

하이브리드 데이터마이닝을 이용한 지능형 이상 진단 시스템[†] Intelligent Fault Diagnosis System Using Hybrid Data Mining

백준걸*, 허준**

* 인덕대학 산업시스템경영과 ** 고려대학교 산업시스템정보공학과

Abstract

The high cost in maintaining complex manufacturing process makes it necessary to enhance an efficient maintenance system. For the effective maintenance of manufacturing process, precise fault diagnosis should be performed and an appropriate maintenance action should be executed. This paper suggests an intelligent fault diagnosis system using hybrid data mining. In this system, the rules for the fault diagnosis are generated by hybrid decision tree/genetic algorithm and the most effective maintenance action is selected by decision network and AHP. To verify the proposed intelligent fault diagnosis system, we compared the accuracy of the hybrid decision tree/genetic algorithm with one of the general decision tree learning algorithm(C4.5) by data collected from a coil-spring manufacturing process.

1. 서론

제조 시스템에서 예기치 않은 공정(process) 이상(fault)의 발생은 시스템의 생산 능력과 제품의 품질을 저하시키고 막대한 손실 비용을 초래하는 원인이 되므로 공정 상태를 정확히 진단(diagnosis)하고 최대한 빠른 시간에 이상 상황 발생 원인을 분석하여 그에 따른 효율적인 보전(maintenance)을 수행함으로써 공정의 신뢰성(reliability)을 향상시키는 일은 매우 중요하다. [Lee, 1996]

일반적으로 제조 시스템에서 공정 이상의 발생 빈도를 줄이기 위한 방법으로는 일정한 시간 간격을 두고 공정 상태를 점검한 후 적절한 보전을 수행하는 시간 기반 예방 보전(time-based preventive maintenance)이 수행되고 있지만 이러한 시간기반 예방 보전은 불필요한 보전 작업을 유발함으로써 공정 가동률을 저하시키고 이에 따른 손실 비용을 발생시키는 단점을 지닌다. 따라서 최근 들어 시간기반 예방 보전의 단점을 보완하기 위한 방법으로 현재의 공정 상태를 진단하고 진단된 공정 상태를 기반으로 보전 작업의 실시 유무를 결정하는 상태기반 보전(condition-based maintenance)

이 수행되고 있다. [Boutillier *et al.*, 2000] 그러나 현재 대부분의 제조 시스템에서는 공정 이상의 발생 원인 분석을 단순히 작업자의 경험과 과거의 기록에만 의존하기 때문에 정확한 공정 이상 발생 원인 분석과 이에 따른 효율적인 보전 정책을 수립하는 것이 불가능하다. 또한 효율적인 공정 관리를 수행하기 위해서는 이상 상황의 발생을 사전에 예방하는 것도 중요하지만, 발생된 공정 이상에 대해 최소의 비용으로 신속하고 정확한 보전을 수행하는 일도 매우 중요하다.

따라서 본 연구에서는 제조 시스템으로부터 수집되어지는 공정 데이터를 기반으로 하이브리드 데이터마이닝(hybrid data mining)을 이용하여 발생 가능한 공정 이상의 유형을 학습하고, 현재의 공정 상태를 기반으로 이상 상황의 발생 여부를 진단할 수 있는 지능형 이상 진단 시스템을 제시하고자 한다. 또한 본 연구에서 제시하는 지능형 이상 진단 시스템은 이상 상황 발생이 진단되었을 경우, 이에 따른 효율적인 보전을 수행할 수 있도록 지원하기 위해 이상 상황의 발생 원인을 분석하고 분석된 원인에 따라 최소의 비용으로 효율적인 보전 작업(maintenance action)을 수행할 수 있는 방법을 제시한다. 본 연구에서 제안하는 하이브리드 데이터마이닝을 이용한 지능형 이상 진단 시스템은 <그림 1>과 같은 구조를 지닌다.

<그림 1>에서 제시한 하이브리드 데이터마이닝을 이용한 지능형 이상 진단 시스템은 우선 제조 시스템으로부터 실시간으로 수집되어지는 공정 상태 데이터를 데이터베이스에 저장하고, 저장된 공정 상태 데이터를 기반으로 학습 모듈(learning module)에 의해 공정 이상의 유형에 대한 학습을 수행한다. 본 연구에서 제안하는 지능형 이상 진단 시스템에서의 학습 모듈을 크게 두 부분으로 나누어 설명될 수 있다. 첫째, 학습 속도를 향상시키기 위해 학습에 사용될 속성(attributes)을 선택(selection)하는 단계로서 본 연구에서는 신경망(neural network)을 이용하여 학습에 사용되어질 속성을 선택하는 방법을 제시한다. 둘째, 학습 속성 선택에 의해 선택되어진 속성을 기반으로 공정 이상의 유형별 학습을 수행하는 단계로서 본 연구에서는 의사결정트리(decision tree)와 유전자 알고리즘(genetic algorithm)을 이용한 하이브리드 학습(hybrid learning) 방법을 제시한다. 하이브리드 데이터마이닝을

[†]이 논문은 2004년도 한국학술진흥재단의 지원에 의해 연구되었음. (KRF-2004-003-D00459)

이용한 구체적인 학습 방법은 3장에서 자세히 설명한다. 또한 본 연구에서 제시한 지능형 이상 진단 시스템은 학습 모듈에 의해 학습된 규칙(rules)을 기반으로 현재의 공정 상태를 진단하고, 만약 현재 공정 상태가 이상 상황으로 진단되었을 경우 효율적인 보전 작업을 선택하기 위해 AHP(Analytic Hierarchy Process)를 이용한 효율적인 의사결정 방법을 제시한다. AHP를 이용한 보전 작업 선택 방법에 대한 구체적인 설명은 4장에서 자세히 기술한다.

