, whereas the three temperature parameters were identified less dominant but still contributing to a faulty state. This finding validated the actual problem with the test computer, namely that its fan was malfunctioning.

References

[1] N. Vichare, P. Rodgers; V. Eveloy; and M. Pecht; "Environment and Usage Monitoring of Electronic Products for Health Assessment and Product Design," International Journal of Quality Technology and Quantitative Management. vol. 2, no. 4, 2007, pp. 235-250.

[2] J. Gu; N. Vichare; T. Tracy; and M. Pecht; "Prognostics Implementation Methods for Electronics," 53rd Annual Reliability and Maintainability Symposium (RAMS), Florida, 2007.

[3] G. Zhang, C. Kwan; R. Xu; N. Vichare; and M. Pecht; "An Enhanced Prognostic Model for Intermittent Failures in Digital Electronics," IEEE Aerospace Conference, Big Sky, MT, March 2007.

[4] N. Vichare; and M. Pecht; "Enabling Electronic Prognostics Using Thermal Data," Proceedings of the 12th International Workshop on Thermal Investigation of ICs and Products, Nice, Côte d'Azur, France, 27-29 September 2006.

[5] N. Vichare, P. Rodgers; and M. Pecht; "Methods for Binning and Density Estimation of Load Parameters for Prognostics and Health Management, International Journal of Performability Engineering, vol. 2, no. 2, April 2006.

[6] A. Fraser; N. Hengartner; K. Vixie; and B. Wohlberg; "Incorporating Invariants in Mahalanobis Distance based Classifiers: Application to Face Recognition," in International Joint Conference on Neural Networks (IJCNN), (Portland, OR, USA), Jul 2003.

[7] J. E. Jackson; and G. S. Mudholkar; "Control Procedures for Residuals Associated With Principal Component Analysis," Technometrics, vol. 21, no. 3, 1979.

[8] J. Liu, K. Lim; R. Srinivasan; and X. Doan; "On-Line Process Monitoring and Fault Isolation Using PCA," Proceedings of the 2005 IEEE International Symposium on, Mediterranean Conference on Control and Automation, 2005, pp. 658 – 66.

[9] G. Taguchi, and R. Jugulum; The Mahalanobis-Taguchi Strategy: A Pattern Technology System, Wiley, 2002.

[10] G. Taguchi, S. Chowdhury; and Y. Wu; The Mahalanobis–Taguchi System, New York: McGraw-Hill. 2001.

[11] E. B. Martin, A. J. Morris; and J. Zhang; "Process Performance Monitoring Using Multivariate Statistical Process Control," IEEE Proceeding of Control Theory Application, vol. 143, no.2, March 1996.

[12] H. Chen, G. Jiang; C. Ungureanu; and K. Yoshihira; "Failure Detection and Localization in Component Based Products by Online Tracking," KDD, 2005.

[13] H. Wang, Z. Song; and P. Li; "Fault Detection Behavior and Performance Analysis of Principal Component Analysis Based Process Monitoring Methods," American Chemical Society, vol. 41, 2002, pp. 2455 – 2464.

[14] H. Yue, and S. J. Qin; "Reconstruction-Based Fault Identification Using a Combined Index," American Chemical Society, vol. 40, 2001, pp. 4403-4414.

Sachin Kumar received the B.S. degree in Metallurgical Engineering from the Bihar Institute of Technology and the M.Tech. degree in Reliability Engineering from the Indian Institute of Technology, Kharagpur. He is currently pursuing the Ph.D. degree in Mechanical Engineering at the University of Maryland, College Park. His research interests include reliability, electronic system prognostics, and health and usage monitoring of systems.

Vasilis Sotiris received the B.S. degree in Aerospace Engineering from Rutgers University in New Brunswick, New Jersey and the M.S. degree in Mechanical Engineering from Columbia University in New York. He worked as a Systems Engineer for Lockheed Martin Corporation concentrating on software development projects for the Federal Aviation Administration. He is currently pursuing the Ph.D. degree in Applied Mathematics at the University of Maryland, College Park. His research interests are in the field of applied statistics and computational mathematics related to diagnostics and prognostics for electronic systems.

Michael Pecht has a B.S. in Acoustics, an M.S. in Electrical Engineering and an M.S. and Ph.D. in Engineering Mechanics from the University of Wisconsin at Madison. He is a Professional Engineer, an IEEE Fellow and an ASME Fellow. He has received the 3M Research Award for electronics packaging, the IEEE Award for chairing key Reliability Standards, and the IMAPS William D. Ashman Memorial Achievement Award for his contributions in electronics reliability analysis. He has written over twenty books on electronic products development, use and supply chain management. He served as chief editor of the IEEE Transactions on Reliability for eight years and on the advisory board of IEEE Spectrum. He has been the chief editor for Microelectronics Reliability for over eleven years and an associate editor for the IEEE Transactions on Components and Packaging Technology. He is a Chair Professor and the founder of the Center for Advanced Life Cycle Engineering (CALCE) and the Electronic Products and Systems Consortium at the University of Maryland. He has also been leading a research team in the area of prognostics, and formed the Prognostics and Health Management Consortium at the University of Maryland. He has consulted for over 50 major international electronics companies, providing expertise in strategic planning, design, test, prognostics, IP and risk assessment of electronic products and systems.