

계층구조 접근에 의한 복합시스템 고장진단 기법

배 응 환*, 이 석 희**

Fault Diagnosis Method of Complex System by Hierarchical Structure Approach

Yong-Hwan Bae*, Seok-Hee Lee**

ABSTRACT

This paper describes fault diagnosis method in complex system with hierarchical structure similar to human body structure. Complex system is divided into unit, item and component. For diagnosing this hierarchical complex system, it is necessary to implement special neural network. Fault diagnosis system can forecast faults in a system and decide from current machine state signal information. Comparing with other diagnosis system for single fault, the developed system deals with multiple fault diagnosis comprising Hierarchical Neural Network(HNN). HNN consists of four level neural network, first level for item fault symptom classification, second level for item fault diagnosis, third level for component symptom classification, fourth level for component fault diagnosis. UNIX IPC(Inter Process Communication) is used for implementing HNN with multitasking and message transfer between processes in SUN workstation with X-Windows(Motif). We tested HNN at four units, seven items per unit, seven components per item in a complex system. Each one neural network operate as a separate process in HNN. The message queue take charge of information exchange and cooperation between each neural network.

Key Words: Multiple Fault Modeling(다중고장모델링), Multiple Fault Diagnosis(다중고장진단), IPC(Inter Process Communication), Hierarchical Neural Network(계층신경망), Complex System(복합시스템), Hierarchical Analysis(계층분석), Hierarchical Process Model(HPM:계층 프로세스모델), Hierarchical Component Model(HCM:계층 요소모델), Hierarchical Fault Model(HFM:계층고장모델)

1. 서론

산업사회의 발달로 무인화 공장들의 요구에 따라 시스

템은 더욱 복잡해지고 설계에 많은 인력과 지식이 필요하고, 그 기능들도 매우 다양해 질 것이다. 따라서 이와 같은 첨단장치의 많은 부분들은 주로 인체구조와 그 기능을

* 안동대학교 기계공학교육과 (정회원)

** 부산대학교 생산기계공학과, 기계기술연구소 연구원(중신회원)

모방하므로 매우 복잡해 질 것으로 고려된다. 이러한 장치들은 주로 기계장치계, 전기, 전자장치계, 유압장치계, 연소장치, 화학반응장치 등의 각기 다른 물리현상과 메카니즘에 의하여 지배되며, 이러한 장치들은 상호간에 긴밀한 연관성을 가지고 움직이므로 이들 사이에서 일어나는 고장진단은 매우 복잡하다. 따라서 자동화된 복합시스템의 자동고장진단시스템에 대한 연구가 필수적이다.⁽¹⁾

지금의 고장진단 시스템은 주로 센서신호를 입력하여 그 특징패턴을 추출하여 고장증상 패턴매칭에 의하여 고장진단을 하는 것을 기본으로 한다. 그러나 복합자동화(가공, 화학프로세스)시스템에서 일어나는 고장들은 아주 다양하여, 완전한 진단은 매우 불가능하고 경제적인 측면에서도 고려되어야 하는 문제이다.

선진국의 고장진단 연구는 주로 인공지능의 한 연구분야로서 진단의 일반적 방법론에 대한 수많은 연구가 진행되어 왔다. 아울러 여러 가지 응용분야에 전문가시스템에 관한 연구가 본격적으로 진행되고 있는데 대표적으로는 기계고장진단⁽²⁾, 전자시스템고장진단⁽³⁾, 항공기 고장진단⁽⁴⁾, 플랜트 고장진단⁽⁵⁾, 용광로 상태진단⁽⁶⁾ 등에 관한 연구가 매우 활발하다. 이러한 시스템의 특징은 고장전파속도가 매우 빠르고, 미세 요소의 고장이 전체시스템 고장으로 전파되는 특성을 지니고 있어서 초기 빠른 고장진단이 필수적이다. 자동화 복합시스템은 주로 모듈별로 분할되어 있고, 고장물리 특성상 고장원인은 한가지 물리현상이라기 보다는 다수의 복합된 물리현상의 요인에 의하여 발생되므로 한꺼번에 여러 개의 고장이 동시에 일어날 수 있는데, 각각 물리현상에 대한 지식을 종합하여 진단해야 하는 복합 다중문제 해결기법에 관한 연구가 필요하며, 이를 위해서 이들 소규모 진단 모델 사이의 정보교환이 절대 필수적이다.

따라서 본 연구에서는 이런 형태의 고장의 종류와 위치를 정확히 구별해 내기 위하여 고장추론이 병렬화되어야 하고, 각 고장추론은 최종적으로 각기 다른 기능을 수행하는 복합적인 요소로 설계된 장치에서 일어난 고장진단을 목적으로 하는 인간의 추론구조 바탕 위에 경험적인 지식(shallow knowledge)으로는 해결할 수 없는 다양한 수식, 물리적 법칙이나 기타 수학적 심층지식 모델(deep knowledge)을 통합한 복합시스템 실시간 지적 다중고장진단 시스템을 지원하기 위한 하나의 계층화 다중병렬처리 신경망을 개발하는 것이다. 이것은 하나의 서브시스템이 고장을 진단하는 도중 다른 하나의 서브시스템에서 고장이 일어나면 또한 진단을 시작할 수 있도록

멀티프로세싱기법을 이용한다. 다중진단은 프로세스 생성 소멸을 이용한 에이전트(agent)개념을 이용하여 분산처리하고, 이러한 각 에이전트의 출력정보를 각 에이전트들에게 송수신하기 위하여 블랙보드(blackboard)개념을 이용하고, 이와 같은 연구를 위해서 다중 운영체제(UNIX)와 네트워크 통신이 필수적이며 진단 모델들간의 통신은 IPC(Inter-Process Communication)설비에 의하여 이루어진다.

2. 이론적 배경

감시하고자 하는 장치의 고장진단은 알려진 참조값에 대한 현재의 작동파라메타의 비교를 통하여 이루어진다. 장치가 제대로 작동하지 못할때 전문가가 그의 경험을 이용하여 이것을 진단하려고 한다. 그리고 외부 입력신호에 대한 시스템의 측정된 출력데이터를 참조하여 정확한 진단을 수행한다. 진단의 정확도는 전문가의 지식, 특히 경험이나 기계응답에 대한 그의 감각에 의존한다. 인간 전문가에 의한 고장진단의 핵심은 기계의 각 부분이 어떻게 작동되고 또한 그들이 어떠한 인과관계에 따라 작동되는지에 관한 완벽한 정보적 매칭을 통하여 수행된다. 자동 고장 진단시스템이 다양한 기계에서 기능이상을 진단하기 위하여 점차적으로 채택되어지고 있다. 진단시스템은 지식베이스나 인간전문가가 해결하기를 바라는 각기 다른 영역의 문제를 해결하는 추론엔진을 사용하는 컴퓨터 프로그램이다. 지식베이스의 모듈형태와 그들의 사고방식을 설명하는 능력은 진단영역에서 진단셸(diagnosis shell)을 선택할 수 있는 기준을 부여한다. 1세대 진단시스템의 가장 중요한 적용분야는 규칙기반 파라다임을 개발로 특징지워진다. 실험적 환경에서 이러한 접근의 상당한 성공에도 불구하고 심각한 한계는 80년대 초기에 명확해 졌다. 많은 연구자들은 규칙기반접근의 장점과 한계를 아주 극단적으로 분석했다.

