

선박엔진 통합시스템의 실시간 고장진단 기법에 관한 연구

정경열* · 김달현** · 류길수***

A Study on the Real-time Fault Diagnosis of Integrated Ship Engine Systems

K. Y. Chung · D. H. Kim · K. S. Rhyu

〈목 차〉

- | | |
|-----------------------|-------------------|
| I. 서 론 | V. 고장진단시스템의 세부 모듈 |
| II. 선박엔진 통합시스템의 전체 구성 | VI. 결 론 |
| III. 전문가 시스템 | * 참고문헌 |
| IV. 고장진단 시스템의 데이터베이스 | |

Abstract

This paper presents the diagnostic method of an integration system where both diagnosis system and monitoring system are combined. Knowledge base for ship engine fault diagnosis is implemented using hybrid knowledge representation technique by EE(Element Expert) shell, and using a certainty-factor method to handle uncertainties of evidences and rules. Whenever alarm occurs, the monitoring system sends alarm data to the diagnosis system. At the same time, the diagnosis system converts alarm data type to diagnosis data type, verifies alarm data, and calculates data's trend. To map alarm data to evidences of knowledge base, the diagnosis system uses two databases. The real-time diagnosis results are provided by a reasoning mechanism implemented using visual C++ for driving OLE automation server of EE shell.

* 한국기계연구원

** 한국해양대학교 대학원 자동화정보공학부

*** 한국해양대학교 자동화정보공학부 교수

I. 서 론

선박엔진 고장진단 시스템이란 모니터링 시스템에서 수집한 선박기관실내의 모든 기기들에 대한 실시간 데이터를 이용하여 현재의 선박기관실의 고장을 어떻게 신속, 정확히 판단하는가에 대한 문제를 해결하는 것을 목표로 하는 시스템이다.[1,2] 선박엔진에서 발생하는 고장은 선박의 운용에 치명적인 영향을 줄 수 있으므로 빠른 인식과 처리가 요구된다. 현재는 선박기관사의 경험에 의해 처리되는 경우가 대부분이다. 그러나 최근에는 승선기피 현상이 많아져서 고도의 기술을 가진 기관사가 부족해지고 있는 실상이다. 이에 따라 우리는 작은 고장이 대형사고로 이어지는 경우를 종종 목격하곤 한다. 이에 대한 대처 방안으로 인공지능을 이용한 고장진단 시스템의 구축이 필수적이라고 할 수 있다. 모니터링 시스템과 고장진단 시스템을 선박엔진 통합시스템이라 한다. 최근 PC 하드웨어의 발전으로 처리속도가 현격하게 증가하고 있으며, 실시간 시스템의 구축에 필수적인 멀티태스킹 기법을 지원하는 OS와 개발 툴 들이 많이 제공되고 있다. 현재 가장 많은 사용자층을 확보하고 있는 Windows 98을 대표적인 예로 들 수 있으며, 실시간 시스템을 구성하기에 충분한 기능을 제공한다.

본 논문에서는 선박엔진 통합시스템에서의 실시간 고장진단에서 사용될 전문가시스템과 모니터링 시스템간의 통합을 목표로 하고 있다. 이를 위해 알람 데이터들의 처리방법, 처리된 데이터를 이용한 고장진단방법 등에 대하여 논한다.

II. 선박엔진 통합시스템의 전체 구성

선박기관실 통합시스템은 Fig.1과 같이 구성되어 있으며 대형선박을 대상으로 한다[1]. Local Unit (제어시스템)들은 선박기관실 기기의 데이터를 측정하거나 제어한다. Main Unit(모니터링시스템)은 선박기관실 기기의 데이터를 수집하여 현재의 알람상태를 계산하고 화면에 알기 쉽게 표시해 주거나 사용자의 입력에 의한 선박기관실 기기의 제어 정보를 전송한다. Gateway는 Local Unit과 Main Unit의 분산적인 데이터를 통합하여 전송한다. Main Unit들 및 고장진단 시스템과 Gateway와의 통신은 LAN을 이용한 TCP/IP 프로토콜을 이용하고, 접속상태에 관계없이 동작을 하기 위해 브로드캐스트 방식을 이용한다. Local Unit들과 Gateway사이의 통신은 표준 RS-485를 이용한 마스터/슬레이브 형태의 통신 방식을 이용한다. Diagnosis Unit(고장진단시스템)은 Main Unit에서 전송되는 알람 정보와 gateway에서 전송되는 데이터를 이용하여 알람에 대한 고장을 진단한다.

