Automatic Tonnage Monitoring for Missing Part Detection in Multi-Operation Forging Processes

Yong Lei
State Key Laboratory of Fluid Power Transmission and Control,
Zhejiang University,
Hangzhou 310027, China
e-mail: ylei@zju.edu.cn

Zhisheng Zhang
College of Mechanical Engineering,
Southeast University,
Nanjing 210096, China
e-mail: oldbc@seu.edu.cn

Jionghua Jin[1]
Department of Industrial and Operations Engineering,
University of Michigan,
Ann Arbor, MI 48105
e-mail: jhjin@umich.edu

In multi-operation forging processes, the process fault due to missing parts from dies is a critical concern. The objective of this paper is to develop an effective method for detecting missing parts by using automatic classification of tonnage signals during continuous production. In this paper, a new feature selection and hierarchical classification method is developed to improve the classification performance for multiclass faults. In the development of the methodology, the signal segmentation is conducted at the first step based on an offline station-by-station test in a forging process. Afterwards, the principal component analysis is conducted on the segmented tonnage signals to generate the principal component (PC) features to be selected for designing the classifier. Finally, the optimal selection of PC features is integrated with the design of a hierarchical classifier by using the criterion of minimizing the probabilities of misclassification among classes. A case study using a real-world forging process is provided in the paper, which demonstrates the effectiveness of the developed methodology for detecting and diagnosing the missing parts faults in the multiple forging operation process. The classifier performance is also validated through the cross-validations to achieve a given average classification error. [DOI: 10.1115/1.4002531]

Keywords: hierarchical classification, multi-operation forging process, monitoring and diagnosis, tonnage signal

1 Introduction

Multi-operation forging processes with transfer or progressive dies are being increasingly used due to their high productivity and low manufacturing costs. In such processes, multiple dies/stations are designed to work together to produce a complete product in each stroke. One of the critical process faults is the missing part from dies during production, which does not only results in defective parts but also may damage expensive dies. Therefore, automatic monitoring and detection of missing parts in such processes is highly demanded in industrial practice.

With the rapid development of sensor and computer technology, online tonnage monitoring systems are widely used in most industrial practices. For example, Fig. 1 shows four strain gage sensors mounted on the four uprights of a forging press machine. These sensors are used to measure the aggregated tonnage forces exerted on all dies. The summation of these four tonnage sensor signals is referred to as the "total tonnage signal" in this paper. The mean profile of the total tonnage signals reflects the properties of the incoming raw materials, workpiece geometry, process setups, die working conditions, etc. Meanwhile, the variations in these tonnage signals reflect the natural process variations due to inherent process disturbance factors, including the variations in lubrication distribution, die temperatures, material uniformity, etc. Since tonnage force signals can provide a wealth of information about the process operation conditions, it is always desirable to effectively use these tonnage signals for online process monitoring and quality improvement.

In traditional industrial practices, several simple tonnage features are used for online process monitoring. For example, the maximum and the average of a cycle of signals are the most commonly used monitoring features [1–3]. Unfortunately, when using these simple features, a large amount of profile information provided in tonnage signals is not fully utilized. As a result, a monitoring system based on these simple features often suffers either a high false alarm rate when the process is operating normally or a high missing detection rate when the process is under faulty conditions. Recently, more advanced methods of analyzing tonnage signals have been investigated by applying multivariate statistical analysis and signal processing methods [4–6]. For example, in Ref. [4], T^2 control charts are constructed based on offline tests in order to obtain individual station signals. In Ref. [5], an advanced monitoring method is developed using the Haar transform, and in Ref. [6], a hidden Markov model based fault diagnosis system is developed based on an autoregressive model. Although significant achievements have been made in stamping process monitoring, the application of tonnage signals for forging processes monitoring is still very limited.

In order to develop effective tonnage monitoring systems for forging processes with multiple transfer/progressive dies, the need for additional research can be seen from the following three aspects. The first aspect is that a forging operation is a hot forming process that generates smooth tonnage profiles. As a result, in forging processes, it is harder to identify the actual working boundaries of each die in a complete cycle of total tonnage signals than it is in stamping processes. Therefore, signal segmentation is a necessary step for forging process monitoring and diagnosis.

The second aspect is that forging processes usually include operations that generate relatively small tonnage forces, which concealed in the overall tonnage signal profiles. In this paper, we call this type of operation a "weak operation." An example of these weak operations would be piercing and trimming operations. The corresponding tonnage force signal generated by a weak operation

[1]Corresponding author.

Contributed by the Manufacturing Engineering Division of ASME for publication in the Journal of Manufacturing Science and Engineering. Manuscript received November 5, 2008; final manuscript received August 11, 2010; published online October 4, 2010. Assoc. Editor: Zhongqin Lin.

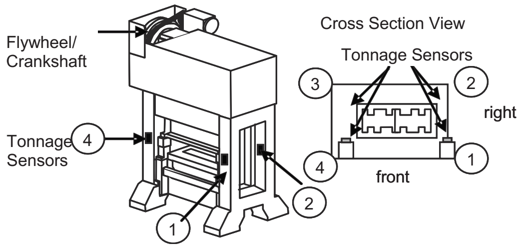
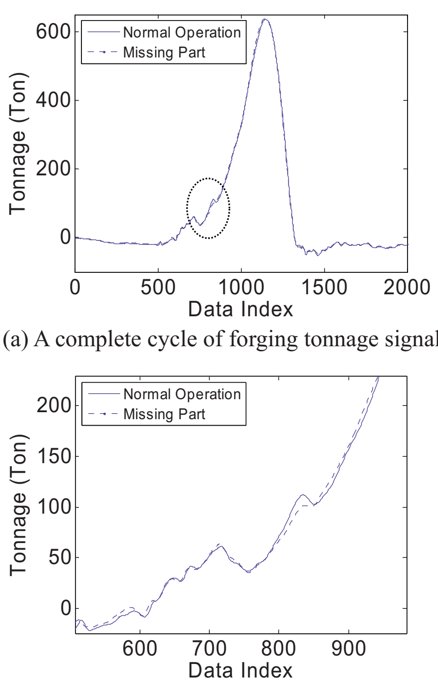


Fig. 1 Sensor distributions in the forging machine

is called a "weak (tonnage) signal." Generally, it is very difficult to detect a missing part at a weak operation die because the resulting change in the total tonnage signal profile is too subtle to be easily detected. For example, Fig. 2(*a*) illustrates two samples of total tonnage signals under two different working conditions. One is from normal operational condition when all dies are loaded with parts; the other signal is from a situation where a part is missing from the piercing operation die. It can be seen that the overall tonnage profiles of these two samples are quite overlapping. After we enlarge the circled local segment of the signals in Fig. 2(*a*), however, the subtle difference between these two samples can be seen in Fig. 2(*b*). But how to effectively detect such a subtle profile change is challenging, especially when the inherent process variability of multiple samples is considered.

The third aspect is that the inherent tonnage variability in forging operations is higher than the variability in stamping operations because the metal deforms much more freely under the high temperatures in forging operations than under low temperatures in stamping operations. Consequently, for the same operation, such as piecing or trimming, the signal-to-noise ratio of the corresponding tonnage signals in a forging process is usually lower than that in a stamping process. Therefore, existing tonnage monitoring methods that are used for stamping processes may not be directly applicable to the forging processes, especially for weak operations in forging processes.