규칙기반 고장진단 시스템을 개발하는데 있어서 중요한 단점은 지식획득이다. 첫째, 통상적으로 전문가와 수동인터뷰에는 애매한 지식들이 많이 포함되어 있다는 것이다. 둘째로 대부분의 지식공학자는 전문가와 효과적으로 인터뷰하도록 훈련되지 않아서 불완전한 지식의 가공에 문제점이 많다. 세번째 몇몇 상황에 대하여 전문가가 존재하지 않아서 그러한 고장상황이 발생하였을 때 진단시스템은 고장을 간과하는 경우가 많아서 대형사고를 초래할 수 있다. 이러한 문제를 해결하기 위하여, 최근에는 애매

한 고장지식의 정량화와 획득된 고장패턴과 유사한 고장을 탐색하는 방법으로 인공지능 기법중의 하나인 fuzzy rule과 neural network이 많이 연구되고 있다.^(7,8,9) 다음은 본 연구의 핵심기법인 복합시스템의 계층화 접근방법에 대하여 살펴보자.

2.1 기계시스템의 계층화 표현

다음은 기계의 각 시스템의 구조적, 기능적, 동적특성에 대하여 살펴보자. 기계시스템도 인체와 마찬가지로 각 기관에 해당하는 부속시스템으로 이루어져 있고, 각 부속시스템은 작은 모듈별 유니트(unit), 유니트는 각 아이템(item)으로, 아이템은 다시 작은 요소(component)로 구성된다. (Fig. 1) 여기서 대부분의 경우 고장은 주로 최소단위인 요소에서 시작하여 그 상위레벨로 전파되고,

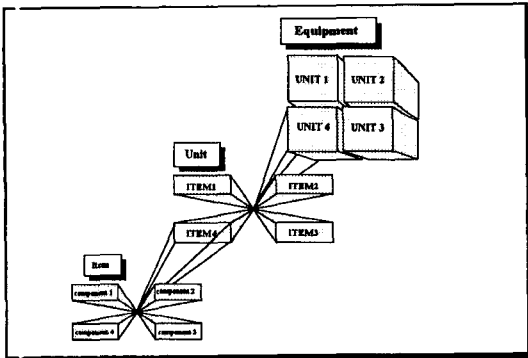


Fig. 1 Hierarchical decomposition of complex system

관찰되는 증상은 한층 높은 상위레벨에서 나타난다. 따라서 컴퓨터시스템이 이와같은 계층적 형태의 시스템을 진단하기 위해서는 구조적인 계층특성과 기능적, 동적 상호연관성을 표현해야 하는데 이와같은 시스템 구조의 계층적 표현방법이 Fig. 2에 나타나 있다.⁽¹⁰⁾ 여기서 수평접선은 부품들 사이의 연결을 표시하고, 연결은 같은 레벨에서만 가능하고 각기 다른 레벨로의 연결은 허락하지 않는다. LISP나 객체지향 프로그래밍언어(OOPL)는 이와같은 계층적모델을 구축하기 위하여 쓰여질 수 있다. Prolog 구현은 계층의 상위레벨에서 하위레벨로 작동하며 다음과 같다.

이와 같은 계층구조를 이용하면, 고장지 의심가는 요소의 부속구조만을 탐색할 필요가 있고, 이중에서 고장을 탐지해 내거나 의심의 소지가 없는 요소들을 제거할 수 있으므로 고장진단을 위하여 아주 효율적이다.

```

Top level
system (Name, A, B) consists_of
|
| subsystem (N1, A, B),
| subsystem (N2, B, C),
| subsystem (N3, C, D),
| subsystem (N4, D, E)
|
|
Second level
subsystem (Name, A, B) consists_of
|
| controller (Cname, A, B, V),
| pump (Pname, V, Fn1, Qin),
| pipework (PWname, Qin, Qout, Qnet),
| tank (Tname, Qnet, L),
| valve (Vname, L, Kv, Qout),
| sensor (Sname, L, Fn2, B)
|
|
Third level
controller (Cname, A, B, V) consists_of
|
| adder (L, A, B, E),
| gain (L, E, Kp, V)
pump (Pname, V, Fn1, Qin) consists_of
|
| gain (L, V, Fn1, Qin)
pipework (PWname, Qin, Qout, Qnet) consists_of
|
| adder (L, Qin, Qout, Qnet),
tank (Tname, Qnet, L) consists_of
|
| integrator (L, Qnet, T, Vol),
| gain (L, Vol, InvArea, L)
valve (Vname, L, Kv, Qout) consists_of
|
| gain (L, L, Kv, Qout)
sensor (Sname, L, Fn2, B) consists_of
|
| gain (L, L, Fn2, B)
|
|
Fourth level
The description continues at the fourth, fifth and other levels in the
same way until the hierarchy is complete.
For example the gain description is :
gain (Name, A, B, C) consists_of
|
| mult (Name, A, B, C)
|
|
    
```

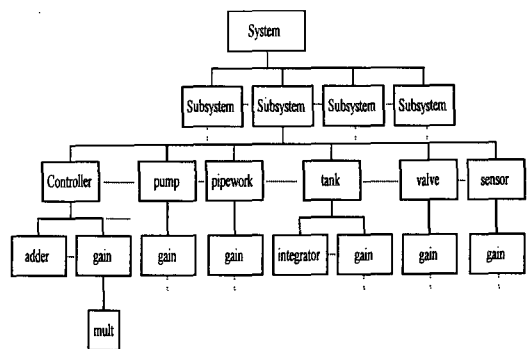


Fig. 2 Hierarchical structure of complex system

2.2 복합시스템의 계층적 지식표현

2.2.1 물리계 모델링 원리 (principle of physical system modeling)

어떤 의미에서 고장은 어떤 기능적 구축조건에 깨뜨림과 예상된 시스템 동정에 벗어나감으로 정의할 수 있다. 만

약 시스템속에 하나의 요소가 고장이 났다고 말한다면 이것은 해당된 어떤 기능을 만족하지 않는다는 것을 의미한다. 시스템고장과 물리적요소의 고장은 그들 기능상실에만 정의될 수 있다. 예를들면 탱크에 조그마한 구멍이 나 있는 탱크를 고려할때 만일 이것이 액체를 저장하는 기능을 하면 이것은 고장이 난 것으로 고려된다. 그러나 만약 이것이 다른 요소를 지지하기 위한 일시적인 장치라면 고장이 아니다. 고장탐지 시스템은 실제 시스템 동정과 예상된 시스템 동정을 비교함으로써 고장을 탐지한다. 그러므로 고장모델을 개발함에 있어서 시스템의 기능을 모델링하는 것이 중요하다. 시스템에서 고장의 소스는 그들의 기능구속조건을 깨뜨린 일련의 부속요소 집합이나 혹은 하나의 부속요소이다. 이러한 이유로 인하여 시스템구조도 역시 모델링되어야 한다.