III. 전문가 시스템

고장진단을 위해서는 인공지능 구현 방법의 하나인 전문가 시스템기법을 이용한다. 전문가 시스템을 처음부터 구축하는 것은 매우 어려우며 많은 비용이 소모되므로 본 연구에서는 ND사의 EE에서

제공하는 OLE Automation Server를 이용한다.

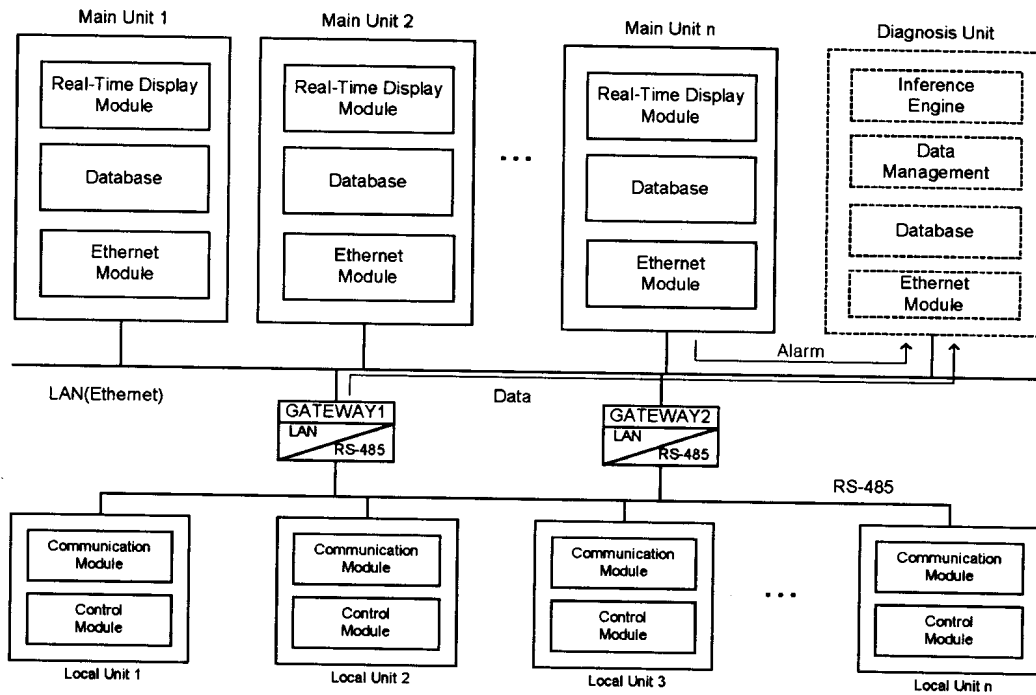


Fig.1 Structure of an Integrated Ship Engine System

1. 일반적인 전문가 시스템

전문가 시스템은 Fig.2와 같이 크게 4가지의 모듈로 분류된다. User Interface는 각각의 Working Memory, Inference Engine, Knowledge Base에 대한 명령어들을 입력받아 수행한다. Working Memory