(a) A complete cycle of forging tonnage signals

(b) Zoomed segmental tonnages

Fig. 2 Tonnage signal comparison due to missing weak operation forces

Recently, some advanced research has been conducted on forging process monitoring [7,8]. For example, in Ref. [7], the principal curve based tonnage monitoring method is developed based on segmented signals. In Ref. [8], principal component analysis is used for monitoring feature extraction based on a whole cycle of tonnage signals. These papers, however, did not investigate the tonnage monitoring for a forging process that consists of weak operation dies.

Unlike the approach of using statistical process control (SPC) charts for detection of abnormal conditions, this paper aims to detect and identify various fault conditions that correspond to parts missing from various operation dies. The proposed tonnage monitoring approach is formulated by solving a multiclass classification problem, i.e., training a multiclass classifier that can identify a part missing from a particular die or an unknown fault condition after detecting the change of tonnage signals.

For the purpose of designing classifiers, a large number of classification methods can be found in literature. These can generally be categorized into two groups in terms of the classifier structure. The first group consists of classifiers that use a single classifier to classify multiclass data. Examples of these advanced classification techniques include neural networks and support vector machines [9]. The second group consists of classifiers that use multiple classifiers to categorize multiclass data. Examples of these classification techniques include hierarchical classifiers, where different hierarchical structures are used in Refs. [9–12] and multiclass Adaboost, which uses multiple weak classifiers simultaneously to vote for a decision [13]. In this paper, we select the hierarchical classifiers because a hierarchical structure can effectively reflect different degrees of class separability at different hierarchical steps. This requirement is critical in this study because of the need to reveal the subtle differences between the classes of weak signals that are hidden in the total tonnage signals.

To design an effective classifier, feature extraction and feature selection play a critical role. Two essential questions need to be answered, i.e., how many features are needed and which features should be selected. In hierarchical classification methods, feature selection has received relatively little attention even though it is important in the classifier training process [14]. There are five commonly used feature selection criteria: the information measure, distance, dependence, consistency, and classification accuracy; their performances in hierarchical classifiers are discussed in Refs. [14–17]. The feature selection methods using the first four criteria are called filter approaches since they do not depend on classification algorithms. In contrast, the feature selection methods based on the classification accuracy criteria are called wrapper approaches, which use the trained classifier outputs to evaluate the selected feature subsets. However, both feature selection approaches have some drawbacks that limit their applications to our problem. In the filter approaches, the measures used for the feature selection usually do not directly represent the misclassification rates of the hierarchical classifiers, which cannot guarantee the performance of the classifier. In the wrapper approaches, the feature subsets are not evaluated during the classifier training in each hierarchical step; therefore, it does not help reduce the misclassification errors in the scenario of detecting weak-force operation signals because in our study, the weak-force signals are overshadowed by strong-force signals.

The objective of this paper is to develop a new tonnage monitoring method that can be effectively used for detecting missing parts in weak operations in multi-operation forging processes. Unlike existing feature selection approaches, we propose that, at each hierarchical classification step, the total misclassification error of each group should be used as the criterion for optimal feature selection. The advantage of this approach is that the performance of the selected feature subset can be directly evaluated in terms of total misclassification errors. Moreover, an inclined type of binary-tree structure [18] is selected for the classifier design to improve the signal-to-noise ratio of features. Using this hierarchi-

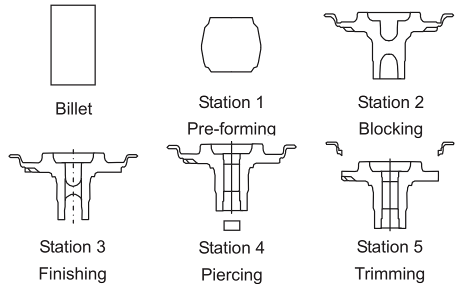


Fig. 3 Sketched billet and workpiece passing through five operations in a forging process

cal tree structure, the features can be better selected to reduce misclassification errors, especially for detecting missing parts at the weak-force operations.

This paper is organized as follows. A multi-operation forging process is introduced in Sec. 2. The proposed method for feature selection and hierarchical classifier design is discussed in detail in Sec. 3. A real-world example of a forging process is illustrated in Sec. 4 to demonstrate the implementation procedures and effectiveness of the proposed method. Finally, concluding remarks are given in Sec. 5.

2 Overview of an Exemplary Forging Process

A real-world forging process is used as an example in this paper to show the development of our methodology. In this process, one final product is produced by passing a raw billet through five individual dies that perform five operations in the following sequence: (1) preforming, (2) blocking, (3) finishing, (4) piercing, and (5) trimming. Figure 3 shows a shape sketch of raw billet, intermediate parts, as well as a final product after each operation. Of these five die operations, the blocking and finishing operations make significant shape changes on workpieces and thus generating large tonnage forces. In contrast, the piercing and trimming operations remove a small amount of material from a workpiece, thus generating small tonnage forces. This fact leads to the difficulty to detect a missing part at the piercing and trimming operations than at other operations.

For the purpose of classifier design, a training data set consisting of six groups of data is collected in which group $i(i=1,\ldots,5)$ corresponds to the fault condition due to a missing part at station i; group 0 corresponds to the normal working condition. For completeness, group 6 is used to represent all unknown potential process faults. Figure 4 shows all training samples of total

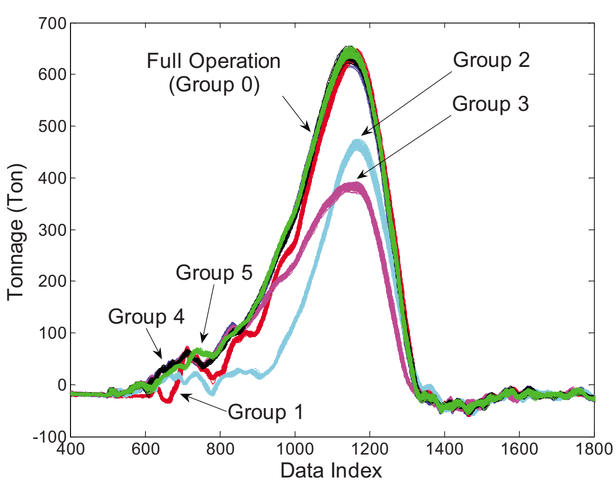


Fig. 4 Sample tonnage signals at six different conditions

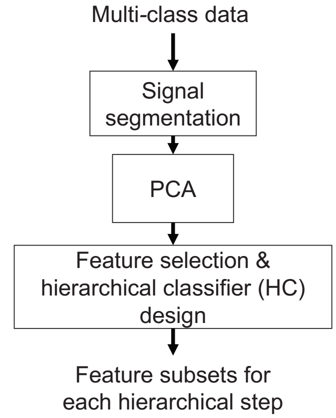


Fig. 5 Training procedures for classifier design

tonnage signals, including 307 samples from the normal operating condition and 69 samples from each of five missing part conditions. It can be seen that it is relatively easy to separate groups 2 and 3 (missing parts at the blocking and finishing dies) from other groups but more difficult to separate groups 4 and 5 (missing parts at the piercing and trimming dies) from group 0 (normal operating condition).