우리가 시스템의 기능을 모델링하기를 원한다고 말할때 우리가 관심을 가지는 성질들(properties)은 첫째는 다양한 서브시스템의 실제적인 기능이 될 것이고, 또 다른 중요한 성질은 각기 다른 서브시스템 사이의 상호작용들일 것이다. 그리고 각기 다른 기능 엔티티들간의 상호관계를 모델링함으로써 우리는 고장의 전파를 발견할 수 있다. 각기 다른 작동상태를 모델링하는 것도 필요하다. 시스템이 각기 다른 작동상태를 반영할 수 있도록 하는 기능모델이 필요하다. 진단을 위하여 고장모델을 따라가야 하고, 시스템의 어느 곳이 목적에 맞게 작동되지 않는지를 모든 길을 간파할 필요가 있다. 즉 간략히 말하면 우리는 기능적 고장들을 충분히 정의할 필요가 있다. 또한 특정의 기능적 고장의 예를 탐지하기 위한 고장탐지 방법을 규정할 필요가 있다. 마지막으로 고장전파향으로 모든 기능적 고장들의 전파영향을 규정해야 한다. 시스템 구조가 모델링 되었을때 우리는 요소에서 고장들만을 모델링할 필요가 있다. 복합시스템들은 원래 매우 복잡하므로 시스템을 서브시스템으로 분해하지 않고 단독으로 모델링한다는 것은 불가능하다. 분해는 계층의 각 레벨과 계층의 깊이에서 표현된 상제사양을 다 취급토록 한다. 시간 임계상황에서 계층의 한결 높은 레벨에서 진단탐색을 멈추는 것을 가능하게 하고, 몇몇 정확성이 낮은 것이지만 진단결과를 만들어 낸다. 예를들면 가스버너 시스템의 고장상황에서 진단시스템은 고장원인요소로써 부속시스템 모듈중 하나인 가스공급 어셈블리를 초기 진단 대상으로 규정할 것이다. 만약 진단할 시간이 많다면 진단시스템은 가스공급장치에서 한층 깊이 추론하여, 고장의 요소로 특수한 밸브가 고장이라고 판단한다. 만약에 시간이 없다면

진단시스템은 고장의 원인으로써 가스공급장치를 규정한 다. 계층의 깊이제어는 진단시스템이 진단해야하는 정확한 정도(계층적 노드 상위, 하위관계)에서 결정된다. 시간 임계상황에 대하여 쉽게 대처될 수 있는 조잡한 정도 낮은 진단으로 분해하는 것이 가능하다. 이러한 이유 때문에 구조와 기능적인 모델은 자동적으로 계층구조를 이루어야 한다. Fig.3은 고장진단 셸에서 계층화시스템이 가지는 시스템의 물리적인 고장특성 지식 구현 방법을 나타낸 것이다. 하나의 시스템은 3개의 부속요소로 이루어져 있고, 이 세 개의 부속시스템은 서로 상호간에 제어한다. 하나의 부속시스템의 고장진단 표현을 세분화해 보면, 현재의 시스템의 상태를 그래픽으로 처리하는 모듈이 있고, 그 기계의 초기 시동시 각 부분의 설정정보를 제공하고, 이상시 테스트 정보, 고장시 진단기능과 진단후 처리방법에 대한 데이터를 제공해야 한다.

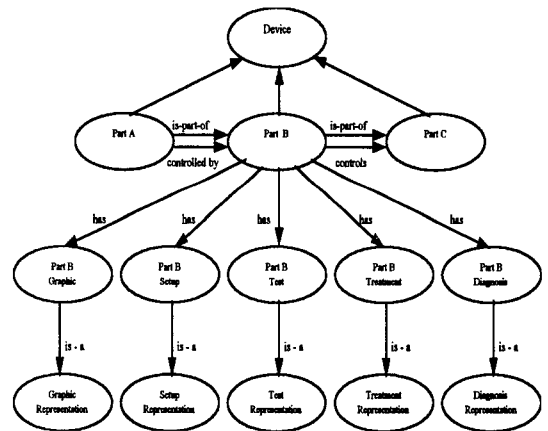


Fig. 3 Representation of device knowledge in the diagnosis shell

2.2.2 계층화 기법에 의한 다중고장 모델링 (multiple fault modeling)

앞절에서 언급되었듯이 강력한 고장모델을 얻기 위하여 시스템의 구조와 기능이 모델링되어야 한다. 어떤 시스템의 구조와 기능적인 측면은 많은 분야의 각기 다른 정보를 포함한다. 그와같은 모델링에 포함된 복잡도 문제들은 다중측면 모델링이라 불리우는^{11,12)}기법을 사용함으로써 접근될 수 있다. 이와같은 기법을 이용하여 개발된 각기 다른 모델들은 시스템의 구조와 기능을 표현한다. 그들은 역시 분해계층에서 추출 작성되었다. 구조와 기능은 각기 다른 측면에서 서로 독립적으로 표현될 수 없다. 광범위

한 모델링의 궁극적인 목적은 이와 같은 상호작용의 표현이다. 따라서 모델링의 방법론은 이와 같은 측면들간의 관계를 정의해야 한다. 다중측면 모델링에서 실제적인 문제는 각각의 상황표현(독립적인 모듈에서 고장발생)이 아니라 그들사이의 상호작용의 표현이다. 즉 다중고장은 여러개의 고장이 독립적으로 일어나는 것이 아니라 각각의 요소들간의 상호작용에 의하여 일어나는 것이다. 즉 이것은 어떤 병의 시발에서부터 합병증이 일어나는 경우이다. 그러나 실제 시스템에서는 여러개의 독립된 부속시스템에서 동시에 고장이 일어날 수 있다. 이러한 목적을 위하여 모델링 기법은 구조적 독립의 선언(SI:structurally independent)과 구조적 의존(SD:structurally dependent)모델링 선언을 구분하여 표현해야 한다. 두가지 모델링 상황은 그들의 모델링 구조들이 각자 상호간에 직접적인 구속조건을 부여하지 않으면 SI로 규정한다. 만약 모델링 상황이 SD로 정의된다면 소위 지배상황이라 불리우는 구조는 의존상황들을 결정한다. 이것은 어떤 시스템에서 그 시스템의 동정을 지배하는 지배요소가 다른 부속요소의 동정을 지배한다. 즉 펌프에서는 물이 나오지 않는 것이 지배적인 요인이고 그 요인은 다른 모든 부속적인 상황에 의하여 발생할 수 있다. 다른 말로하면 지배상황은 다른 의존적인 상황의 구축을 위한 일련의 집합을 제공한다. 시스템의 기능은 Hierarchical Process Model(HPM)로 알려진 지배상황모델을 발생시킨다.^[13] 시스템의 구조는 Hierarchical Component Model(HCM)로 알려진 지배상황모델을 발생시킨다. 이 두가지 지배상황모델은 그들 사이에 관계가 비록 존재하지만 각각 독립적이다. HPM에서 프로세스들은 그들의 기능을 수행하기 위하여 HCM에서 몇몇 요소들을 사용한다. 비슷하게 HCM에서 하나의 요소는 많은 기능을 지원할 것이다. 혹은 이것은 어떤 주어진 시간에 전적으로 어떤 기능도 지원하지 않는다. 이 두가지 상황사이의 관계와 상호작용이 HPM-HCM 상호작용들로 불리우는 하나의 모델형태로 표시된다. Hierarchical Fault Model(HFM)으로 알려진 또다른 부가적인 형태의 모델이 정의될수 있는데 HFM은 하나의 추상적인 모델로써 HPM, HCM, HPM-HCM 상호작용속에서 표현되는 고장진단정보를 통합화를 지원한다.