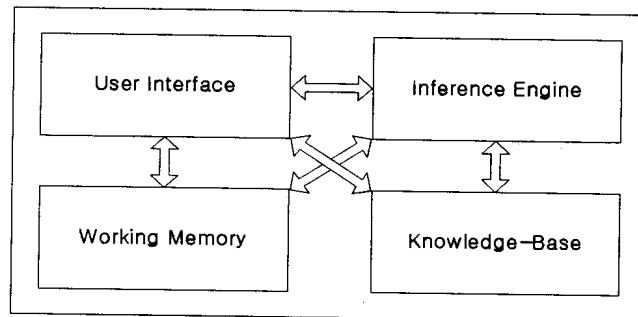


Fig.2 General Expert System Structure

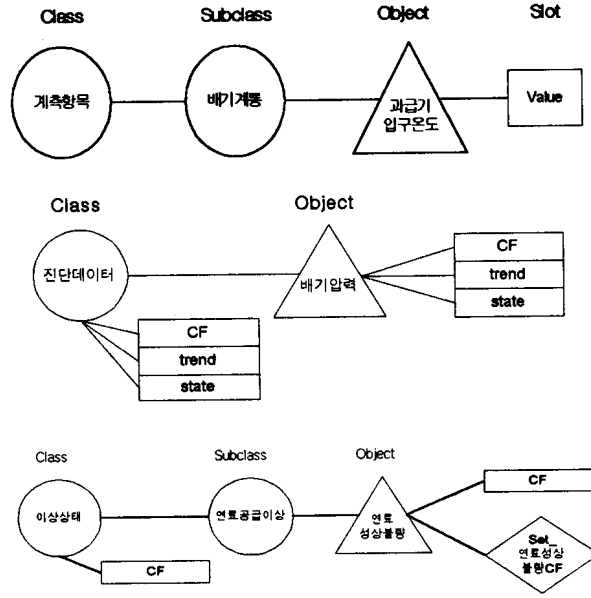


Fig.3 Classes and Objects Structure

는 현재의 작업에 대한 데이터를 저장한다. Knowledge Base는 추론에 사용되는 지식을 데이터의 형태로 변환하여 저장한다. Inference Engine은 지식과 데이터를 이용하여 추론하는 기능을 가지며 인간의 사고기능에 해당한다.

2. 선박엔진의 객체지향분석 및 확신도 속성정의

엔진의 주요 구성요소를 객체로 정의하고, 이것을 계측항목 클래스로 통합하고, 센서의 계측항목을 객체의 속성으로 정의하였다. 계측항목 클래스에 확신도 속성을 정의하여 상속관계에 의해서 모든 계측항목에 발생한 경보의 종류와 트랜드에 의해서 구해진 확신도 값이 저장되게 된다. 이상상태의 경우에는 "Abnormal State" 클래스로 정의하고 확신도 속성을 가진다. 세부 항목은 이를 상속받는 객체로 정의하였다. 확신도를 계산하는 메소드(Method)를 클래스에 정의하여 경보발생시 각 이상상태의 확신도를 계산할 수 있게 하였다.

3. 확신도 기법의 적용

본 연구에서는 모니터링 시스템으로부터 받은 알람의 불확실성과 전문가의 조언으로 만들어진 결정테이블의 수치들의 불확실성이 있다. 정보의 불확실성을 해결하기 위하여 Fig.4와 같이 현재 알람의 종류와 트랜드에 따라 각각의 확신도를 정하여 세부적으로 분류하였다. 계측항목과 이상상태의

관계를 강·중·약·무의 단계로 정하고 이에 따른 확신도를 정한다. 결론에 대한 확신도는 두 가지 확신도의 곱으로 구한다[2,3].