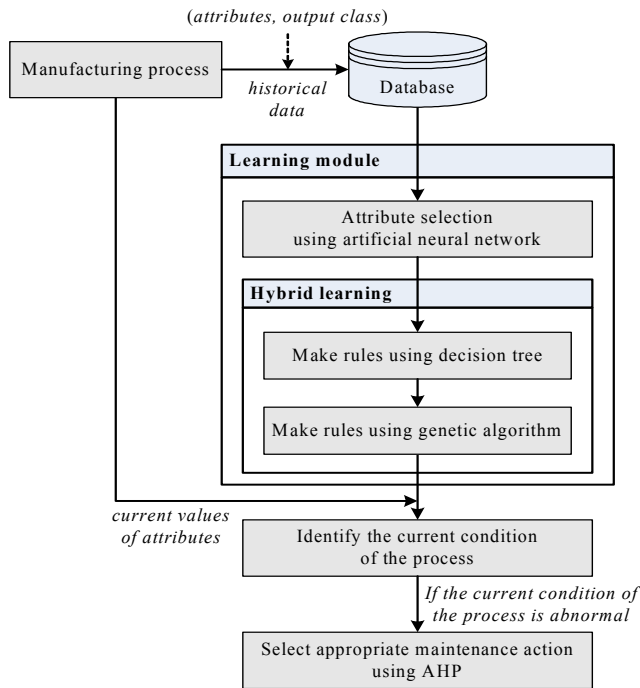


그림 1. 지능형 이상 진단 시스템.

2. 신경망을 이용한 학습 속성 선택

일반적으로 학습을 위해 사용되는 데이터 집합에는 부적절하고 무의미한 데이터가 포함되어 있으며, 이러한 부적절한 데이터는 학습의 효과를 저하시키는 원인이 된다. 따라서 데이터 집합으로부터 결과에 중요한 영향을 미치는 데이터 속성만을 선택하여 학습을 수행함으로써 학습된 결과의 정확도를 높이는 것이 중요하다. 또한 학습에 사용되는 데이터의 속성 수가 줄어든다면 학습을 위해 소요되는 계산 시간이 단축될 수 있다는 장점을 지닌다.

따라서 본 연구에서는 하이브리드 데이터마이닝을 이용한 학습의 효과를 증대시키기 위해서 Onnia et al.(2001)가 제안한 FSA(feature selection algorithm)를 이용한 학습 속성 선택 방법을 사용하고자 한다.

FSA는 FFNN(feed-forward neural network)을 이용하여 학습에 사용될 속성을 선택하는 방법으로써, FFNN에서 은닉층(hidden layer)의 학습을 위해서 다음과 같은 시그모이드 함수(sigmoid function)가 사용된다. [Onnia et al., 2001]

$$f(y) = \frac{2}{1 + \exp(-2 \cdot y)} - 1$$

where $y = \text{input value for the neuron}$

또한 결과층(output layer)의 뉴런(neuron)을 위한 활성화 함수(activation function)는 다음과 같다.

$$f(y) = y$$

FFNN을 학습(training)시키기 위한 알고리즘은 Levenberg-Marquardt 최적화 방법(optimization method)을 이용한 역전파 알고리즘(back propagation algorithm)을 사용한다. 그리고 학습 속성을 선택하기 위해서는 다음과 같은 분류 정확도(CA: Classification Accuracy)를 이용하여 가장 높은 분류 정확도를 갖는 속성 집합을 학습에 필요한 속성으로 선택한다.

$$CA = 100\% \cdot \frac{\text{test_cases}_+}{\text{test_cases}_{total}}$$

단, 위 식에서 test_cases_+ 는 성공적으로 분류된 테스트 케이스(test case)의 수를 나타내고, $\text{test_cases}_{total}$ 은 모든 테스트 케이스의 수를 나타낸다.

3. 하이브리드 데이터마이닝을 이용한 학습

본 연구에서 제시하는 하이브리드 데이터마이닝을 이용한 지능형 이상 진단 시스템에서는 수집된 공정 상태 데이터를 기반으로 공정 이상 유형별 학습을 수행하기 위해 <그림 2>에서 제시하는 바와 같은 하이브리드 학습 방법을 사용한다. <그림 2>에서 제시하는 하이브리드 학습 방법은 일차적으로 수집된 공정 상태 데이터와 학습 속성 선택에 의해 선택된 속성을 기반으로 의사결정트리(decision tree)를 이용한 학습을 수행한다. 본 연구에서 기본 학습 방법으로써 의사결정트리를 사용한 이유는 의사결정트리에 의해 학습된 결과가 사람이 이해하기 쉬운 규칙(rules) 형태로 표현될 수 있기 때문이다. 그러나 의사결정트리를 이용한 학습은 단말 노드(leaf node)의 결과(output class) 값이 오직 하나의 값으로 표현되기 때문에 경우에 따라 의사결정트리의 정확도(accuracy rate)를 감소시키는 원인이 된다. 따라서 본 연구에서는 의사결정트리에 의해 학습된 단말 노드의 정확도가 정해진 기준치(threshold)에 미치지 못하는 경우, 해당 단말 노드의 정확도를 향상시키기 위해 유전자 알고리즘(genetic algorithm)을 이용한 학습방법을 추가적으로 수행하는 방법을 제시하고자 한다.

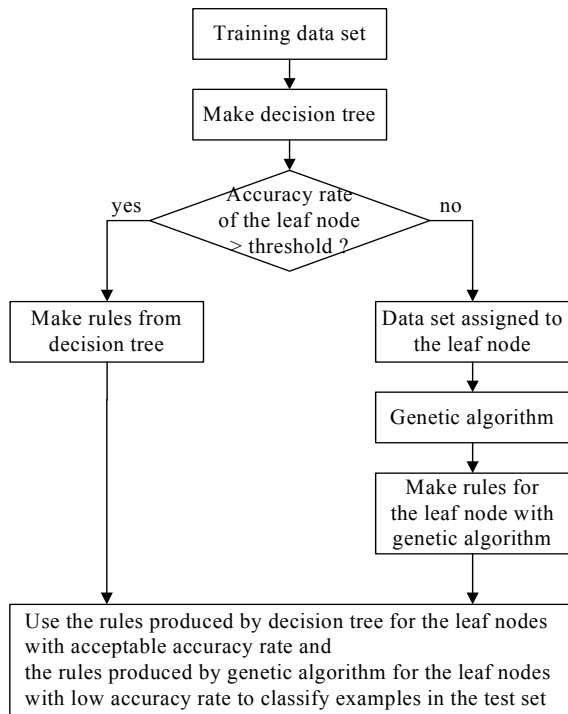


그림 2. 하이브리드 학습(hybrid learning).