2.2.3 계층프로세스모델 (Hierarchical Process Model)

HPM은 시스템에서 에너지와 물질전달을 표현하는 대표적인 표현이다. 이것은 요구된 작동조건을 충족시키기

위하여 사용될 수 있는 개념들을 정의하기 때문에 HPM은 시스템 작동의 기능적인 표현으로 고려될 수 있다. HPM에서 하나의 프로세스에서 다음과 같은 몇몇 중요한 개념이 존재한다.

1. **Subprocesses** : 프로세스의 기능은 많은 부속기능들로 분해될 수 있다. 각 부속기능들은 초기 프로세스의 부속프로세스로 될 수 있다. 이와같은 방식으로 시스템의 계층기능분해가 성취될 수 있다. Fig. 4의 가스보일러 공정의 계층적인 기능분해가 Fig. 5에 보여진다.

2. **Process Variables** : 각 프로세스는 에너지이든, 물질이든 하나의 입력, 출력 변수들을 갖는다. 하나의 프로세스는 HCM으로 부터 몇몇 요소들을 사용하여 그들의 입력변수를 그들의 전달함수에 기초하여 출력변수로 변환한다. 각기 다른 프로세스들은 그 자신의 출력변수를 다른것의 입력변수에 연결함으로써 다른 것과 상호

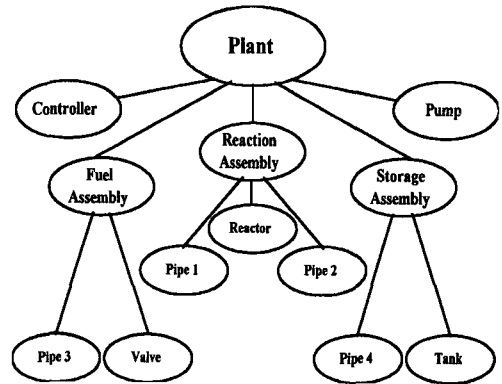


Fig. 4 Hierarchical structural decomposition

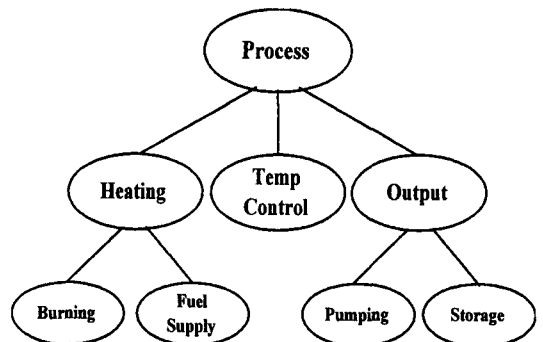


Fig. 5 Hierarchical functional model

작용한다. 이와같은 방식으로 하나의 시스템 내부에 동적 상호작용을 모델링할 수 있다. 구조적인 연결정보를 따라서 이와같은 동적 상호작용은 인과 고장모델을 유도하기 위하여 사용되어질 수 있다.

3. Process States : 하나의 프로세스는 그와 연관된 많은 상태들중 하나로 존재한다. 이것은 어떤 사건이 일어날때 다른 상태로 천이한다. 각기 다른 상태에서 프로세스의 기본적인 차이가 프로세스의 특성을 나타내는 함수가 될 수 있고, 그들의 연관 요소, 그들의 출력의 특성이 될 수 있다.

4. Process Failure Modes : 운전을 통하여 프로세스는 고장의 존재로 인하여 어떤 기능적 구속조건을 깨뜨릴 수 있다. 기능적 구속조건에 이와같은 반칙은 고장모드로써 참조될 수 있다. 어떤 프로세스의 기능적 고장모드를 모델링하는 것이 가능하다. 예를들면 Fig. 4에서 펌핑프로세스는 유체를 펌핑하는 기능을 수행한다. 만약 이 프로세스에 몇몇 고장이 일어난다면 출력에서 유체의 압력은 정상상태보다 높거나 정상상태보다 낮다. 그러므로 펌핑프로세스는 high-pressure, low-pressure 두가지 고장모드를 포함한다. 고장모드의 개념은 기능고장 사양요구조건을 만족하기 위한 기본을 나타낸다.

5. Process Alarms : 하나의 프로세스를 위하여 어떤 수의 고장모드를 규정할 수 있다. 이와같은 방식으로 우리는 기능적 구속조건들의 모든 가능한 반칙들을 모델링할 수 있다. 또한 기능적 반칙들을 탐지하는 방법을 정량적으로 규정할 필요성이 있다. 펌핑프로세스의 경우 우리는 출력압력이 100psi보다 높다면 고압고장모드가 존재한다고 규정할 수 있다. 그리고 만약에 출력압력이 50psi보다 낮다면 저압고장모드가 존재한다. 만약 압력센서가 펌핑프로세스의 출력에 존재하면 고압알람과 저압알람이 발생될 수 있다. 고압알람은 고압고장모드와 관련될 수 있고, 고압모드를 탐지하기 위한 수단으로 간주된다. 알람의 두가지 타입에서 두가지 경우가 가능하다.

- AlarmOn : 기능적인 구속조건에 반칙이 탐지되었다는 것을 표시.
- AlarmOff : 가장 일찍 탐지된 기능적 구속조건 반칙이 없어졌음을 표시

6. Fault Propagation Graph : 하나의 고장모드에 일어나는 후속고장모드들은 프로세스들 사이의 동적 상호작용 때문에 다른 프로세스에서 고장모드로 전파

될 수 있다. 이것은 고장이 일어난 후 실제 환경에서 알람쇄도에 의하여 알 수 있다. 이와같은 알람의 대부분은 시스템의 특정부분 고장에 의하여 다른 부분으로 전파되는 것에 기인한 것이다. 그와같은 모든 고장전파셀연결이 고장전파그래프이다. 고장전파그래프가 임의의 부속프로세스의 고장모드와 전파특성이 하나의 프로세스에 대하여 유도된다. 거기에는 and, or 두가지 형태의 고장전파 링크가 있다. 각 고장전파링크는 4가지 파라메타들에 의하여 비중이 달라진다. 이것들은 고장전파 가능성, 최대고장전파시간, 최소고장전파시간, 링크가 유효한 부고장상태 리스트이다. 고장전파링크의 고장전파가능성은 소스고장모드가 일어난 후 종점고장모드의 발생가능성을 표시한다. 고장전파최소시간과 고장전파최대시간은 소스고장모드가 일어난 후 종점고장모드가 일어나는 시간간격을 정의한다.