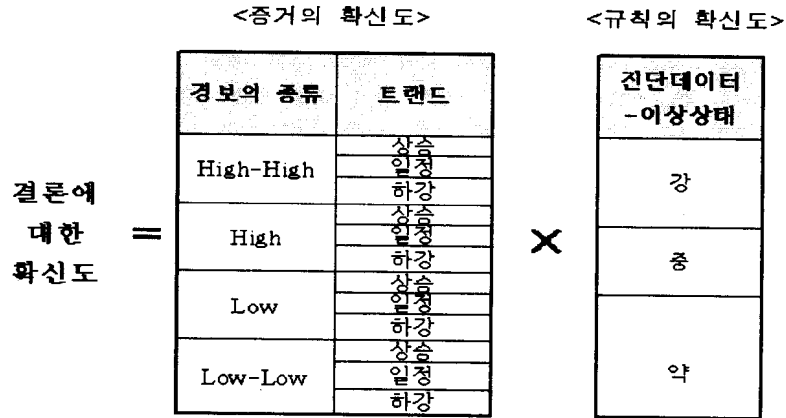


Fig.4 Calculus Method of a Certainty Factor for Conclusion of a Rule

4. 역방향 추론

고장진단에서는 역방향 추론을 이용하여 고장의 원인을 분석한다. 먼저 진단을 위해 확신도를 계산한다. 확신도의 계산은 역방향 추론을 하는 형태로 구성하였다. Calculate_CF규칙이 suggest되면 역방향 추론을 시작한다. Calculate_CF규칙에는 Diagnosis_Data 객체의 state와 trend값에 따른 확신도를 계산하는 H_Increase, H_Constant, H_Decrease, L_Increase등의 규칙이 연결되어 있다. Fig.5는 H_Decrease규칙을 보여준다. 어떤 Diagnosis_Data 객체가 “High”와 “Decrease”상태에 있을 경우 객체의 CF에 지정된 확신도 값을 저장한다.

Fig.6은 고장진단을 위한 규칙의 일부분이다. 먼저 역방향 추론을 위해 Diagnosis_Rule 이 suggest 된다. Diagnosis_Rule은 각 고장종류에 대한 규칙에 연결되어 있으며 Diagnosis_Rule에서 모든 규칙의 탐색이 가능하다. Diagnosis_Rule이 suggest되면 각각의 고장에 대한 규칙을 탐색하게 된다. 각각의 규칙은 적당한 알람의 항목과 연결되어 있다. 그림은 고장항목을 확인하는 부분만 나타나 있다.