3 Methodology Development

In this study, a new hierarchical classification method integrated with optimal feature selection is developed to enhance the performance of detecting weak signals. The training procedure for the proposed classifier is illustrated in Fig. 5. First, the signal segmentation is conducted as a data preprocessing procedure to improve the separability of different groups at their corresponding local segments. Second, principal component analysis (PCA) [19] is used for data dimension reduction by using a few significant principal components (PCs) as features to represent the original total tonnage signals at individual segments. Then, feature selection and the corresponding hierarchical classifier design are accomplished through the supervised training based on the six available groups of training samples. The outputs from the third step are the optimally selected feature subsets and the corresponding decision rules of the resultant hierarchical classifier. These are used to ensure an acceptable misclassification error at each hierarchical classification step. Finally, the overall performance of the developed classifier is evaluated by constructing a confusion matrix. The details involved at each step will be discussed in Secs. 3.1–3.4, respectively.

3.1 Signal Segmentation. In multi-operation processes, when a missing part occurs at an individual die, the total tonnage signal will only partially change at the specific signal segment corresponding to the working range of that particular die. Therefore, for the purpose of missing part detection at a particular die, it is helpful to extract features at the corresponding local data segment that is relevant to the working range of the faulty die rather than using the whole cycle of total tonnage signals. In this way, the class separability of using such extracted features is increased because the local profile changes are enhanced after removing irrelevant data segments. For example, in Fig. 4, groups 2 and 3 are more separable from other groups near the peak tonnage areas at the data segment of [800,1200] and [900, 1200], respectively, but groups 4 and 5 have a higher separability from other groups at the data segment of [600, 900]. Therefore, different data segments show different sensitivities in separating groups of signals.

To conduct data segmentation of total tonnage signals, the working ranges of each individual die need to be known. Ideally, for multi-operation processes, the working ranges of each die can be determined by the timing-charts from the die design, which,

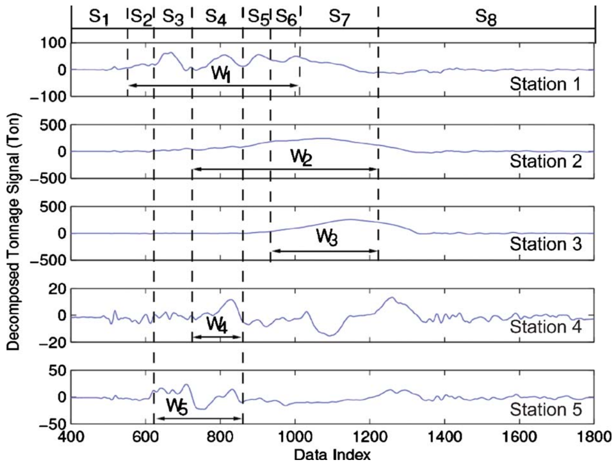


Fig. 6 The decomposed individual signals at each working station

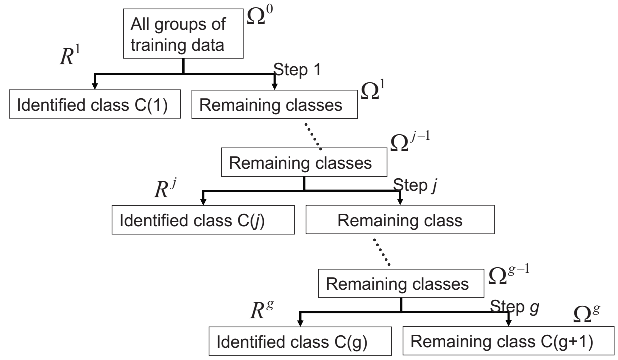


Fig. 7 Hierarchical structure of inclined binary-tree classifier

however, may not always be available in industrial practice. In this case, Jin and Shi [20] proposed a method by conducting a set of offline station-by-station tests to obtain the decomposed tonnage signals for individual dies. Figure 6 shows the decomposed results of the tonnage signal generated at each die station in which the working range of each die is marked as W_i $(i=1,\ldots,5)$, respectively.

After obtaining the working ranges of each die, the data segmentation can be further conducted in which the boundaries of each data segment are marked by either the start point or end point of the working ranges of individual dies. As shown in Fig. 6, a total of eight segments are obtained and only segments S_2-S_7 will be used for feature extraction. By using this type of data segmentation method, an explicit mapping relationship can be obtained for each given data segment, i.e., determining the relevant working dies that are generating the tonnage force at that specific segment. The details of data segmentation method can be found in Ref. [20]. Therefore, the change of the extracted features at the specific data segmentations can be uniquely mapped to the faulty stations from which the parts are missing.

3.2 Feature Extraction Using Principal Component Analysis. In this section, PCA is used to reduce the data dimension and generate features from the selected signal segments. PCA linearly transforms the raw data set into a new set of features called PCs.

Let $\mathbf{X}_k^{S_i} \in \Re^{n_k \times p_{S_i}}$ denotes n_k samples from group k; each sample contains p_{S_i} data points of total tonnage signals at segment S_i. It is assumed that $\mathbf{X}_k^{S_i}$ is normally distributed. Let $\bar{\mathbf{X}}_0^{S_i}$ and $\mathbf{S}_0^{s_i}$ denote the sample mean and sample covariance of $\mathbf{X}_0^{S_i}$ under the normal operation condition. The PCA transform is conducted on $\mathbf{X}_0^{S_i}$ and the resultant eigenvalue-eigenvector pairs are denoted as $(\lambda_1,\mathbf{e}_1),(\lambda_2,\mathbf{e}_2),\ldots,(\lambda_{p_{s_i}},\mathbf{e}_{p_{s_i}})$ with $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{p_{s_i}} \geq 0$ and $\mathbf{e}_j^T=[\mathrm{e}_{j1},\mathrm{e}_{j2},\ldots,\mathrm{e}_{jp_{s_i}}]$. The jth principal component of the vth sample under the normal operation condition is obtained by

$$\mathbf{y}_j = \mathbf{e}_j^T[\mathbf{X}_0^{S_i}(\mathrm{v},\bullet) - \bar{\mathbf{X}}_0^{S_i}] \tag{1}$$

where $v=1,\ldots,n_0$. The sample variance of the jth principal component $\mathbf{y}_j$ is λ_j, $j=1,2,\ldots,p_{S_i}$, and the sample covariance between $\mathbf{y}_j$ and $\mathbf{y}_l$ is zero if $j \neq l$. In addition, the total sample variance is $\mathrm{trace}(\mathbf{S}_0^{s_i})$, which is equal to $\lambda_1+\lambda_2+\cdots+\lambda_{p_{S_i}}$. The percentage of the sample variance explained by the jth principal component is given by $\lambda_j/\mathrm{trace}(\mathbf{S}_0^{s_i})$. In this paper, the principal components with the larger eigenvalues, which contribute 99.7% of the total variance, are considered as the candidate features for further feature selection.