3.1 의사결정트리(decision tree)

의사결정트리는 분류(classification) 또는 예측(expectation)을 위한 의사결정 규칙(rules)을 트리 구조로 도식화하여 표현할 수 있는 데이터마이닝 학습 방법으로써 분류 또는 예측을 위한 추론(inference) 규칙이 사람이 쉽게 이해할 수 있는 형태로 표현될 수 있다는 장점을 지닌다. [Berry and Linoff, 1997] 의사결정트리를 이용한 대표적인 학습 방법으로는 C4.5[Quinlan, 1993]가 주로 사용되어진다. C4.5는 의사결정트리를 구성하는 의사결정 노드(decision node)에서 사용되어질 속성을 선택하기 위해서 정보 이득(information gain) 척도를 이용하여 정보 이득이 가장 큰 속성을 해당 의사결정 노드의 속성으로 사용한다. 정보 이득을 이용한 의사결정 노드의 속성 선택은 의사결정트리를 생성하기 위해 필요한 의사결정 노드의 수를 최소화할 수 있다는 장점을 지닌다. [Han and Kamber, 2001]

본 연구에서 제시하는 하이브리드 데이터마이닝을 이용한 지능형 이상 진단 시스템의 학습 모듈에서는 수집된 공정 상태 데이터를 기반으로 C4.5를 이용하여 공정 상태를 추론하기 위한 학습을 수행하였으며, 구체적인 학습 결과는 5장에서 자세히 기술한다.

3.2 유전자 알고리즘 (genetic algorithm)

의사결정트리를 이용한 학습은 비교적 빠른 시간에 사용자가 이해하기 쉬운 형태로 추론을 위한 규칙을 도출할 수 있다는 장점을 제공하지만 추론의 결과를 나타내는 단말 노드(leaf node)가 오직 하나의 결과 값(output class)을 나타낼 수 있으므로 단말 노드에 속한 데이터의 결과 값이 혼재되어 있을 경우 추론 규칙의 정확도(accuracy rate)가 저하될 수 있다는 문제점을

지닌다. 본 연구에서는 이러한 문제점을 해결하기 위한 방법으로 유전자 알고리즘을 이용하여 의사결정트리의 정확도를 보정할 수 있는 하이브리드 학습 방법을 제시하고자 한다. 본 연구에서 제안하는 하이브리드 학습은 <그림 2>에서 제시한 바와 같이 의사결정트리에 의해 학습된 결과에서 단말 노드의 정확도가 주어진 기준치(threshold)에 미달되는 경우 해당 단말 노드에 할당된 데이터 집합을 기반으로 유전자 알고리즘을 이용하여 공정 상태 추론을 위한 추가적인 규칙을 생성할 수 있는 방법을 제시한다.

의사결정트리는 추론의 결과를 나타내는 단말 노드가 오직 하나의 결과 값을 나타낼 수 있다는 단점을 지닌 반면에, 유전자 알고리즘은 모든 가능한 결과 값에 대한 규칙을 정의할 수 있다는 장점을 지닌다. 즉, 유전자 알고리즘은 모든 가능한 결과 값에 대한 최적의 염색체(genome)를 선택하는 학습을 수행할 수 있고, 최종적으로 선택되어진 염색체를 이용하여 모든 가능한 결과 값에 대한 추론 규칙을 정의할 수 있다. [Carvalho and Freitas, 2004] 본 연구에서는 제시하는 하이브리드 학습 모듈에서 유전자 알고리즘을 이용한 학습을 수행하기 위해서는 다음 사항에 대한 구체적인 정의가 필요하다.

- 유전자 알고리즘을 이용한 학습을 수행하기 위한 염색체(genome) 설계.
- 초기 염색체 집단(population) 생성.
- 유전자 알고리즘의 학습 절차

3.2.1 염색체(genome) 설계

본 연구에서는 유전자 알고리즘에서 사용되어질 염색체 구조를 <그림 3>과 같이 정의한다.

본 연구에서 제안하는 유전자 알고리즘을 위해 사용되어지는 염색체는 <그림 3>에서 나타낸 바와 같이 n 개의 유전자(gene)로 구성되며, n 은 의사결정트리에서 정확도가 기준치(threshold)에 미달되는 단말 노드에서 해당 단말 노드의 상위 노드(의사결정 노드)에 사용되어진 속성들을 제외한 나머지 속성의 수를 나타낸다.

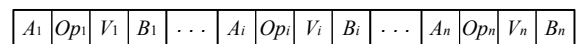


그림 3. 염색체 구조.

염색체를 구성하는 n 개의 유전자는 $\langle A_i, Op_i, V_i, B_i \rangle$ 집합으로 구성되며, A_i 는 i 번째 속성을 나타내고, Op_i 는 A_i 를 위해 사용되어지는 연산자(operator)를 나타낸다. 본 연구에서는 Op_i 로 사용될 수 있는 연산자로서 $\geq, <, =$ 를 사용한다. 또한 V_i 는 속성 A_i 를 위해 사용되어지는 분기점(splitting point)을 나타내고, B_i 는 활성 비트(active bit)으로써 해당 유전자가 학습을 위해 사용되어지는지를 나타내는 식별자로 사용되는 값을 나타낸다. 활성 비트 B_i 에 대한 추가적인 설명은 3.2.3 유전자 알고리즘 학습 절차에서 자세히 기술한다. 염색체를 구성하는 유전자를 예를 들어 설명하면, 만약 속성 "coiling_time"이 i 번째 속성으로 사용되어지고 이때의 규칙이 "coiling_time < 107"로 표현되어진다면, Op_i 는 "<", V_i 는 "107"로 정의되어진다.