2.2.4 계층요소모델 (Hierarchical Component Model)

HCM은 시스템의 공간적 분해를 표현하는 탁월한 방법론이다. 물리적 요소계층은 결합체(assembly), 부속결합체(subassembly), 기초요소로 규정되고, 시스템의 작동에서 그들의 실제 기능으로 부터 독립적으로 규정된다.⁽¹⁴⁾ 공정프로세스의 계층적 구조분해는 요소와 부속요소로 분해되고 Fig. 5에 보여진다. HCM은 요소들 사이의 관계가 아직 모델링되지 않은 의미에서 단순한 부분계층이다. HCM에 있어서 요소는 많은 방향에서 활동된 기능을 수행하는데 실패할 수 있다. 예를들면 유체를 전달하는데 책임이 있는 파이프는 누출을 일으킬 수 있고, 막힐 수 있다. 두 경우에 파이프는 유체를 전달하는 주요기능을 수행하는데 실패한다. 요소속에서 그와같은 고장들은 고장난 상태로써 참조된다. HCM에서 각 요소에 대하여 원하는 모든 고장난 상태의 리스트와 현존하는 모든 부속요소리스트 정보가 요구된다.

2.2.5 계층프로세스모델-계층부속모델 상호작용

HPM에서 하나의 프로세스는 HCM에서 몇몇 요소들을 사용함으로써 그들의 기능을 수행하는 하나의 엔티티이며, 프로세스 고장모드 사이의 고장전파가 알려져 있다. 비록 그와같은 효과들이 표현되지만 그러나 프로세스 고장모드에 대한 고장난 요소의 효과에 대하여 아무것도 알려져 있지 않다. 하나의 요소가 고장을 일으켰을때 그에 따른 고장상태가 나타나고, 이것을 사용하고 있는 프로세스들은 고장을 전개할 것이다. 만약 이상적인 고장진단

시스템에서는 이와같은 상호작용이나 고장난 상태들의 요소간의 고장전파와 프로세스 고장모드는 모델링되어야 한다. HCM에서 각 요소의 고장경우에 대하여 HPM에서 프로세스에 대한 일련의 고장전파에 대한 지식이 요구된다.

2.2.6 계층고장모델 (Hierarchical Fault Model)

진단추론 알고리즘에 적절한 정보는 HCM과 HPM 그리고 HCM-HPM사이의 상호작용에서 발견되어진다. 어떤 사실을 단순화하기 위하여 모든 적절한 고장진단정보를 포함하는 요약된 통합모델이 정의된다. 이 새로운 모델이 계층고장모델(HFM)이다.⁽¹⁵⁾ 이 모델속에서 모든 정보는 자체 규정되어지거나 다른 모델에서 획득된다. 이와같은 모델을 만드는 이유는 역시 진단추론기법의 이해를 높이는 것과 고장모델링기법표현을 단순화하는 것이다. 고장모델로써 불리워지는 내용들은 HPM에서 각 프로세스를 위하여 정의되어진다. HPM에서 어떤 프로세스의 고장모델은 프로세스의 고장전파 viewpoint 정보, 고장정보셀 즉, 그들의 모든 부속시스템의 모든 고장모드의 알람 연결들, 요소들의 고장난 상태들과 모든 고장모드들 사이의 고장전파 링크스와 모든 부속셀의 고장모드들에 대한 정보를 포함한다. HPM에서 모든 프로세스들의 고장모델들의 셀은 계층고장모델(HFM)이다.

3. 다중고장진단을 위한 계층형 신경망 구현

본 연구를 시작함에 있어서 시스템의 적용이 가장 문제이었다. 원래 복합시스템이란 앞에서 정의 했듯이 다양한 물리현상에 의하여 작동되는 시스템으로 우주선이나 기타 복잡한 압연시스템, 대형정밀 복합시스템들을 들 수 있으나 이와같은 복합시스템을 실제 실험을 위하여 제작하는 것은 비용과 규모에 있어서 수많은 인력이 소요되므로 이는 거의 불가능하다. 그렇다고 해서 이와같은 시스템이 산업현장에 쓰이지 않느냐 하면 그렇지 않다. 예를들면 우리나라의 경우 압연시스템과 파이프밴딩 머신과 같은 기계는 본 연구대상 복합시스템과 잘 일치하는 것들이다. 그러나 이와같은 복합시스템이 존재한다손 치더라도 실제적으로 고장을 다중으로 일으킨 다음에 센서를 통하여 신호를 입력한다는 것은 거의 불가능하다. 본 연구의 핵심은 이러한 계층형 신경망의 유기적인 연결동작에 바탕을 두었다. 본 연구의 실제 적용은 실제 시스템을 한꺼번에 맞추어서 고장을 낸다는 것은 불가능하며, 본 연구에

서는 이와같은 제약조건 때문에 가상의 시스템을 정의하고, 각 신경망에서 사용되는 고장학습데이터는 여러 가지 단독적으로 운행되는 시스템(모터, 베어링, 화학프로세스)등에서 채취된 고장신호를 입수하여 학습하였으며, 본 연구의 대상 복합시스템은 이와같은 단독적으로 운전되는 시스템들이 하나로 조합된 가상의 복합시스템으로 정의한다. 먼저 이와같은 복합시스템은 4개의 단독시스템으로 이루어져 있고, 각 시스템은 또 7개의 아이템 요소로 구성되어 있으며, 각각의 아이템 요소는 7개의 기계부품요소로 이루어져 있다고 가정한다. 가상의 여러개의 가상의 고장신호데이터를 입력받아 온라인 모니터링과 다른 기타 모듈들에 대한 특정의 입력조건에 맞도록 신호전처리를 실시하는 전처리 모듈들로 보내어진다. 이때 프로세스는 공유메모리를 이용하여 두개의 모듈에 동시에 센서정보를 보낸다. 보내진 정보는 이것을 참조하여 서브 D/B에 저장되어 있는 각종의 정상상태값과 비교하고, 정해진 기준치를 넘으면 고장부위(fault location)가 정해지는데 이 고장부위는 주로 서브시스템으로 분리된다. 서브시스템에 이상이 생기면 정밀진단에 들어간다. 메인진단모듈은 부속시스템의 정밀진단을 위하여 특정의 부속시스템 진단시물레이터는 정밀진단에서는 먼저 유니트를 대표하는 센서의 입력을 근거로 현재의 고장이 어느 위치로부터 일어났는지를 계층형신경망(Hierarchical Neural Network)을 이용하여 진단한다. 하나의 계층형신경망은 주로 2단계로 나누어져 추론하는데, 초기단계는 단일고장을 분류해내는 역할을 하고 두번째 단계는 의심이 가는 고장부위를 테스트하여 확실한 고장진단을 수행한다. 하나의 상위레벨 계층형신경망은 부속시스템의 아이템(item)들의 이상여부를 가려내기 위하여 시물레이션되고, 또다른 하위레벨 계층형신경망은 아이템내부의 요소(component)의 고장을 구별하기 위하여 시물레이션 되어진다.

3.1 실시간 다중프로세스 생성 및 대체

계층적 고장진단을 수행하기 위한 계층형 신경망을 효율적으로 운영하기 위해서는 각자 독립된 프로세스의 다중처리가 필수적이다. 이를 위하여 본 연구에서는 UNIX 운영체제에서 사용자가 새로운 프로세스를 생성할 수 있는 유일한 방법은 fork()시스템 호출을 부르는 것이다. fork()를 부른 프로세스를 부모프로세스라하고, 새로이 생성된 프로세스를 자식프로세스라고 부른다. fork()시스템 호출구문은 다음과 같다.⁽¹⁶⁾

```
pid = fork();
```

fork()시스템 호출로 부터 복귀할 때, 두개의 프로세스는 복귀값인 pid를 제외하고는 사용자 수준의 문맥이 똑같은 상태이다. 한편 fork()된 자식프로세스는 자신의 영역을 다른 프로세스로 대체시키기 위해서 execl()시스템 호출을 수행한다.