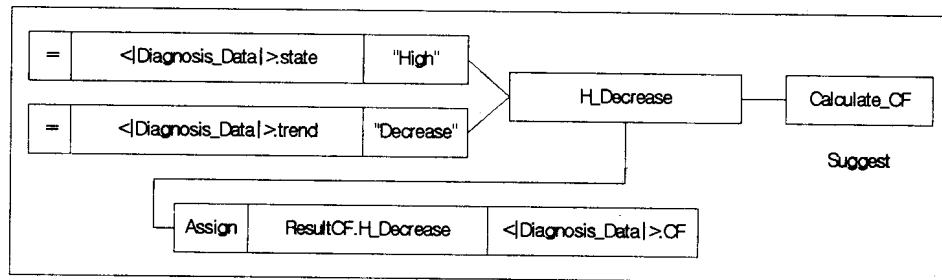


Fig.5 Rule for Calculating Certainty Factor

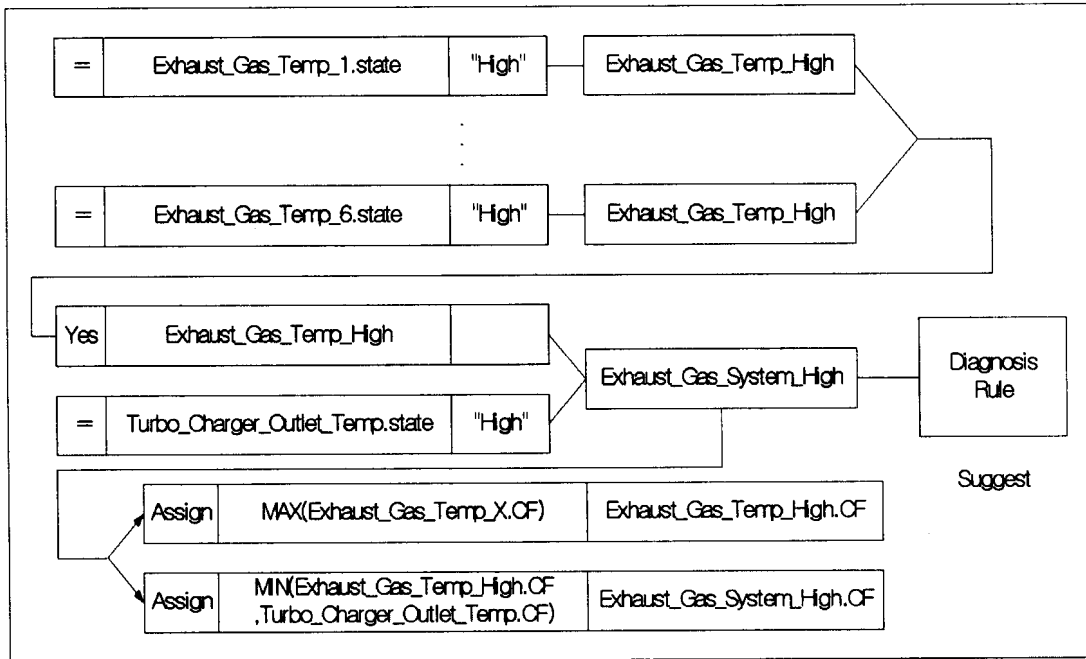


Fig.6 Rule for Diagnosis

5. OLE Automation Server

인공지능 툴을 이용할 경우 실시간 고장진단을 구현할 수 없다. 이러한 문제를 해결하기 위하여 툴에서 제공하는 라이브러리를 이용하였다. 라이브러리는 ND사에서 제공하는 컴포넌트로 EE에서 제공하는 전문가시스템의 모든 기능을 프로그램 개발에 사용할 수 있게 한다. [5]-[8]

6. 선박엔진 고장진단시스템의 구성

Main Unit에서 전송되는 알람데이터를 고장진단에 사용한다. 시스템은 크게 Data Acquisition thread, Trend Calculate thread, Inference Thread로 나뉘어 지며 각 모듈의 관계는 Fig.7과 같다. 각각의 스레드는 데이터의 수집, 진단데이터로 변환, 추론을 한다. 데이터베이스와 추론을 처리하기 위하여 외부 라이브러리를 사용한다. 멀티 스레드는 기법을 사용하여 병렬처리되며 실시간 고장진단이 가능하다. 데이터의 정확성을 위해 상호배제 및 이벤트 기법을 사용하여 동기화 되어 동작한다.

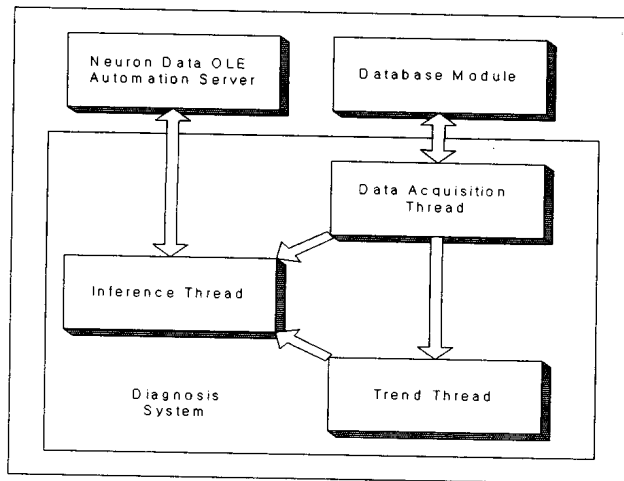


Fig.7 Structure of a Fault Diagnosis System

IV. 고장진단 시스템의 데이터베이스

본 연구에서는 gateway에서 전송되는 데이터를 고장진단시스템에서 활용 가능한 형태로 변환하기 위해 데이터베이스를 이용하고 있다. 빠른 처리를 위하여 계산된 알람정보를 변환 없이 쓰는 직접사상 데이터베이스와 지식베이스에 적당한 형태로 변환시켜 주는 변환 데이터베이스의 두 가지로 나뉘어 진다. 모니터링 시스템에 있는 데이터베이스와 연계하여 동작할 수 있도록 구성하였다.