3.2.2 초기 염색체 집단 생성

유전자 알고리즘을 이용한 학습을 수행하기 위해서는 먼저 초기 염색체 집단(population)을 생성해야 한다. 본 연구에서는 초기 염색체 집단의 크기(p)를 사용자가 적절히 선택하는 방법을 사용하며, 이때 사용되는 초기 염색체 집단의 크기에 따라 유전자 알고리즘의 학습 효율이 달라질 수 있다. 초기 염색체 집단의 크기에 따른 유전자 알고리즘의 학습 효율 분석은 5장에서 자세히 기술한다. 초기 염색체 집단으로 사용되는 각각의 염색체를 구성하는 유전자의 A 는 유전자 알고리즘을 이용한 학습이 적용되는 의사결정트리의 단말 노드 위치에 따라 자동으로 결정된다. 또한 Op 는 가능한 연산자 중에서 무작위(random)로 선정되며, B 는 <그림 5>에서 제시한 활성 비트 할당 알고리즘에 의해 결정되어진다. 마지막으로 V 는 다음과 같은 세 가지 할당 규칙이 무작위로 적용되어 값이 결정되어진다.

- 정보 이득(information gain) 척도를 이용하여 선택되어진 분기점(splitting point)을 V_i 값으로 할당한다.
- 데이터 집합에 속한 데이터들이 갖는 해당 속성(A_i)의 중간 값(median value)을 V_i 값으로 할당한다.
- 속성 A_i 가 가질 수 있는 값의 범위 내에서 무작위로 선정된 값을 V_i 값으로 할당한다.

3.2.3 유전자 알고리즘 학습 절차

본 연구에서 제시하는 유전자 알고리즘은 위에서 기술한 초기 염색체 집단을 기반으로 <그림 4>에서 제시한 절차에 의해 학습을 수행한다.

```

Step 1. Initialize population
Step 2. Evaluate the initial population with fitness function.
Step 3. WHILE(termination condition)
    /* Create a new population */
    Step 3.1 Select the  $(1-r)p$  genome which has the highest fitness function value.
    Step 3.2 Crossover
    Step 3.3 Mutation
    Step 3.4 Adjust active bit in each gene for each genome.
    /* Evaluate the new population */
    Step 3.5 Evaluate the new population with fitness function.
    
```

그림 4. 유전자 알고리즘 학습 절차.

<그림 4>에서 제시한 유전자 알고리즘 학습 절차에서 사용되는 적합도 함수(fitness function)는 다음과 같이 정의된다. [Hand, 1997]

$$fitness = \left(\frac{TP}{TP + FN} \right) \times \left(\frac{TN}{FP + TN} \right)$$

만약, P 가 주어진 규칙에 의해 예측되어진 결과 클래스(output class)를 나타내고, N 은 주어진 규칙에 의해 예측된 결과 클래스와 다른 결과 클래스를 나타내는 값이라고 가정하면, TP 는 결과 클래스 P 를 갖는

데이터가 정확하게 P 로 예측되어진 데이터의 수를 나타내고, FP 는 결과 클래스 N 을 갖는 데이터가 P 로 잘못 예측되어진 데이터의 수를 나타낸다. 또한 FN 은 결과 클래스 P 를 갖는 데이터가 N 으로 잘못 예측되어진 데이터의 수를 나타내고, TN 은 결과 클래스 N 을 갖는 데이터가 정확하게 N 으로 예측되어진 데이터의 수를 나타낸다.

Step 3.1은 이전 세대에 생성된 염색체 집단으로부터 다음 세대로 계승되어질 염색체를 선택하는 단계로서, r 은 교배(crossover)에 의해 대체될(replaced) 염색체의 비율(fraction)을 나타내고, p 는 초기 염색체 집단의 크기를 나타낸다. 즉, Step 3.1에 의해 이전 세대의 염색체 집단 중에서 가장 높은 적합도 함수 값을 갖는 $(1-r)p$ 개의 염색체가 다음 세대를 위한 염색체로 계승되어진다.

Step 3.2는 다음 세대의 염색체 집단에 사용될 새로운 염색체를 교배에 의해 생성하는 단계로써 $rp/2$ 쌍(pairs)의 염색체가 일점(one point) 교배에 의해 새로운 유전자를 갖는 염색체로 생성되어진다.

Step 3.3은 유전자 알고리즘이 국소 탐색(local search)에 빠지는 오류를 방지하기 위해 사용하는 돌연변이(mutation) 단계로써 아주 작은 확률(m)을 사용하며 돌연변이에 의해 새로운 염색체를 생성하는 단계를 나타낸다. 돌연변이 확률(m)에 따른 유전자 알고리즘의 효율 분석은 5장에서 자세히 기술한다.

Step 3.4는 Step 3.1~Step 3.3.에서 의해 생성된 다음 세대를 위한 새로운 염색체 집단의 각각의 염색체를 구성하는 유전자의 활성 비트를 조정하는 단계로써 <그림 5>에서 제시한 활성 비트 할당 알고리즘에 의해 활성 비트를 조정한다.

```

Step 1. Compute  $InfoGain(cond_i)$  for each  $i$ th gene in each genome belong to population.
Step 2. Repeat Step 2.1 ~ Step 2.2 until the number of iteration reaches to the number of genes.
    Step 2.1 Generate random value( $rv$ ) with  $U(0, 1)$ .
    Step 2.2 IF( $rv < InfoGain(cond_i)$ ) THEN Set  $B_i$  as 1. ELSE Set  $B_i$  as 0.
    
```

그림 5. 활성 비트 할당 알고리즘.