For training our classifier, the jth principal component of v'-th sample $(v'=1,\ldots,n_k)$ of group k in the same segment (S_i) is also obtained by the following projection:

$$\mathbf{y}_j' = \mathbf{e}_j^T[\mathbf{X}_k^{S_i}(\mathrm{v}',\bullet) - \bar{\mathbf{X}}_0^{S_i}] \tag{2}$$

It should be noticed that the projections are made based on the eigenvectors $\mathbf{e}_j^T$ and sample mean $\bar{\mathbf{X}}_0^{S_i}$, which are obtained from the samples under the normal operating condition.

3.3 Feature Selection and Hierarchical Classifier Design. Considering the different degrees of class separability at different data segments, especially the low separability of the weak signals, it is very difficult to classify all the missing part conditions through one classifier using the same features. Therefore, a hierarchical classifier is used for classifying multiple missing part conditions through a special inclined type of binary-tree structure [18], i.e., only one missing part condition will be separated from others at each step of classification.

The hierarchical structure of such a classifier and the resultant sample spaces after each classification step are shown in Fig. 7. In this figure, Ω^j and R^j denote the remaining unclassified samples and the identified group samples at step j, respectively. At the beginning, let Ω^0 denotes all available training samples of all g known groups. Following the tree structure, at each step j, $(j=1,\ldots,g)$, one group will be identified, and $C(j)$ is used to denote the identified group index. For example, if group l is identified at step j, we have $C(j)=l$. The final remaining group $C(g+1)$ is used to denote all unlearnt faulty operation conditions.

In the design of the proposed hierarchical classifier, the key issues include how to determine which group is to be classified at step j, how many features are essential for this classifier design, and which optimal feature subsets are selected for this classifier design. These issues will be addressed in details as follows.

In order to define the criterion for determining which group is classified at step j, let π_l denote group l and α_{π_l} denote the total probability of misclassifying the samples in π_l to all other classes throughout all classification steps. The smallest misclassification error of α_{π_l} $(\forall l \neq C(1),\ldots,C(j-1))$ is used as the criterion for determining which group is identified at step j. This criterion ensures that π_l is the most correctly classified group at step j. The calculation of α_{π_l} is given in Eq. (5), which will be discussed later in this section.

The flowchart for integrating feature subset selection and classifier design is shown in Fig. 8. Let $\mathbf{y}$ denote a sample represented by the selected PC features, which can be either univariate or multivariate. At step j, $\mathbf{y}$ is to be classified as π_l based on the criterion of the minimum expected misclassification errors [21], i.e., the following Eq. (3) should be held.

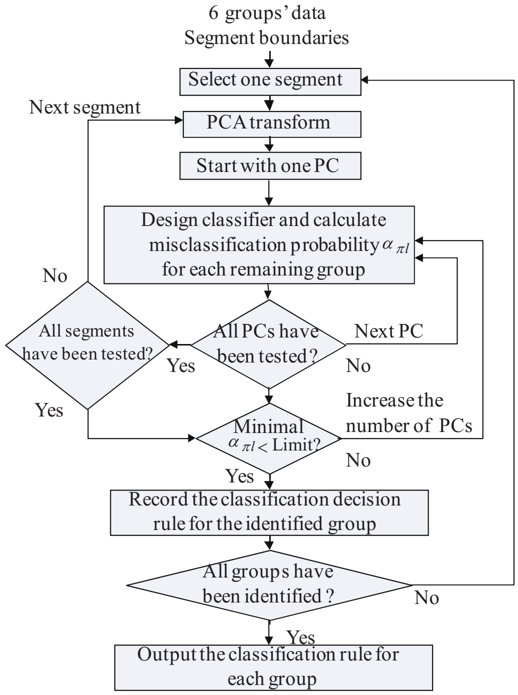


Fig. 8 Flowchart for optimal feature subset selection and classifier design

$$P(\pi_l)f_l(\mathbf{y}) > P(\pi_k)f_k(\mathbf{y}), \quad \text{for all possible } k, k \neq l \tag{3}$$

where $f_k(\bullet)$ denotes the probability density function of group k associated with the selected feature subset. $P(\pi_k)$ denotes the prior probability of group k, which can be specified based on either engineering knowledge or estimated from the training data set. If all the fault classes are completely preknown, the condition of $\Sigma_{k=1}^{g} P(\pi_k)=1$ is satisfied. Otherwise, $1-\Sigma_{k=1}^{g} P(\pi_k)$ represents the total probability of all unknown fault classes. Since PCs used as features follow a multivariate normal distribution, Eq. (3) is equivalently represented as

$$\ln P(\pi_l)f_l(\mathbf{y}) = \max_k \ln P(\pi_k)f_k(\mathbf{y}) = \ln P(\pi_l) - \frac{p}{2}\ln(2\pi) - \frac{1}{2}\ln|\mathbf{S}_l| - \frac{1}{2}(\mathbf{y}-\bar{\mathbf{x}}_l)^T \mathbf{S}_l^{-1}(\mathbf{y}-\bar{\mathbf{x}}_l)$$

where $\bar{\mathbf{x}}_l$ and $\mathbf{S}_l$ denote the sample mean and sample covariance matrix of the selected PCs of group l.

In order to determine the minimum number of the features for classifying group l from other remaining unclassified groups at step j, the classifier is designed with the first effort of using a single optimal feature that is selected from all the candidate PCs with the minimal classification error α_{π_l} for group l. As shown in Fig. 8, if its classification performance based on a single feature is not satisfied, the dimension of the feature subset will be increased until the stopping criteria is satisfied. In the worst case, it is possible that the stopping criterion is not satisfied after all features are exhaustedly searched. In this case, all the features will be used. It should be notified that the increase of the number of features leads to the increase of the discrimination distances between normally distributed classes [21], which can improve the classification accuracy. However, an inappropriate high dimension of features will lead to an over fitting problem at the training stage. Therefore, in the paper, we study how to determine the minimum number of features under a given satisfying classification performance, which can ensure a good robustness performance for future unknown test data. At each searching step under the given dimension of the feature subset, the optimal features are exhaustedly searched from all candidate PCs at all data segments. In this way, the feature interactions can be well considered in the design of classifiers. Moreover, the selected number of candidate PC features is generally not more than ten. Therefore, the exhausted searching will be feasible at the training stage.

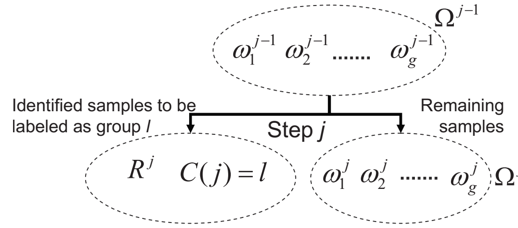


Fig. 9 Sample spaces at hierarchical step j

As shown in Fig. 9, at the $(j-1)$th step, the remaining samples of group i in Ω^{j-1} are denoted as ω_i^{j-1}, $i=1,2,\ldots,g$. Let P_i^j denotes the conditional probability of classifying ω_i^{j-1} to the identified sample space R^j at step j,

$$P_i^j \equiv \Pr\{x \in R^j | \omega_i^{j-1}\} \tag{4}$$

where $i=1,2,\ldots,g$, $j=1,2,\ldots,g$. The detailed calculation of P_i^j can be found in Appendix A.