1. 직접사상 데이터베이스

Fig.8과 같이 지식베이스의 계측항목에 있는 것으로 알람상태를 변환 없이 사용할 수 있는 데이터들에 대한 정보를 모아 놓은 것이다. 저장되는 정보는 지식베이스에서 사용되는 객체의 이름과 그에 대한 모니터링시스템에서 사용되는 알람데이터베이스의 ID의 번호이다. 빠른 응답시간을 보장하기 위하여 변환 데이터베이스와 구분하여 작성하였다.

Description	Sensor ID
Cooling_Fresh_Water_Cooler_Inlet_Temp_1	145
Cooling_Fresh_Water_Cooler_Outlet_Temp_1	146
Engine_Inlet_Cooling_Fresh_Water_Temp	18
Engine_Outlet_Cooling_Fresh_Water_Temp	19
Engine_Outlet_Lubricating_Oil_Temp	0
Exhaust_Gas_Back_Pressure	0
Exhaust_Gas_Temp_1	49
⋮	

Fig.8 Direct Mapping Database

2. 진단데이터 변환 데이터베이스

알람정보를 사용할 수 없는 진단데이터의 계산을 위한 데이터베이스이다. 고장진단의 경우 두 가지 항목에 대한 연산결과를 이용하여 진단을 행하는 경우가 있다. 이를 처리하기 위하여 Fig.9와 같은 데이터베이스를 작성하였다.

- Description : 지식베이스의 Diagnosis Data 객체의 이름으로 데이터를 전송하기 위한 포인트로 사용된다.
- First ID : 첫 번째 피 연산자.
- Second ID : 두 번째 피 연산자.
- Operator : 두 개의 피 연산자에 대한 연산방법을 지정한다.
(0 : force Normal, 1 : Subtract)
- LL : Low_Low 경고를 위한 경계 값.
- AL : Low 경고를 위한 경계 값.
- AH : High 경고를 위한 경계 값.
- HH : High_High 경고를 위한 경계 값.

모니터링 시스템에서 지정된 항목에 대한 데이터 전송이 없다면 Operator의 값이 0이 되며 현재 상태는 Normal상태를 가진다. LL, AL, AH, HH는 각각의 알람에 대한 경계 값을 저장한다.

Description	First ID	Second ID	Operator	LL	AL	AH	HH
Cooling_Fresh_Water_Cooler_Inlet_and_Outlet_Differencial_Temp_1	145	146	1	20	50	0	0
Cooling_Fresh_Water_Cooler_Inlet_and_Outlet_Differencial_Temp_2	177	178	1	20	50	0	0
Lubricating_Oil_Cooling_Fresh_Water_Differencial_Temp	0	0	0	0	0	0	0
Turbo_Charger_Inlet_and_Outlet_Differencial_Temp_A	56	44	1	0	0	30	50
Turbo_Charger_Inlet_and_Outlet_Differencial_Temp_B	57	45	1	0	0	30	50

Fig.9 Conversion Database

V. 고장진단시스템의 세부 모듈

1. Data Acquisition thread

이 스레드는 Fig.10에서 보여주는 것과 같이 먼저 gateway에서 전송되는 데이터와 Main Unit에서 전송되는 알람정보를 수집한다. Main Unit에서 알람정보를 전송할 경우 데이터를 이용하여 알람정보를 계산한다. 계산된 알람정보와 Main Unit에서 전송된 정보가 다를 경우 Main Unit에 데이터베이스의 업데이트를 요구한다. 알람이 발생하였을 경우 알람발생 유·무를 Inference thread에 전송한다. 알람이 발생하였을 경우 gateway에서 전송된 데이터를 고장진단의 형태로 계산한다.

고장진단은 선박기관실 기기의 고장으로 한정하며 모니터링 시스템의 고장은 처리하지 않는다.