<그림 5>에서 제시한 활성 비트 할당 알고리즘에서 사용된 $InfoGain(cond_i)$ 은 다음과 같이 정의된다.

$$InfoGain(cond_i) = Info(G) - Info(G | cond_i)$$

where

$$Info(G) = - \sum_{j=1}^c \left(\frac{|G_j|}{|T|} \times \log_2 \left(\frac{|G_j|}{|T|} \right) \right)$$

$$Info(G | cond_i) = - \left(\frac{|V_{i1}|}{|T|} \right) \sum_{j=1}^c \left(\left(\frac{|V_{ij1}|}{|V_{i1}|} \right) \times \log_2 \left(\frac{|V_{ij1}|}{|V_{i1}|} \right) \right) - \left(\frac{|V_{i2}|}{|T|} \right) \sum_{j=1}^c \left(\left(\frac{|V_{ij2}|}{|V_{i2}|} \right) \times \log_2 \left(\frac{|V_{ij2}|}{|V_{i2}|} \right) \right)$$

단, 위 식에서 G 는 결과 클래스 속성을 나타내고, c 는 G 가 가질 수 있는 값의 개수를 나타낸다. 그리고

$cond$ 는 i 번째 유전자($\langle A_j, Op_i, V_i \rangle$)에 의해 정의되는 조건을 의미한다. 또한 $|G_j|$ 는 j 번째 결과 클래스 값을 갖는 데이터의 수를 나타내고, $|T|$ 는 전체 데이터의 수를 나타낸다. $|V_i|$ 는 i 번째 유전자에 의해 정의되는 조건을 만족하는 데이터의 수를 의미하고, $|V_{ij}|$ 는 i 번째 유전자에 의해 정의되는 조건을 만족하고 j 번째 결과 클래스 값을 갖는 데이터의 수를 의미한다. $| \neg V_i |$ 는 i 번째 유전자에 의해 정의되는 조건을 만족하지 않는 데이터의 수를 의미하고, $| \neg V_{ij} |$ 는 i 번째 유전자에 의해 정의되는 조건을 만족하지 않고 j 번째 결과 클래스 값을 갖는 데이터의 수를 의미한다.

4. AHP를 이용한 보전 작업 선택

본 연구에서 제시한 지능형 이상 진단 시스템은 3장에서 기술한 하이브리드 데이터마이닝에 의해 학습된 규칙을 기반으로 현재의 공정 상태를 진단하고, 만약 현재 공정 상태가 이상 상황으로 진단되었을 경우 진단된 이상 상황 유형에 따른 효율적인 보전 작업(maintenance action)을 선택할 수 있는 방법을 제시한다. 본 연구에서 제시하는 보전 작업 선택 방법은 진단된 공정 이상을 해결할 수 있는 가능한 보전 작업들을 찾기 위해 의사결정 네트워크(decision network)를 이용하여 모든 가능한 대안들을 도식적으로 기술하고, 의사결정 네트워크로부터 유추된 대안들 중에서 AHP를 이용하여 최적의 대안을 선택할 수 있는 방법을 제시한다.

4.1 의사결정 네트워크(Decision Network)

하이브리드 데이터마이닝에 의해 학습된 규칙을 기반으로 현재의 공정 상태가 이상 상황으로 진단되면, 진단된 공정 이상을 해결할 수 있는 가능한 보전 작업들을 찾기 위해 의사결정트리와 유전자 알고리즘에 의해 학습된 규칙을 기반으로 모든 가능한 대안들을 도식적으로 기술하는 의사결정 네트워크를 <그림 6>에서 기술한 의사결정 네트워크 생성 절차에 의해 작성한다.

- Step 1.** Find detected abnormal node.
- Step 2.** Find normal nodes.
- Step 3.** Create paths from the abnormal node to each normal nodes.
Repeat Step 3.1 ~ 3.3 by the number of normal nodes founded in Step 2.
- Step 3.1** Set the current location of abnormal node and targeted normal node.
- Step 3.2** Find the split node for the abnormal node and the targeted normal node.
- Step 3.3** Add the arc from the split node to the targeted normal node.
- Step 4.** Construct decision network

그림 6. 의사결정 네트워크 생성 절차.

의사결정트리와 유전자 알고리즘에 의해 학습된 규칙을 기반으로 의사결정 네트워크를 작성하는 구체적인 사례는 5장에서 자세히 기술한다.

4.2 AHP를 이용한 보전 작업 선정

AHP(Analytical Hierarchy Process)를 이용한 의사결정 방법은 복잡한 문제를 계층화하여 주요 요인과 세부 요인들로 나누고, 이러한 요인들에 대한 쌍대 비교(pairwise comparison)를 통해 중요도를 산출하여 상대적 중요도 또는 선호도를 체계적인 비율 척도(ratio scale)로 계량화함으로써 정량적인 분석이 가능하도록 지원한다. AHP를 이용하여 가능한 대안들을 평가하기 위해서는 상대적인 중요도를 나타내는 비율 척도가 정의되어야 하는데 일반적으로 <표 1>에서 제시한 비율 척도가 주로 사용된다. 그리고 각각의 요인들에 대한 상대적 중요도의 가중치는 일반적으로 0에서 1사이의 값이 사용된다. [Satty, 1980]

표 1. 비율 척도(ratio scale).

척도	정의	설명
1	동급 (equal)	두 가지 요소가 인접한 상위 목표의 기준에서 볼 때 똑같이 중요
3	약간 중요 (moderate)	한 요소가 다른 요소에 비해 약간 중요
5	중요 (strong)	한 요소가 다른 요소에 비해 중요
7	매우 중요 (very strong)	한 요소가 다른 요소에 비해 매우 중요
9	절대적 중요 (extreme)	한 요소가 다른 요소에 비해 절대적으로 중요
2, 4, 6, 8	중간 척도들	위의 척도들의 중간 값이 요구될 때

본 연구에서는 의사결정 네트워크에 의해 유추된 가능한 보전 작업 대안들을 AHP를 이용하여 평가함으로써 진단된 이상 상황에 대한 최적의 보전 작업을 선택할 수 있는 방법을 제시한다. AHP를 이용한 보전 작업 선택에 대한 구체적인 사례는 5장에서 자세히 기술한다.

5. 사례 분석

본 연구에서는 제조 시스템으로부터 수집된 공정 데이터를 기반으로 하이브리드 데이터마이닝을 이용하여 공정 이상 유형별 학습을 수행하고, 학습된 이상 상황 유형별 규칙을 기반으로 현재의 공정 상태를 진단하고, 만약 현재 공정 상태가 이상 상황으로 진단되었을 경우 AHP를 이용하여 최적의 보전 작업을 선택할 수 있는 하이브리드 데이터마이닝을 이용한 지능형 이상 진단 시스템을 제시하였다. 본 연구에서 제시한 지능형 이상 진단 시스템의 성능을 평가하기 위해 아래에 기술한 제조 시스템을 대상으로 구체적인 사례 분석을 수행하였다.