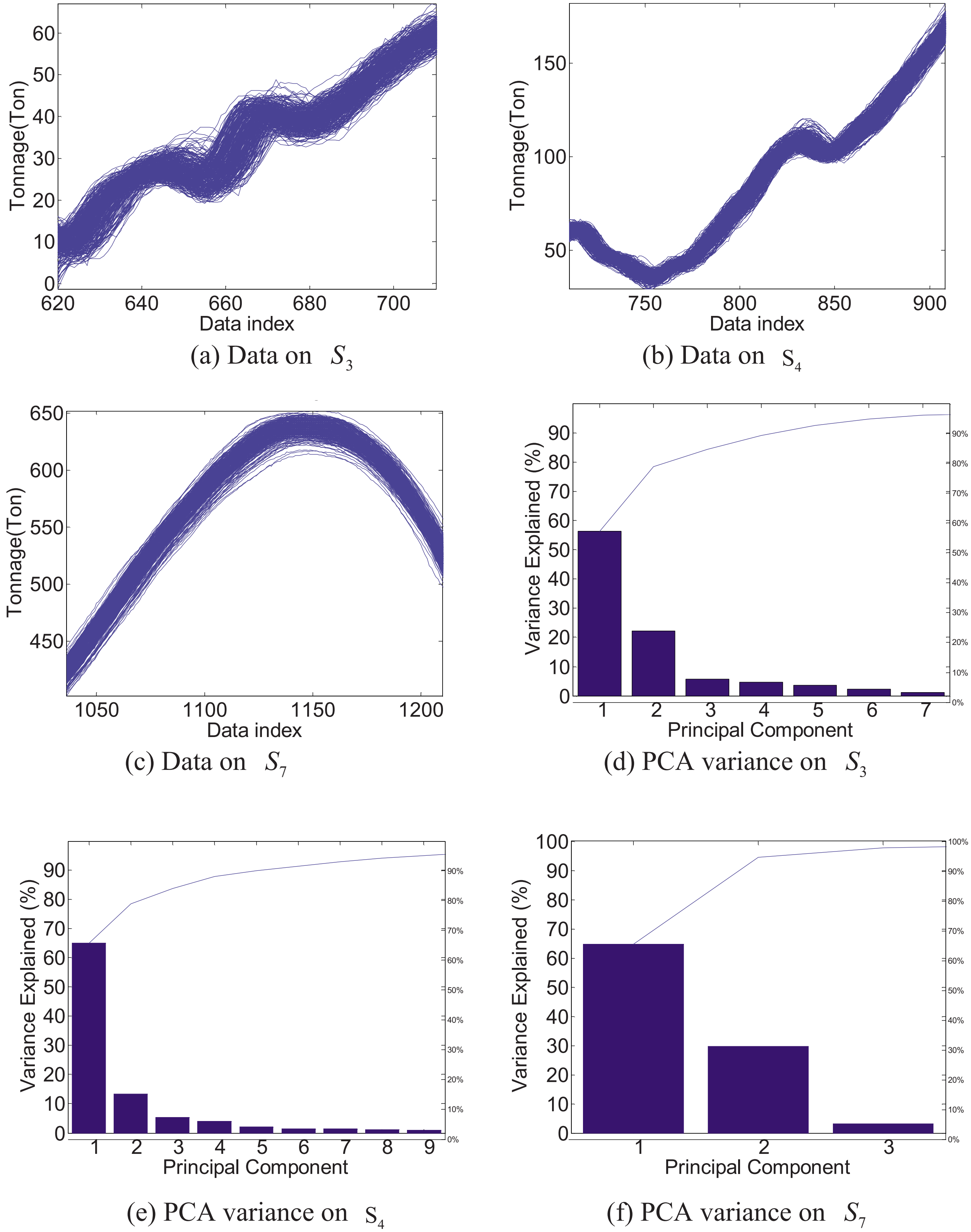
In this paper, the total misclassification error α_{π_l} $(C(j)=l)$ is defined as the probability of misclassifying group l into all other classes. This probability can be calculated as Eq. (5) based on its complementary probability, i.e., one minus the probability of correct classification of group l $(C(j)=l)$ at step j.

$$\alpha_{\pi_l} = \Pr\{x \notin R^j, C(j)=l | \pi_l\} = 1 - P_l^j \prod_{s=1}^{j-1}(1-P_l^s), \quad j = 1,2,\ldots,g \tag{5}$$

The detailed derivation of Eq. (5) can be found in Appendix B. In Eq. (5), it can be seen that although α_{π_l} has a propagated chain effect, it is only affected by the classifier design from step 1 to step j. Therefore, it is possible to calculate α_{π_l} at step j and use it as a criterion for determining the optimal class at step j. As mentioned previously, in order to select an optimal feature subset for classification, we iteratively select PCs and segments that can give the minimal α_{π_l} at step j. Moreover, the selection of the number of features (PCs) is determined by the minimal number of features that can satisfy the condition of $\alpha_{\pi_l} <$ limit in which the limit is prespecified based on the acceptable misclassification error in the given application.

Finally, at the last step (step g), where only one group is left to be identified in Ω^{g-1}, a T^2 control chart is used to identify the last group from the unlearnt conditions using the feature subset selected at the $(g-1)$th step. For a user prespecified type I error α, sample $\mathbf{x}$ in ω_i^{g-1} is classified to group l $(l=C(g))$ if $(\mathbf{x}-\bar{\mathbf{x}}_l)\mathbf{S}_l^{-1}(\mathbf{x}-\bar{\mathbf{x}}_l)^{\mathrm{T}} < \chi_\alpha^2$ is satisfied.

3.4 Classifier Performance Assessment. The classification performance for the designed classifiers can be finally evaluated by constructing a confusion matrix $\mathbf{M} \in \Re^{g\times(g+1)}$ after all classification steps. Let M_{kl} denote the probability of samples in group k (π_k) to be classified as group l (π_l), which is identified at step j, i.e., $C(j)=l$. It can be calculated by



(a) Data on S_3 (b) Data on S_4 (c) Data on S_7 (d) PCA variance on S_3 (e) PCA variance on S_4 (f) PCA variance on S_7

Fig. 10 Three selected segments and PCA analysis results

$$M_{kl} = \begin{cases} \Pr\{x \in (R^j \cap \pi_k), C(j) = l\} = P_k^j \prod_{s=1}^{j-1} (1 - P_k^s), & j = 1, 2, \ldots, g \\ \Pr\{x \in (\Omega^g \cap \pi_k), C(j) = l\} = \prod_{s=1}^{g} (1 - P_k^s), & j = g + 1 \end{cases} \tag{6}$$

Table 1 PC selection result

Step	Segment	PC	Classified group
1	S_4	Second	2
2	S_7	First	3
3	S_3	Second	1
4	S_3	Third	5
5	S_3	Third	4

(See Appendix C for the detailed proof.)

An alternative to evaluating the classification performance is to use the similar concepts of type I error and type II error in statistical process control. Since at each step j, only one class of group l ($C(j)=l$) is separated from the remaining groups, type I error can be defined as the total probability that the data in class l (π_l) would be falsely classified into all other classes; at the same time, type II error for class l can be defined as the total probability that all other classes would be misclassified into class l. Based on these definitions, type I error of the group l is exactly equal to α_{π_l}. Type II error, denoted as β_{π_l}, can be calculated by Eq. (7).