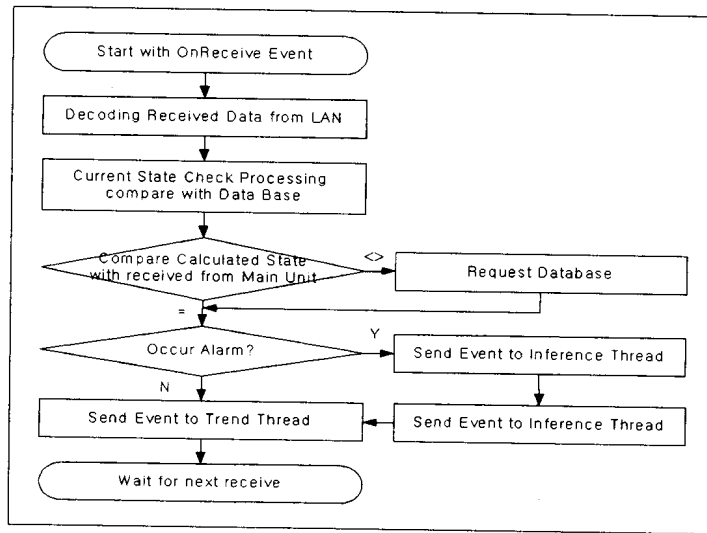


Fig.10 Data Acquisition thread Algorithm

2. Trend Calculate thread

Data Acquisition thread에서의 계산종료 이벤트 전송을 기다린다. 현재 데이터를 상승·하강·유지의 상태로 계산한다. 데이터의 급격한 변화는 실제 엔진에서 발생 할 수 없기 때문에 오류로 가정한다. Fig.11에서 보여주는 것과 같이 데이터의 오류를 막기 위하여 트렌드는 일정수의 이전데이터의 평균과 현재 값의 비교를 이용하여 계산한다. 저장되는 이전데이터의 수가 많을 경우 트렌드의 변화에 대한 반응이 느려지며, 적을 경우 오류에 대한 보정율이 낮아진다. 데이터의 수는 실험에 의해 결정하였다.

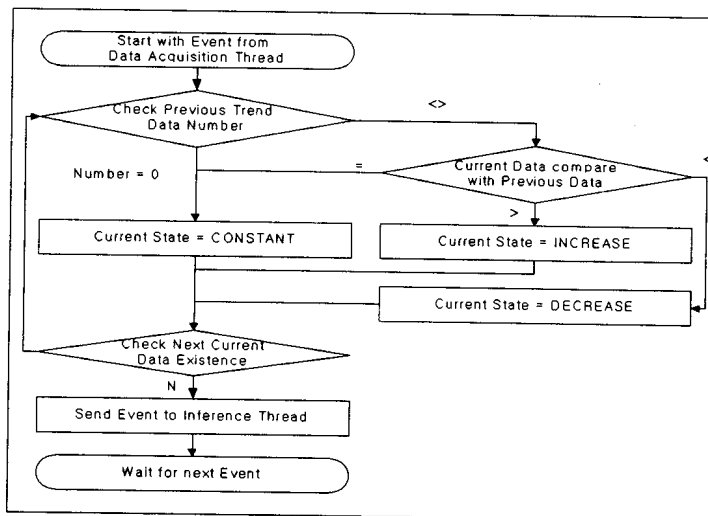


Fig.11 Trend Calculate thread Algorithm

3. Inference thread

Data Acquisition thread와 Trend Calculation thread에서 계산된 정보를 OLE Automation Server로 전송하여 추론하며 Fig.12의 그림과 같이 동작한다. 먼저 직접사상 데이터베이스에 존재하는 데이터를 검색하여 전송한다. 변환 데이터베이스의 정보를 검색하여 알람정보를 전송한다. 모니터링시스템에서 사용되지 않는 항목은 "Normal" 상태로 두어 진단에 영향을 주지 않도록 한다. Calculate_CF 규칙을 suggest하여 확신도를 계산한다. 계산이 끝나면 Diagnosis_Rule을 suggest하여 추론을 시작한다. OLE Automation Server는 추론중에 사용자의 프로그램에서 처리되어야 하는 것에 대한 2가지의 핸들러를 제공한다. 첫 번째 Remote Question 핸들러는 추론에 필요한 데이터가 충분하지 않을 경우에 프로그램에게 데이터 요청하기 위해 사용된다. 두 번째 Remote Execute 핸들러는 사용자프로그램에서 처리해야 할 명령어에 대한 처리권한을 넘겨준다. 지식베이스에서 결과를 출력하는 프로그램 루틴은 사용자의 프로그램에 맞게 변형되어야 한다.