5.1 대상 시스템

본 연구에서 제시한 지능형 이상 진단 시스템의 구체적인 사례를 분석하기 위해 코일 스프링을 생산하는 제조 시스템으로 수집된 공정 데이터를 기반으로 구현한 사례를 제시한다.

본 연구의 대상이 되는 코일 스프링 제조 공정은 <그림 7>에서 기술한 바와 같은 공정을 지니고 있으며, <그림 7>에 표시된 데이터 수집 포인트에서 수집된 공정 데이터의 속성(attribute)은 <표 2>에서 자세히 기술하였다.

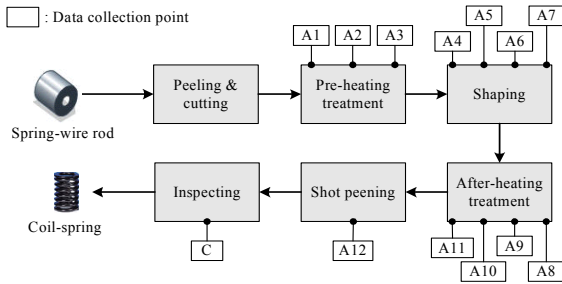


그림 7. 코일 스프링 제조 공정.

<표 2>는 코일 스프링 제조 공정으로부터 수집되어지는 공정 데이터의 속성과 결과 클래스를 기술한 것으로서, 12개의 속성(Ai, i=1,...,12)은 코일 스프링 제조 공정의 상태를 나타내고, 결과 클래스 C는 완성된 코일 스프링의 높이를 나타낸다. 공정 상태를 나타내는 속성들의 대부분이 사전 열처리(pre-heating treatment), 성형(shaping), 사후열처리(after-heating treatment) 공정에 집중되어 있는 이유는 해당 공정의 작업이 결과 클래스 값인 코일 스프링 높이에 가장 중요한 영향을 미치기 때문이다.

5.2 학습 속성 선택

<표 2>에서 기술한 12개의 속성 중에서 결과 클래스에 중요한 영향을 미치는 속성을 선택하기 위해 본 연구에서는 2장에서 기술한 FSA를 이용한 학습 속성 선택 방법을 사용한다. <그림 8>은 FSA에 의해 분류 정확도(CA)가 높은 속성들이 선택된 결과를 나타낸 그림으로써 12개의 속성 중에서 6개의 속성(A1, A3, A4, A5, A7,

A9)이 학습을 위한 속성으로 선택되어졌음을 나타내고 있다.

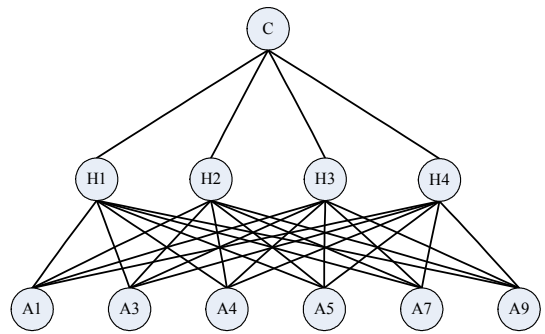


그림 8. FSA를 이용한 학습 속성 선택.

5.3 하이브리드 데이터마이닝을 이용한 학습

5.3.1 의사결정트리 생성

본 연구에서 제시하는 지능형 이상 진단 시스템은 학습 속성 선택 단계에서 선택된 속성을 기반으로 의사결정트리와 유전자 알고리즘이 혼합된 하이브리드 학습을 통해 공정 상태에 대한 학습을 수행한다.

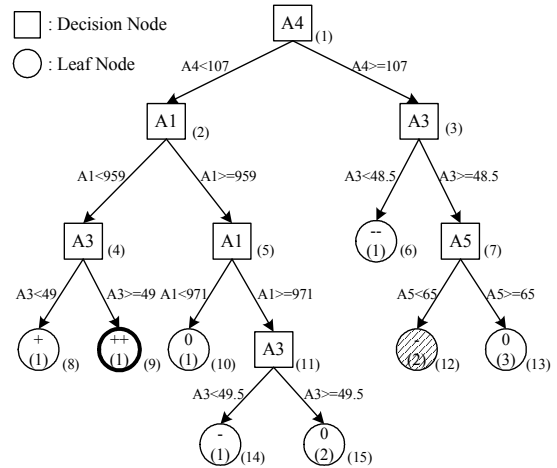


그림 9. 의사결정트리에 의한 학습 결과.

표 2. 코일 스프링 제조 공정에서의 공정 속성과 결과 클래스.

	Name	Description	Type	Range of value
Attributes	A1	Temperature of burner	Continuous	960±20 °C
	A2	Moving speed of working beam	Continuous	10.6±0.5 mm/min
	A3	Staying time in pre-heating station	Continuous	49.8±1 min
	A4	Coiling time	Continuous	106.6±1.5 Sec
	A5	Pressure of hydraulic power	Continuous	40 ~ 80 Kg/m ²
	A6	The status of the servo valve	Discrete	Normal/abnormal
	A7	Temperature of hydraulic fluid	Continuous	30 ~ 55 °C
	A8	Temperature of the oil in the direct quenching tank	Continuous	40 ~ 60 °C
	A9	Passing time in the direct quenching tank	Continuous	12.4±20 min
	A10	Temperature of the tempering	Continuous	420±10 °C
	A11	Passing time in tempering	Continuous	180±5 min
	A12	The status of sand for shot peening	Discrete	Normal/abnormal
Output class	C	Height of coil-spring	Discrete	--, -, 0, +, ++

<그림 9>는 하이브리드 데이터마이닝을 이용한 학습 단계 중 의사결정트리를 이용하여 학습을 수행한 결과를 나타내는 그림으로써, 본 연구에서는 C4.5 알고리즘에 기초한 SAS E-Miner를 사용하여 학습을 수행하였다.