3.2 실시간 다중프로세스간 메시지전달 및 동기화 설비

본 연구에서는 계층형신경망에서 각 계층간의 신경망이 입력데이터와 결과를 특정의 처리기(OR 프로세스, Result agent)에 전달하기 위하여 메시지큐(message queue)를 사용하는데 메시지큐는 여러개의 프로세스들간의 통신을 위하여 제공되는 것이다. 메시지는 세계의 시스템 호출을 통하여 이루어진다.⁽¹⁷⁾ 메시지 id를 얻기 위한 msgget()과 다른 프로세스에서 메시지를 받기 위한 msgrcv(), 다른 프로세스에 메시지를 주기 위한 msgsnd()이다. 본 연구에서 이용한 메시지 설비는 UNIX 시스템의 IPC(Inter Process Communication) 설비중 메시지큐를 사용하였다. 메시지 큐의 설비는 하나의 상위 모듈은 하나의 메시지큐를 이용하여 다수의 하위 모듈과 통신할 수 있다. 다수 프로세스간 메시지전달은 다음과 같은 3가지 과정을 거쳐 이루어진다. (Fig.6)

1) 메시지큐 생성 : 특정의 메시지큐를 생성한다.

```
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/msg.h>
```

```
int msg_qid;
key_t key;
.
.
.
msg_qid=msgget(key, IPC_CREAT|0666);
```

2) 메시지 수신 : 특정의 메시지를 수신한다.

```
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/msg.h>
```

```
int msg_qid, size, flags, retval;
struct my_msg{
long mtype;
char mtext[SOMEVALUE];
}message;
.
.
.
retval=msgsnd(msg_qid, &message, size, flags);
```

3) 메시지 송신 : 특정의 메시지를 송신한다.

```
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/msg.h>

int msg_qid, size, flags, retval;
struct my_msg{
long mtype;
char mtext[SOMEVALUE];
}message;
long msg_type;
.
.
.
retval=msgrcv(msg_qid, &message, size, msg_type, flags);
```

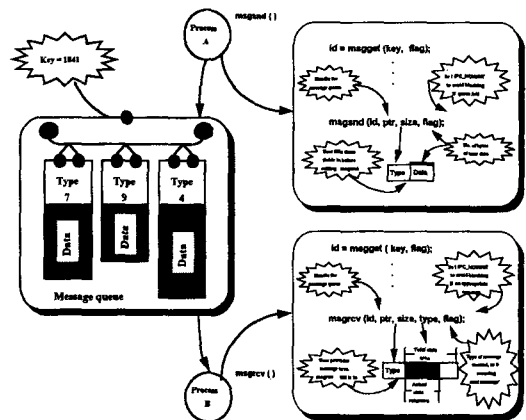


Fig. 6 Message queue operation example

3.3 실시간 계층형 다중신경망 구현에

일반적으로 시스템 내부에서 일어나는 현상은 다양한 수식모델과 다양한 정보패턴을 가지고 있으므로 이러한 정보를 빠르게 처리하기 위해서는 프로세스의 다중처리(multitasking)가 필요하고, 또한 각각의 다른 형태의 정보를 모듈별로 처리함으로써 진단시스템의 효율성 구조를 간략화 시킬 수 있지만 이들사이의 연관관계는 각 프로세스의 협조(cooperation)를 통하여 이루어져야만 한다. 하나의 요소의 고장진단중 다른 요소고장이 생기면 그것을 진단하기 위해서 해당 계층형 신경망이 트리거 되도록 되어 있고, 각각의 계층형 신경망은 실시간으로 자기의 많은 부속시스템에 대한 고장진단을 실시간으로 수행한다.

본 연구에서 고장진단 대상으로 삼은 시스템은 복합장치시스템으로 이 시스템을 구성하는 요소는 구체적으로 4개의 서브시스템, 7개의 아이템, 7개의 부속요소로 구성된다. 센서는 각 아이템의 고장특성을 대표하는 서브시스템 센서 4개와 부속요소의 고장을 대표하는 아이템센서가 7개로 가정한다. 각 센서는 그 하부요소에 대한 고장징후를 포함한다고 가정한다. 이 시스템은 특정기간 동안 쉬지 않고 작동되어야 하고 또한 예기치 않는 사고로 인하여 막대한 손실을 초래하여 생산성 저하를 가져올 뿐만 아니라 수리에 어려움이 많다. 따라서 이와같은 시스템은 각 부분별로 나누어 진단을 수행해야 하고 고장종류도 단일고장(single fault), 다중고장(multiple fault)

이 생길 수 있다. Fig. 7은 본 연구를 통하여 개발된 계층형 신경망의 구조를 나타낸 것이다. 하나의 단위 계층형 신경망은 first level, second level, third level, forth level로 이루어져 있는데, 이것은 하나의 유니트를 고장진단할 수 있고, 최대 4개의 유니트를 다중고장진단할 수 있도록 설계되어 있다. 또한 신경망의 학습은 back propagation 학습 알고리즘을 사용하였고, 각각의 신경망요소의 입력은 7개, 은닉층은 30개, 출력층은 7개이고, $\eta = 0.1$, $\alpha = 0.9$ 로 세팅하고, 반복은 최소에러가 0.008보다 적을때 까지 계속된다. 요구되는 반복수 평균은 6800이다. 학습은 먼저 오프라인으로 각 level의 신경망을 전부 학습시키면서 각 신경망의 입력샘플 데이터 개수, 및 입력노드수, 출력노드수, η , α , 은닉층 수, 각 은닉층마다의 노드갯수, 최대반복 학습횟수, 최소에러등을 파일에 저장해두고, 또한 각 신경망 노드의 최적 weight를 또 다른 파일형식으로 저장하고 있다가, 일단 first level의 부모프로세스에 의한 진단이 시작되면, 프로세스는 first 신경망 프로그램이 시작되면서 자기와 다른 프로그램의 동시시작을 위해서 fork()명령을 수행하면, first와 똑같은 이미지 프로세스가 하나가 생성되면서 하나는 first본래의 기능을 수행하고, 새롭게 생성된 다른 하나의 프로세스는 OR프로세스로 대체된다. 여기서 최상위 레벨인 first에서 최하위 레벨인 forth레벨의 각각의 요소신경망은 이미 각각의 특성에 맞는 고장데이터들을 offline학습한 후 최종상태의 weight값과 신경회로망 특성