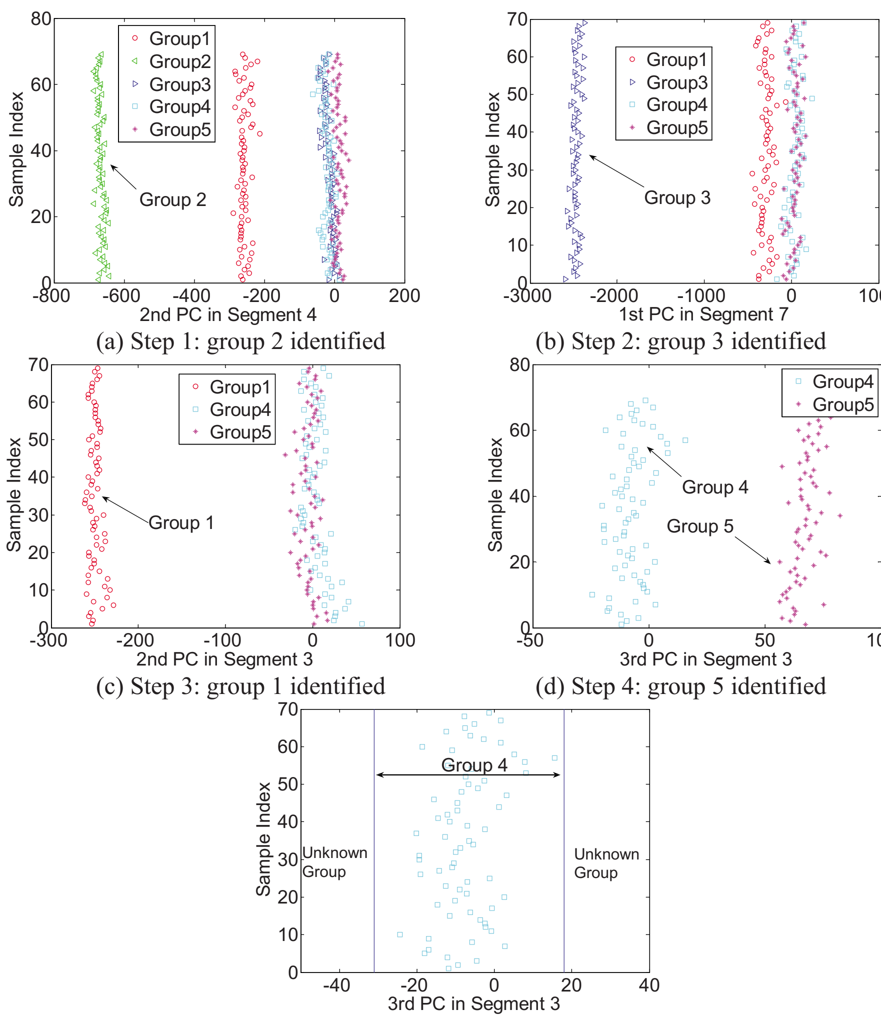
$$\beta_{\pi_l} = P\{x \in R^j, C(j) = l | \pi_k, k \neq C(j)\} = \sum_{k=1,k\neq l}^{g} P(\pi_k) \times M_{kl} \tag{7}$$

4 Case Study

In this section, we will illustrate the detail implementation procedures step by step to show how to apply the proposed methodology to a real-world forging process. The analysis results including PCA, PC feature selection, and the classifier performance assessment are given.

4.1 PCA Results Based on Segmental Tonnage Signals. As discussed in Sec. 3.2, the PCA transform is performed on each selected data segment of the training signals under the normal working condition. The resultant eigenvectors $\mathbf{e}_j(1 \leq j \leq p_{s_i})$ will be used as the orthogonal projections to obtain PCs as features for classifier design. Figures 10(*a*)–10(*c*) show the tonnage profiles of 307 training samples at the final selected segments (S_3, S_4, and S_7) under the normal working condition and Figs. 10(*d*)–10(*f*) show the Pareto plots of the selected candidate PCs at the corresponding segments that include 99.75% of total variance.

4.2 Optimal Feature Subsets. Table 1 shows the selected



(a) Step 1: group 2 identified

(b) Step 2: group 3 identified

(c) Step 3: group 1 identified

(d) Step 4: group 5 identified

(e) Step 5: group 4 identified

Fig. 11 Classification results step by steps (a)–(e)

Table 2 Final confusion matrix (%)

Classified group (k)	1	2	3	4	5	6
1	100	0	0	0	0	0
2	0	100	0	0	0	0
3	0	0	100	0	0	0
4	0	0	0	100	0	0
5	0	0	0	0	99.39 (1.35)	0.61 (1.35)

The value less than 10^{-3} is treated as 0.

Table 3 Classification probabilities for each group (%)

Classified group (k)	1	2	3	4	5
α_{π_k}	0	0	0	0	0.61 (1.34)
β_{π_k}	0	0	0	0	0

The value less than 10^{-3} is treated as 0.

feature subsets at each hierarchical classification step. It can be seen that segment S_4 is selected at the first step for classifying group 2, and segment S_7 is selected at the second step for classifying group 3 by using PC2 and PC1, respectively. These classification results are consistent with our previous discussion, i.e., the fact that groups 2 and 3 are more easily separated from other groups at segments 4 and 7, respectively. Also, groups 2 and 3 are identified at the first two steps because they are more easily separated than other groups. For classifying groups 1, 4, and 5, segment 3 is selected, which are used to enhance their separability for group 4 and 5 (weak signals). Figures 11(a)–11(e) visually show the selected features at each step by using resultant optimal feature subset listed in Table 1.

4.3 Classifier Performance Assessment. The performance of the proposed hierarchical classifier is assessed using cross-validations in which the validation classification errors provide the estimate of the classifier's performance on the test data sets. Generally, if a classifier has a large number of parameters to train, a large sample size of the training data set should be used and the portion r of the validation data set is small ($r=10-25\%$ of the total sample data). It has been shown that the sample size of the training set affects the bias of the classification errors while the sample size of the validation test set affects the variance of classification errors [9]. In this case study, $r=25\%$ of the data in each group are randomly selected as the test data set, and the remaining 75% of the data are used as the training data. The choice of $r=25\%$ is to consider a good tradeoff between a satisfying average classification errors and a small variance of classification errors using 100 repeated cross-validations. Table 2 shows the resultant confusion matrix, and Table 3 shows type I and type II errors defined in Sec. 3.4, respectively. In both Tables 2 and 3, the numbers that are given inside and outside of the parentheses in each cell are the corresponding standard deviations and average values throughout 100 repeated cross-validation tests. It can be seen that the resultant averages and variances of all classification errors are small, which demonstrates the effectiveness of our proposed method and a good choice of the test sample size in the cross-validations. It should be notified that the selected features at each step are not changed under different randomly selected training and test samples at all cross-validation tests, which shows the robustness of the proposed feature selection procedure.