5.3.2 유전자 알고리즘

의사결정트리를 이용한 학습은 학습의 결과를 나타내는 단말 노드(leaf node)가 오직 하나의 결과(output class) 값만을 나타냄으로써 단말 노드에 속한 데이터의 결과 값이 혼재되어 있을 경우 생성된 추론 규칙의 정확도(accuracy rate)가 저하될 수 있다는 문제점을 지닌다. 따라서 본 연구에서는 의사결정트리에 의해 학습된 결과에서 단말 노드의 정확도가 주어진 기준치(threshold)에 미달되는 경우 해당 단말 노드에 할당된 데이터 집합을 기반으로 유전자 알고리즘을 이용하여 공정 상태 추론을 위한 추가적인 규칙을 생성할 수 있는 방법을 제시하였다.

<그림 9>에서 기술한 의사결정 트리의 단말 노드들에 대한 정확도를 계산한 결과 빗금으로 표시된 단말 노드(12)의 정확도가 74%로 계산되었고, 이는 본 연구에서 정의한 기준치 80%에 미달되었다. 따라서 해당 단말 노드는 의사결정트리의 전체 정확도를 감소시키는 원인이 되므로, 해당 단말 노드에 할당된 데이터를 기반으로 유전자 알고리즘을 이용한 학습을 수행함으로써 학습 결과의 정확도를 높이는 작업을 수행한다. 본 연구에서는 3.2절에서 기술한 유전자 알고리즘 학습 절차에 의해 학습을 수행하였으며, 다음과 같은 네 가지 정책에서 설정한 파라미터 값에 의해 학습을 수행하였다. (p 는 초기 염색체 집단의 크기, r 은 교배에 의해 대체될 염색체의 비율, m 은 돌연변이 확률을 의미한다.)

- 정책 1 : $p = 10, r = 20\%, m = 1\%$
- 정책 2 : $p = 10, r = 20\%, m = 2\%$
- 정책 3 : $p = 20, r = 20\%, m = 1\%$
- 정책 4 : $p = 20, r = 20\%, m = 2\%$

<표 3>은 본 연구에서 제시한 하이브리드 학습의 성능을 평가하기 위해 학습에 사용된 데이터의 수에 따른 정확도(accuracy rate)를 측정된 결과로써 두 번째 열(DT)에 표시된 결과는 단순히 의사결정트리만을 이용하여 학습을 수행하였을 때의 정확도를 나타낸 값이고, 나머지 4개의 열(정책 1, 정책 2, 정책 3, 정책 4)에 표시된 결과는 본 연구에서 제시한 하이브리드 데이터마이닝에 의해 학습을 수행하였을 때의 정확도를 나타낸 값이다. 단, 각각의 정책은 위에서 기술한 바와 같이 유전자 알고리즘을 이용하여 학습을 수행할 때 설정되어지는 파라미터 값에 따라 분류된 정책을 의미한다.

<표 3>에서 나타난 바와 같이 본 연구에서 제안한 하이브리드 데이터마이닝을 이용한 학습이 단순히 의사결정트리만을 이용하여 학습을 수행한 결과보다 높은 정확도를 보장할 수 있음을 알 수 있고, 하이브리드 학습에서 유전자 알고리즘을 이용하여 학습을 수행할 때 정책 4에 의해 파라미터 값을 설정하는 것이 학습

의 정확도를 높일 수 있음을 알 수 있다.

표 3. 하이브리드 학습의 정확도 비교.

(단위: %)

데이터 수	DT	정책1	정책2	정책3	정책4
100	78.2	78.5	79.1	80.1	80.0
1000	81.2	81.3	81.4	81.8	81.9
5000	82.8	82.8	83.5	83.7	83.7
10000	84.2	84.5	84.5	85.0	85.2

따라서 본 연구에서는 정책 4에 의해 설정된 파라미터 값을 기반으로 유전자 알고리즘을 이용한 학습을 수행하였고, 정확도가 기준치에 미달되는 단말노드(12)에 대해 다음과 같은 규칙을 추가하였다.

IF((A<106.5) & (A9<12.3)) THEN 0(4).
 IF((A1≥106.5) & (A9<12.3)) THEN -(2).
 IF((A1<105) & (A7<41)) THEN --(2).

5.4 보전 작업 선정

본 연구의 대상이 되는 코일 스프링 공정에서 결과 클래스는 완성된 코일 스프링의 높이(height)를 나타내는 값으로써 --, -, 0, +, ++로 표현되는 5가지 종류의 값을 갖는다. 이 때 0은 코일 스프링의 높이가 정상 범위에 속한다는 것을 의미하고, -(+)는 코일 스프링의 높이가 정상 범위보다 약간 낮은(높은) 경우를 나타내며, --(++)는 코일 스프링의 높이가 정상 범위보다 매우 낮은(높은) 경우를 나타낸다. 일반적으로 코일 스프링의 수요자인 자동차 회사에서는 --와 ++ 값을 갖는 코일 스프링을 제외한 모든 코일 스프링을 납품받으므로 본 연구에서는 --와 ++ 값을 갖는 코일 스프링이 생산되는 경우를 공정 이상 상황으로 정의한다.

본 연구에서는 하이브리드 데이터마이닝에 의해 학습된 규칙을 기반으로 현재의 공정 상태를 진단하고, 만약 현재 공정 상태가 이상 상황으로 진단되었을 경우 진단된 이상 상황 유형에 따른 최적의 보전 작업을 선택하기 위해 의사결정 네트워크를 이용하여 모든 가능한 대안들을 도식적으로 기술하고, 의사결정 네트워크로부터 유추된 대안들 중에서 AHP를 이용하여 최적의 대안을 선택할 수 있는 방법을 제시하였다.

만약 현재의 공정 상태가 <그림 9>에서 진하게 표시한 단말 노드(9)에 의해 ++(1)로 진단되었다면, 진단된 이상 상황을 정상 상태(0)로 복구하기 위해 수행할 수 있는 가능한 보전 작업들을 <그림 6>에서 제시한 의사결정 네트워크 생성 절차에 의해 도식적으로 표현하고, 의사결정 네트워크에 의해 유추된 가능한 모든 대안들을 AHP를 이용하여 평가함으로써 최적의 보전 작업을 선택한다. 단말 노드(9)에 의해 ++(1)로 진단된 이상 상황을 정상 상태로 복구하기 위해 수행할 수 있는 가능한 보전 작업들을 도식적으로 표현한 의사결정 네트워크는 <그림 10>과 같다.