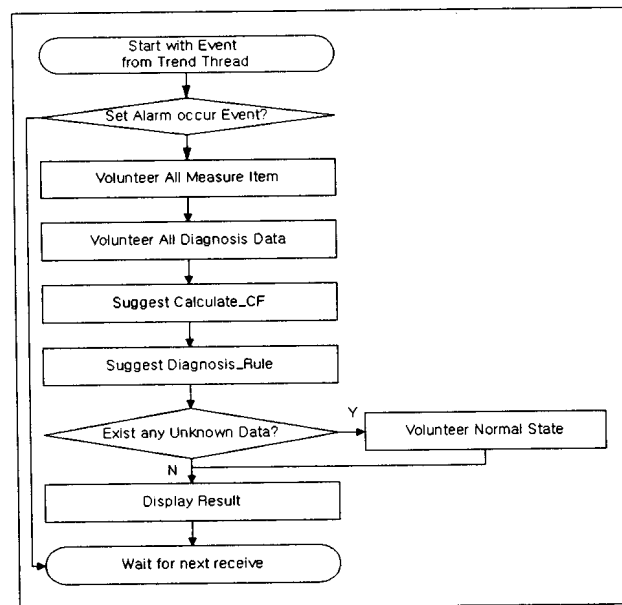


Fig.12 Inference thread Algorithm

VI. 결 론

본 논문에서는 선박엔진 통합시스템에서의 실시간 고장진단 기법에 관하여 논하였다. 선박기관실 기기의 데이터는 LAN을 이용한 TCP/IP 통신기법을 이용하였고, 모니터링시스템과 고장진단을 위한 전문가시스템의 통합에 중점을 두었다. 모니터링 시스템과 고장진단 시스템이 하나의 프로그램으로

구현될 경우 처리해야 할 데이터가 많아 복잡해지기 때문에 별도의 시스템으로 온라인 상에 구현하였다. 실시간 처리와 멀티태스킹 기법을 적용하기 위하여 상호배제와 이벤트를 Time-out기법과 같이 사용하여 충분히 빠른 응답속도를 가질 수 있게 하였다.

현 단계에서는 지식베이스가 일부 기기에 한하여 구축되어 있기 때문에 알람을 지식베이스에 맞추어 변환하는 기능이 매우 간단하다. 방대한 지식베이스를 완성시키기 위해서는 추가적인 연산 기법에 대하여 연구할 필요가 있으며 사용자의 입력에 의한 진단기능도 지원해야할 사항이다. 또한 설명 기능에 대해서도 계속적인 연구를 진행해야 할 것이다.

참고문헌

- [1] 최재곤, “선박 기관실 모니터링 시스템을 위한 실시간 데이터 처리에 관한 연구”, 한국해양대학교 석사학위논문, 1998.
- [2] 최옥현, “선박 엔진 고장진단 전문가시스템의 구현에 관한 연구”, 한국박용기관학회 1998 추계학술대회 논문집.
- [3] Joseph Giarratano, Gray Riley “Expert Systems, Principles and Programming”, PWS, 1994
- [4] John Durjin, “Expert Systems Design and Development”, Prentice Hall.
- [5] “Element Environment User’s Guide”, Neuron Data.
- [6] “Element Environment IRE Language Programmer’s Guide”, Neuron Data.
- [7] “OOScript Language Reference”, Neuron Data.
- [8] 전병선, “ActiveX 프로그래밍 가이드”, 파워북. 1997.