5 Conclusion

In this study, a new feature selection and hierarchical classification method is developed for missing part detection in multi-operation forging processes. In the development of the method, first, both the data segmentation and the PCA transform are applied to reduce the dimension of the production data. Second, PCA is used to extract features from the selected data segments under various missing part conditions. Third, a hierarchical feature selection procedure is developed to minimize the misclassification probabilities among different groups. Finally, a corresponding hierarchical classification rule is developed for classifying various missing parts conditions. In addition, the performance of the corresponding classifier in terms of misclassification errors is evaluated using a confusion matrix. A real-world forging process is used to illustrate the implementation procedures and demonstrate the effectiveness of the proposed methodology.

Acknowledgment

This research is partially supported by the NSF Career Award No. DMI-0549306, the NSF under Grant No. DMI-0541750, and the 21st Century Jobs Fund under Grant No. F015886.

Appendix A: The Calculation of P_i^j

$$P_i^j = \Pr\{\mathbf{x} \in R^j | \omega_i^{j-1}\} = \int_{\Omega'} f_i(\mathbf{x}) \times P^j(\pi_i)\mathrm{d}\mathbf{x}$$

$$\text{for } j = 1, \ldots, g-1$$

where $\Omega' := \{x \in \Re^D | f_l(x)P^j(\pi_l) > f_k(x)P^j(\pi_k), \forall k \in \{C(j+1), \ldots, C(g)\}\}$ denotes the decision region falling, which the data are classified as group l at step j, thus $l=C(j)$. $f_i(\bullet)$ denotes the probability density function of group i. In this study, we assume $f_i(\mathbf{x}) \sim N(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$, where $\boldsymbol{\mu}_i \in \Re^D$ and $\boldsymbol{\Sigma}_i \in \Re^{D\times D}$, and D denotes the dimension of the feature subset. Because PCs follow a multivariate normal distribution, in which the variance of the selected PC is equal to its corresponding eigenvalue and the covariance between each pair of PC features is equal to zero due to the orthogonal transform of PCA. Therefore, the parameters of the density functions of PC features in each group can be estimated by the sample mean and sample variance of the PC features in each group of training samples. $P^j(\pi_i)$ denotes the prior probability of group i at step j, which can be calculated by $P^j(\pi_i) = P(\pi_i)\Pi_{s=1}^{j-1}(1-P_i^s)$. This probability is explained as the probability of samples in group i being kept in the remaining set of the unclassified samples after step $j-1$. Since P_i^s denotes the probability of the samples in group i that are classified at each of the previous step s ($s=1, \ldots, j-1$). Therefore, $\Pi_{s=1}^{j-1}(1-P_i^s)$ represents the total probability of the samples in group i that are not classified before step j.

At the last step ($j=g$), since only one known group is unidentified in Ω^{g-1}. A T^2 control chart is constructed to identify the known group. Hence, given prespecified type I error α, we have

$$P_i^g = \Pr\{x \in R^g | \omega_i^g\} = \Pr\{(\mathbf{x} - \bar{\mathbf{x}}_l)\mathbf{S}_l^{-1}(\mathbf{x} - \bar{\mathbf{x}}_l)^{\mathrm{T}} < \chi_\alpha^2 | \mathbf{x} \in \omega_i^g\}$$

where $l=C(g)$, $\bar{\mathbf{x}}_l$ denotes the sample mean of group l, and $\mathbf{S}_l^{-1}$ denotes the sample covariance of group l.

Appendix B: The Derivation of Equation (5)

$$
\begin{aligned}
\alpha_{\pi_l} &= \Pr\{\text{Sample in group } l \text{ to be classified as other groups}\} \\
&= \Pr\{x \notin R^j, C(j) = l|\pi_l\} \\
&= 1 - \Pr\{x \in (R^j \cap \pi_l), C(j) = l|\pi_l\} \\
&= 1 - \{\Pr[x \in (R^j \cap \pi_l), C(j) = l|x \in \Omega^{j-1}] \cdot \Pr[x \in (\Omega^{j-1} \cap \pi_l\ ] \\
&\quad + \underbrace{\Pr[x \in (R^j \cap \pi_l), C(j) = l|x \in R^{j-1}]}_{0} \cdot \Pr[x \in (R^{j-1} \cap \pi_l)]\} \\
&= 1 - \Pr[x \in (R^j \cap \pi_l), C(j) = l|x \in \Omega^{j-1}] \cdot \Pr[x \in (\Omega^{j-1} \cap \pi_l)] \\
&= 1 - P_l^j \cdot \Pr[x \in (\Omega^{j-1} \cap \pi_l)]
\end{aligned}
$$

where

$$
\begin{aligned}
\Pr[x \in (\Omega^{j-1} \cap \pi_l)] &= \Pr[x \in (\Omega^{j-1} \cap \pi_l)|x \in (\Omega^{j-2} \cap \pi_l)] \cdot \Pr[x \in (\Omega^{j-2} \cap \pi_l)] \\
&\quad + \underbrace{\Pr[x \in (\Omega^{j-1} \cap \pi_l)|x \in (R^{j-2} \cap \pi_l)]}_{0} \cdot \Pr[x \in (R^{j-2} \cap \pi_l)] \\
&= \Pr[x \in (\Omega^{j-1} \cap \pi_l)|x \in (\Omega^{j-2} \cap \pi_l)] \cdot \Pr[x \in (\Omega^{j-2} \cap \pi_l)] \\
&= \{1 - \Pr[x \in (\Omega^{j-1} \cap \pi_l)|x \in (\Omega^{j-2} \cap \pi_l)]\} \cdot \Pr[x \in (\Omega^{j-2} \cap \pi_l)] \\
&= \{1 - \Pr[x \in (\Omega^{j-1} \cap \pi_l)|x \in (\Omega^{j-2} \cap \pi_l)]\} \cdot \{1 - \Pr[x \in (\Omega^{j-2} \cap \pi_l)|x \in (\Omega^{j-3} \cap \pi_l)]\} \cdot \Pr[x \in (\Omega^{j-3} \cap \pi_l)] \\
&= \{1 - \Pr[x \in (\Omega^{j-1} \cap \pi_l)|x \in (\Omega^{j-2} \cap \pi_l)]\} \cdots \{1 - \Pr[x \in (\Omega^{1} \cap \pi_l)|x \in (\Omega^{0} \cap \pi_l)]\} \cdot \Pr[x \in (\Omega^{0} \cap \pi_l)] \\
&= \prod_{s=1}^{j-1} \{1 - \Pr[x \in (\Omega^{s} \cap \pi_l)|x \in (\Omega^{s-1} \cap \pi_l)]\} = \prod_{s=1}^{j-1} (1 - P_l^s)
\end{aligned}
$$

therefore,

$$
\alpha_{\pi_l} = 1 - P_l^j \prod_{s=1}^{j-1} (1 - P_l^s), \quad l = 1, 2, \ldots, g, \quad C(j) = l
$$