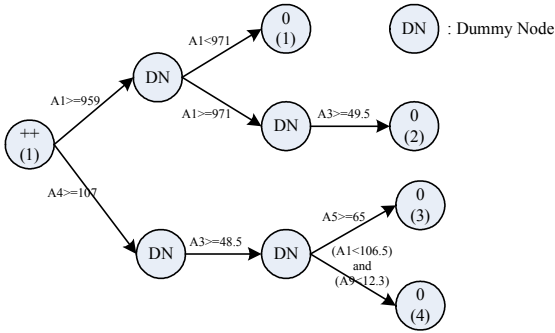


그림 10. ++(1)을 위한 의사결정 네트워크.

<그림 10>의 의사결정 네트워크로부터 유추된 가능한 보전 작업 대안들은 <표 4>와 같다. <표 4>에서 기술한 대안들 중에서 최적의 대안을 찾기 위해 본 연구에서는 AHP를 이용하여 최적의 보전 작업을 선정하였으며 대안 선정을 위해 필요한 요인별 쌍대 비교표는 <표 5>와 같다. 또한 <표 6>은 각각의 요인에 대해 대안별 쌍대 비교를 정의한 표이다.

본 연구에서는 <표 5>, <표 6>에 의해 정의된 쌍대 비교표를 기반으로 Expert Choice를 이용하여 AHP를 이용한 대안 분석을 수행하였으며, 분석 결과 대안 3(Maintenance action 3)이 최적의 보전 작업으로 선정되었다.

표 5. 주요 요인별 쌍대 비교표.

	속성 개수	작업 시간	비용
속성 개수	1	3	7
작업 시간	1/3	1	5
비용	1/7	1/5	1

표 6. 주요 요인에 대한 대안별 쌍대 비교표.

속성 개수				
	대안1	대안2	대안3	대안4
대안1	1	1/3	1/5	1/7
대안2	3	1	1/3	1/5
대안3	5	3	1	1/3
대안4	7	5	3	1
작업 시간				
	대안1	대안2	대안3	대안4
대안1	1	1/5	1/3	1/7
대안2	5	1	3	1/3
대안3	3	1/3	1	1/5
대안4	7	3	5	1
비용				
	대안1	대안2	대안3	대안4
대안1	1	1/3	3	1/5
대안2	3	1	5	1/3
대안3	1/3	1/5	1	1/7
대안4	5	3	7	1

6. 결론

본 연구에서는 제조 시스템으로부터 수집되어지는 공정 데이터를 기반으로 하이브리드 데이터마이닝(hybrid data mining)을 이용하여 발생 가능한 공정 이상 유형을 학습하고, 현재의 공정 상태를 기반으로 이상 상황의 발생 여부를 진단할 수 있는 지능형 이상 진단 시스템을 제시하였다. 또한 본 연구에서 제시한 지능형 이상 진단 시스템은 하이브리드 데이터마이닝에 의해 학습된 규칙을 기반으로 현재의 공정 상태를 진단하고, 만약 현재 공정 상태가 이상 상황으로 진단되었을 경우 진단된 이상 상황 유형에 따른 효율적인 보전 작업을 찾기 위해 의사결정 네트워크를 이용하여 모든 가능한 대안들을 도식적으로 기술하고, 의사결정 네트워크로부터 유추된 대안들 중에서 AHP를 이용하여 최적의 대안을 선택할 수 있는 방법을 제시하였다.

참고문헌

[1] Berry, M. and Linoff, G., *Data Mining Techniques: For Marketing, sales, and Customer Support*, John Wiley&Sons, 1997.
 [2] Boutilier, C., Dearden, R., and Goldszmidt, M., "Stochastic dynamic programming with factored representation" *Artificial Intelligence*, vol.121, no.1-2, pp.49-107, 2000.

표 4. ++(1) 이상 상황 해결을 위한 보전 작업 대안들.

Maintenance action 1 [++(1)→0(1)]: Adjust(A1)
Maintenance action 2 [++(1)→0(2)]: Adjust(A1) & Adjust(A3)
Maintenance action 3 [++(1)→0(3)]: Adjust(A4) & Adjust(A3) & Adjust(A5)
Maintenance action 4 [++(1)→0(4)]: Adjust(A4) & Adjust(A3) & Adjust(A5) & Adjust(A1) & Adjust(A9)

2005 한국경영과학회/대한산업공학회 춘계공동학술대회
2005년 5월 13일 ~ 14일, 충북대학교

- [3] Carvalho, D. R. and Freitas, A., “A hybrid decision tree/genetic algorithm method for data mining”, *Information Science*, vol.163, no.1/3, pp.13-35, 2004.
- [4] Fu, L., *Neural Networks in Computer Intelligence*, McGraw-Hill, 1994.
- [5] Han, J. and Kamber, M., *Data Mining: Concepts and Techniques*, Morgan Kaufmann, 2001.
- [6] Hand, D. J., *Construction and Assessment of Classification Rules*, John Wiley&Sons, 1997.
- [7] Lee, J., “Measurement of machine performance degradation using network”, *Computers in Industry*, vol.30, no.3, pp.193-209, 1996.
- [8] Michalewicz, Z., *Genetic algorithms + Data Structures=Evolution Programs*, 3rd Ed., Springer, Berlin, 1996.
- [9] Mitchell, T. M., *Machine Learning*, McGraw-Hill, New York, 1997.
- [10] Onnia, V., Tico, M., and Saarinen, J., “Feature selection method using neural network”, *Proceedings of International Conference on Image Processing*, vol. 1, pp.513-516, 2001.
- [11] Quinlan, J. R., *C4.5: Programs for machine learning*, Morgan Kaufmann, 1993.
- [12] Saaty, T. L., *The Analytic Hierarchy Process*, McGraw-Hill, New York, 1980.