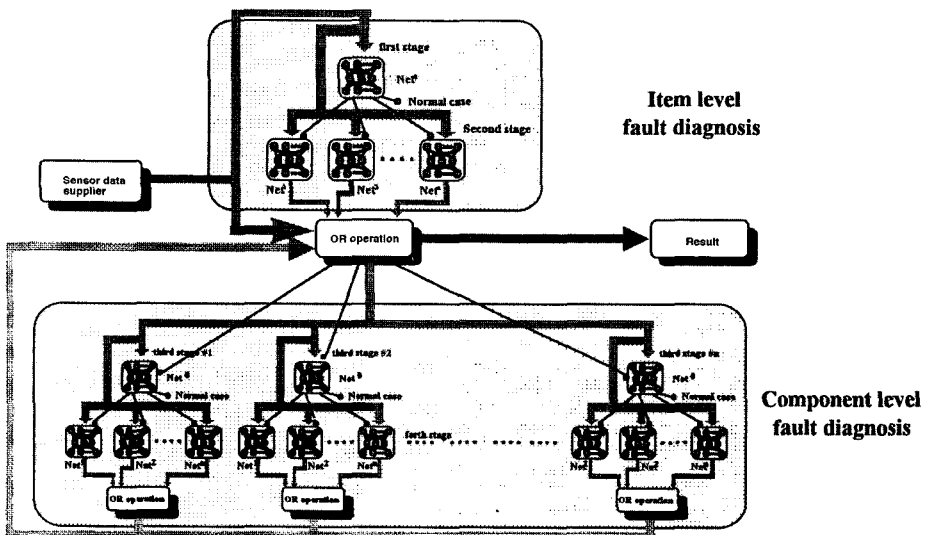


Fig. 7 Hierarchical Neural Network(HNN) operation

치를 파일 형식으로 저장하고 있다가 first와 각 second 신경망에 입력되어 고장분류 준비를 완료한 상태에서 first는 현재의 프로세스의 고장이 포함된 파라메타값을 입력받아 고장을 분류하는 동안에 OR프로세스는 하부 요소부분의 학습데이터와 신경망 특성값을 입력받아 third와 forth의 신경망을 고장진단 준비를 완료한다. first신경망이 고장진단 결과를 출력하면, OR프로세스로 메시지큐를 통하여 그 결과가 전달되고, OR프로세스는 고장이 존재하는 신경망이 포함된, 예를 들면 출력이 0010000이면 3번째 신경망을 시작시키고, 0011111이면, 3-7번째의 신경망을 시작시킨다. 여기서 second레벨의 신경망은 Net1의 경우 아이템 1에 관련된 이중고장만, Net7의 경우는 아이템 7에 관련된 이중고장을 학습 완료한 상태이다. 고장데이터가 입력되고, second 레벨 신경망이 수행되고, 그 결과가 다시 OR프로세스로 보내지고 OR프로세스는 second레벨의 각 신경망에서 결과를 받아 최종적으로 OR operation을 수행하여 결과를 출력한다. 본연구의 연구대상인 반응기는 구조상 second레벨까지의 신경망의 운용으로 진단가능하다. 만일 진단된 각 고장들이 그 하부의 또 다른 부속요소에 기인된다면, OR프로세스는 위의 이제까지 수행된 순서와 똑같이 수행하기 위하여 세부진단에 들어간다. 여기서 third레벨은 각 고장 아이템 1에서 아이템 7의 각각 고장을 일으키는 그 하부 요소가 7개 있다면 그 각각의 하부요소(component)를 진단하기 위하여 위와같은 수행을 반복한다. 그리고 OR프로세스는 이제 더 이상 진단이 필요없으므로 third 레벨 신경망을 죽이고, 또한 forth 레벨 신경망도 죽인다. 그리고 OR은 더 이상 필요하지 않는 first 신경망을 삭제시키고, 본 프로그램의 가장 부모프로세스는 다시 OR프로세스를 마지막으로 종료시킨다.

출력층속에 first level의 노드출력은 일반적으로 단지 고장징후를 인식하는 부분이므로, 여기서 조금의 어떤 고장징후 확신도를 1이라 하면 반드시 그 고장이 존재하는 것으로 하고, 0이면 그고장 징후가 존재하지 않는 것으로 하여 0.5보다 클 때 징후가 존재하는 것으로 고려하고, second level에서 출력은 first level에서 인식된 고장징후의 확실성을 검증해 주는 부분으로 이것의 출력은 확신도 0.5로는 불충분하므로 거의 고장확신도가 0.9보다 큰 경우에 아이템 고장이 존재하는 것으로 하였으며, third level에서는 노드출력이 0.5이상이면 component속에 고장징후가 존재하고, forth level에서는 출력노드가 0.9 이상이면 component에 고장이 존재하는 것으로 한다.

Fig.8은 본 연구에서 개발된 계층형 신경망의 flow chart를 나타낸 것이다.

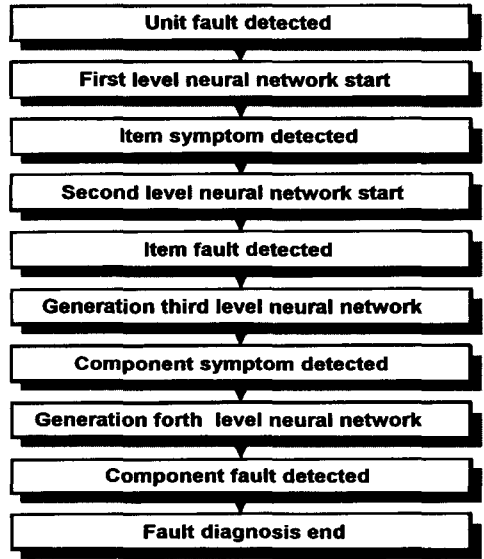


Fig. 8 Flowchart of Hierarchical Neural Network

먼저 유니트에서 고장이 감지되면 시그널 전처리기에서 FFT 고장특징 패턴이 first level신경망 요소에 입력시키면, 이상징후가 있는 유니트로부터 고장패턴이 감지되면 first level 신경망은 second level 신경망을 생성하는데, 형성되는 신경망은 이미 사전에 오프라인으로 학습된 상태에서 조절(turning)된 weight값으로 고장분류준비를 완료하여, 각각의 아이템 센서를 통하여 입력된 고장패턴을 분류한다. Second level신경망 개는 각각의 7개 아이템들의 각각의 고장에서 비롯된 다중고장 징후를 학습하고 있다. 여기서 first level의 신경망의 갯수는 1개이고 second level의 신경망의 갯수는 7개이다. 이 결과는 OR프로세스로 넘겨져서 third level의 하나의 신경망은 first level과 같은 역할을 수행하는데 하나의 아이터 속에는 7개의 요소(component)가 포함되어 있다고 가정한다. Fig.9는 전체 계층형 신경망의 작동형태를 나타낸 것으로 X-Window Motif를 사용하여 구현하였다. 4가지 색으로 구분되는 각각의 창들은 하나의 유니트를 나타낸 것으로 (a)는 고장유니트가 2개인 경우이고, (b)는 3개, (c)는 4개일때의 구현모습을 나타낸 것이다. 유니트당 요소신경망의 계수를 계산해 보면 64개이고 만약 4개의 유니트가 다중고장을 일으켰을 때 동시에 형성되는 요

4. 결론

고장전파속도가 매우 빠르고, 미세요소의 고장이 전체 시스템 고장으로 전파되는 특성을 지닌 자동화 복합시스템은 주로 모듈별로 분할되어 있고, 고장물리 특성상 고장원인은 한 가지 물리현상이라기 보다는 다수의 복합된 물리현상의 요인에 의하여 발생되므로 한꺼번에 여러 개의 고장이 동시에 일어날 수 있으므로 초기 빠른 고장진단이 필수적이며, 각각 물리현상에 대한 지식을 종합하여 진단해야 하는 복합 다중문제 해결기법에 관한 연구가 필요하며, 이를 위해서 이들 소규모 진단 모델 사이의 정보교환이 절대 필수적이다.