Appendix C: The Derivation of Equation (6)

$$
\begin{aligned}
M_{kl} &= \Pr\{\text{sample in group } k \text{ to be classified as group } l \text{ that is identified at step } j\} \\
&= \Pr[x \in (R^j \cap \pi_k), C(j) = l] \\
&= \Pr[x \in (R^j \cap \pi_k), C(j) = l|x \in \Omega^{j-1}] \cdot \Pr[x \in (\Omega^{j-1} \cap \pi_k\ ] \\
&\quad + \underbrace{\Pr[x \in (R^j \cap \pi_k), C(j) = l|x \in R^{j-1}]}_{0} \cdot \Pr[x \in (R^{j-1} \cap \pi_k)] \\
&= \Pr[x \in (R^j \cap \pi_k), C(j) = l|x \in \Omega^{j-1}] \cdot \Pr[x \in (\Omega^{j-1} \cap \pi_k)]
\end{aligned}
$$

where

$$
\begin{aligned}
\Pr[x \in (\Omega^{j-1} \cap \pi_k)] &= \Pr[x \in (\Omega^{j-1} \cap \pi_k)|x \in (\Omega^{j-2} \cap \pi_k)] \cdot \Pr[x \in (\Omega^{j-2} \cap \pi_k)] \\
&\quad + \underbrace{\Pr[x \in (\Omega^{j-1} \cap \pi_k)|x \in (R^{j-2} \cap \pi_k)]}_{0} \cdot \Pr[x \in (R^{j-2} \cap \pi_k)] \\
&= \Pr[x \in (\Omega^{j-1} \cap \pi_k)|x \in (\Omega^{j-2} \cap \pi_k)] \cdot \Pr[x \in (\Omega^{j-2} \cap \pi_k)] \\
&= \{1 - \Pr[x \in (\Omega^{j-1} \cap \pi_k)|x \in (\Omega^{j-2} \cap \pi_k)]\} \cdot \Pr[x \in (\Omega^{j-2} \cap \pi_k)] \\
&= \{1 - \Pr[x \in (\Omega^{j-1} \cap \pi_k)|x \in (\Omega^{j-2} \cap \pi_k)]\} \cdot \{1 - \Pr[x \in (\Omega^{j-2} \cap \pi_k)|x \in (\Omega^{j-3} \cap \pi_k)]\} \cdot \Pr[x \in (\Omega^{j-3} \cap \pi_k)] \\
&= \{1 - \Pr[x \in (\Omega^{j-1} \cap \pi_k)|x \in (\Omega^{j-2} \cap \pi_k)]\} \cdots \{1 - \Pr[x \in (\Omega^{1} \cap \pi_k)|x \in (\Omega^{0} \cap \pi_k)]\} \cdot \Pr[x \in (\Omega^{0} \cap \pi_k)] \\
&= \prod_{s=1}^{j-1} \{1 - \Pr[x \in (\Omega^{s} \cap \pi_k)|x \in (\Omega^{s-1} \cap \pi_k)]\} = \prod_{s=1}^{j-1} (1 - P_k^s)
\end{aligned}
$$

$$
\Rightarrow M_{kl} = P_k^j \prod_{s=1}^{j-1} (1 - P_k^s), \quad C(j) = l, j \le g
$$

Similarly, if $C(g+1)=l$, we have

$$M_{kl}=\prod_{s=1}^{g}(1-P_k^s)$$

References

[1] Grogan, R., 2002, “High Speed Stamping Process Improvement Through Force and Displacement Monitoring,” Helm Instrument Company Technical Report.

[2] Knussmann, K. D., and Rose, C., 1993, “Signature-Based Process Control (SbPC™),” Signature Technologies Technical Report, pp. 311–325.

[3] Auto Body Consortium, 1997, “Total Tonnage Signature Decomposition for a Transfer/Progressive Die Process Analysis of Station-by-Station Test at A. J. Rose 250 Ton Press,” NIST Advanced Manufacturing Program: NZS Technical Report No. NZS-204.

[4] Jin, J., 2004, “Individual Station Monitoring Using Press Tonnage Sensors for Multiple Operation Stamping Processes,” ASME J. Manuf. Sci. Eng., **126**(1), pp. 83–90.

[5] Zhou, S., Sun, B., and Shi, J., 2006, “An SPC Monitoring System for Cycle-Based Waveform Signals Using Haar Transform,” IEEE. Trans. Autom. Sci. Eng., **3**(1), pp. 60–72.

[6] Ge, M., Du, R., and Xu, Y., 2004, “Hidden Markov Model Based Fault Diagnosis for Stamping Processes,” Mech. Syst. Signal Process., **18**(2), pp. 391–408.

[7] Kim, J., Huang, Q., Shi, J., and Chang, T.-S., 2006, “Online Multichannel Forging Tonnage Monitoring and Fault Pattern Discrimination Using Principal Curve,” ASME J. Manuf. Sci. Eng., **128**, pp. 944–950.

[8] Zhou, S., and Jin, J., 2005, “An Unsupervised Clustering Method for Cycle-Based Waveform Signals in Manufacturing Processes,” IIE Trans., Qual. Reliab. Eng., **37**, pp. 569–584.

[9] Duda, R. O., Hart, P. E., and Stork, D. G., 2000, *Pattern Classification*, 2nd ed., Wiley-Interscience, New York.

[10] Vapnik, V., 1995, *The Nature of Statistical Learning*, Springer-Verlag, New York.

[11] Hsu, C.-W., and Lin, C.-J., 2002, “A Comparison of Methods for Multiclass Support Vector Machines,” IEEE Trans. Neural Netw., **13**(2), pp. 415–425.

[12] Vapnik, V., 2000, *Advances in Neural Information Systems*, MIT, Reading, MA.

[13] Zhu, J., Zou, H., Rosset, S., and Hastie, T., 2005, “Multi-Class Adaboost,” http://www.stat.lsa.umich.edu/~jizhu/pubs/

[14] Safavian, S. R., and Landgrebe, D., 1991, “A Survey of Decision Tree Classifier Methodology,” IEEE Trans. Syst. Man Cybern., **21**(3), pp. 660–674.

[15] Wang, Y. Y., and Li, J., 2008, “Feature Selection Ability of the Decision-Tree Algorithm and the Impact of Feature Selection/Extraction on Decision-Tree Results Based on Hyperspectral Data,” Int. J. Remote Sens., **29**, pp. 2993–3010.

[16] Hsu, W. H., 2004, “Genetic Wrappers for Feature Selection in Decision Tree Induction and Variable Ordering in Bayesian Network Structure Learning,” Inf. Sci., **163**, pp. 103–122.

[17] Kohavi, R., and John, G. H., 1997, “Wrappers for Feature Subset Selection,” Artif. Intell., **97**, pp. 273–324.

[18] Wang, A., and Liu, J., 2008, “An Improved Multiclassifier for Soft Fault Diagnosis of Analog Circuits,” *Trends in Intelligent Systems and Computer Engineering*, Springer, New York.

[19] Jackson, J. E., 1991, *A User’s Guide to Principal Components*, Wiley, New York.

[20] Jin, J., and Shi, J., 2005, “Press Tonnage Signal Decomposition and Validation for Transfer or Progressive Die Processes,” ASME J. Manuf. Sci. Eng., **127**, pp. 231–235.

[21] Johnson, R., and Wichern, D., 1998, *Applied Multivariate Statistical Analysis*, Wiley, New York.