따라서 본 연구에서는 각기 다른 기능을 수행하는 복합적인 요소로 설계된 장치에서 고장의 종류와 위치를 정확히 구별해 내기 위한 실시간 지적 다중고장 진단시스템을 지원하기 위한 하나의 계층화 다중병렬처리 신경망을 개발하였다. 이것은 하나의 서브시스템이 고장을 진단하는 도중 다른 하나의 서브시스템에서 고장이 일어나면, 기존의 진단프로세스를 그대로 유지하면서 다른 진단 프로세스를 시작할 수 있도록 멀티프로세싱기법을 이용하였다. 본 연구의 핵심은 실제 시스템을 한꺼번에 맞추어서 고장을 낸다는 것은 거의 불가능하기 때문에 유기적인 연결 동작에 바탕을 둔 모듈화된 복합시스템의 계층구조에 바탕을 둔 계층형 신경망을 개발하는 것이다. 각각의 신경망셀은 주 신경망셀의 필요에 따라 생성되거나 소멸시킨다. 각각의 모듈간 제어는 메시지전달에 의하여 행하여 진다. 본 시스템에서 제시된 제어기법은 UNIX운영체제 IPC(Inter Process Communication)설비를 이용하였다.

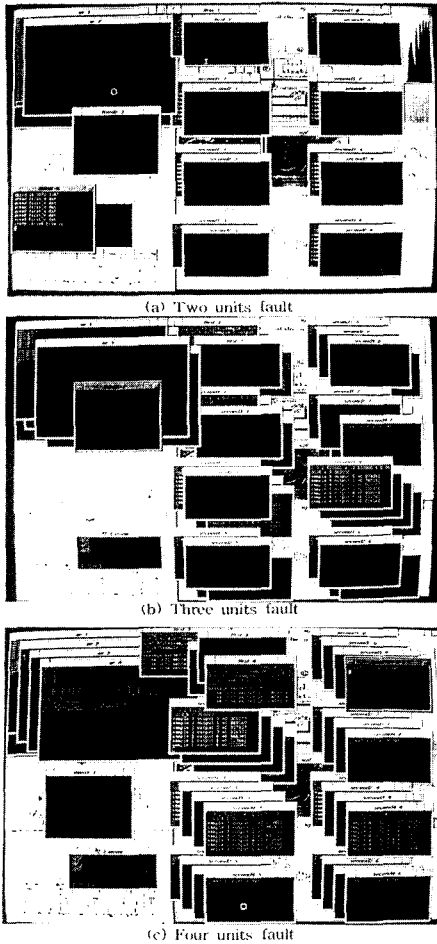


Fig. 9 Picture of hierarchical neural network

소 신경망의 수는 256개의 요소신경망이 동시에 멀티타스킹을 하면서 고장진단을 한다. 하나의 유닛의 고장진단 완료시간은 대략 270초 정도 소요되었고, 두 개의 유닛 고장진단 시간은 580초 정도 소요되었으며, 세 개의 유닛 고장진단 시간은 910초 정도로 고장유닛 수가 늘어남에 따라 네 개 유닛 고장진단시 1430초 정도로 유닛 수가 늘어나면서 약간씩의 시간 지연을 나타내었는데 이것은 운영체제(Operating System)에서의 프로세스 스케줄링에 부하가 늘어남으로 인하여 생긴 것임을 알 수 있다. 본 계층형 신경망은 SUN Sparc I 워크스테이션에서 수행되었다.

참고 문헌

1. Chryssolouris, G., Domroese, M. and Beaulieu, P., "Sensor Synthesis for Control of Manufacturing Processes", Transactions of the ASME, Vol.114, pp.158-174, 1992.
2. Monsen, P.T., Manolakos, E.S. and Dzwonczyk, M., "Helicopter gearbox fault detection and diagnosis using analog neural networks", Conference Record of the Asilomar Conference on Signals, Systems & Computers, Vol.1, pp.381-385, 1993.

3. Guhmann, C. and Filbert, D., "Fault diagnosis of electric low-power by analyzing the current signal", IFAC Symposia Series, No.6, pp.141-146, 1992.
4. Wang, K. "Rocket engine fault diagnosis by using neural networks", Journal of Propulsion Technology, Vol.4, pp.18-23, 1993.
5. Yang, J.O. and Jang, S.H., "Diagnostic expert system for the nuclear power plant based on the hybrid knowledge approach", IEEE Transactions on Nuclear Science, Vol.36, No.6, pp.2450-2458, 1989.
6. Narazaki, H. and Iwatani, T., "AI tool and its applications to diagnosis problems", ISIJ International, Vol.30, No.2, pp.98-104, 1990.
7. Comly, J.B., Bonissone, P.P. and Dausch, M.E., "Fuzzy logic for fault diagnosis", Proceedings of the International Society for Optical Engineering, Vol. 138, pp.390-400, 1991.
8. Becraft, W.R., "Diagnostic applications of artificial neural networks", Proceedings of the International Joint Conference on Neural Networks, Vol.3, pp.2807-2810, 1993.
9. Stanely, G.M., "Neural networks for fault diagnosis based on model errors or data reconciliation", Advances in Instrumentation and Control : International Conference and Exhibition, Vol.48, pp.1701-1710, 1993.
10. Geffner, H., "Distributed Diagnosis of Systems with Multiple Faults", Proceedings of 3rd conf. on AI Applications, pp.156-162, 1987.
11. Joel, R., "Diagnosing multiple faults in SSM/PMAD", Proceedings of Intersociety Energy Conversion Engineering Conference, Vol.1, pp.284-290, 1990.
12. Tzafestas, S., Christou, T. and Yip, Y.T., "Multi-fault diagnosis of digital systems using nonmonotone reasoning", Proceedings of the IEEE Conference on Decision and Control, Vol.3, pp.2604-2608, 1991.
13. Padalkar, S., Karsai, G., Sztipanovits, J., Okuda, K. and Miyasaka, N., "Real-time fault diagnostics with multiple aspect models", Proceedings-IEEE International Conference on Robotics and Automation, Vol.1 pp.803-808, 1991.
14. Zheng, X., Yang, S., Zhou, A. and Shi, H., "A Knowledge-Based Diagnosis System for Automobile Engines", The International Journal of Advanced Manufacturing Technology, Vol.3, No.3, pp.159-169, 1988.
15. Narayanan, N., "A Methodology for Knowledge Acquisition and Reasoning in Failure Analysis of Systems", IEEE Transaction on Systems, Cybernetics, Vol.17, NO.2, pp.274-288, 1987.
16. Stevens, W.R., Advanced Programming in the Unix Environment, Addison Wesley, pp.188-212, 1992.
17. 홍릉과학출판사 편집부, UNIX 시스템 프로그래밍, 홍릉과학출판사, pp.238-247, 1991.
18. 배용환, 자동차 시스템의 실시간 지적 다중고장 진단 제어기법에 관한 연구, 부산대학교 박사학위 논문, 1996.