

원전 열교환기 Tubesheet Map 자동 Drawing 기능 구현

송재주, 신진호, 이봉재
한국전력공사 전력연구원
e-mail:jjisong@kepri.re.kr

The Implementation of Tubesheet Map Automatic Function for Nuclear Power Plant Heat Exchanger

Jae-Ju Song, Jin-Ho Shin, Bong-Jae Yi
KEPRI(Korea Electric Power Research Institute)

요 약

열교환기(Heat Exchanger)는 원전 운영에 핵심적인 설비이다. 따라서 열교환기 전열관의 건전성 유무를 판단 키 위해 정기적으로 비파괴검사의 일종인 와전류검사(ECT, Eddy Current Testing)를 수행하고 있다. 이러한 와전류검사 공정은 크게 3가지로 분류할 수 있다. 첫 번째는 전열관 상태검사를 위한 신호데이터 취득공정이고, 두 번째는 취득된 신호를 판독하여 전열관의 건전성 여부를 진단하는 평가공정, 마지막으로 평가공정에서 발생하는 방대한 데이터를 토대로 추이분석 자료를 제공할 수 있는 공정이 필요하다. 본 논문에서는 세 번째 공정에서 필요한 열교환기 Tubesheet를 사용자 정의에 따라 자동으로 구현할 수 있는 기능을 정의하고, 평가공정에서 발생한 데이터를 직접 Tubesheet상에 Mapping 처리하여 시각적으로 전 열관 상태를 확인할 수 있는 프로그램 구현방법을 제시한다.

1. 서 론

원자력발전소가 효율적으로 운영되기 위해서는 수많은 설비가 유기적으로 결합되어 각각의 기능을 최대한 발휘 해야 적절한 발전량을 항상 유지할 수 있다. 특히 원전 핵심설비로서 증기발생기(Steam Generator)와 열교환기(Heat Exchanger)는 설비의 안정성과 그 효율적인 측면에서 상당히 중요한 설비로 간주되고 있다. 발전소 가동연수가 계속 증가함에 이러한 설비의 이상유무를 판단하는 것은 원전 운영업무에 핵심 요소로 취급되고 있다. 설비 이상유무 검사방법으로 비파괴검사의 일종인 와전류탐상검사가 있다. 와전류탐상 검사란 특정한 설비에 와전류신호를 작용시켜 그 변조된 신호를 수집하고, 수집된 신호를 바탕으로 평가자가 그 신호를 판독하여 건전성 유무를 검사하는 방법이다. 이 검사과정에서는 상당한 분량(매 1주 기마다 보통 40만 건 정도)의 전열관 상태정보가 생성된다. 이러한 데이터는 향후 전열관 분석정보 제공에 기초자료로 활용되기 위해 체계적으로 보관되어야 하며, 이를 바탕으로 담당자에게 시각적으로 전열관 정보를 제공키 위한 새로운 전산 시스템이 필요하게 되었다. 한편 전력연구원에서는 원자력발전소 담당부서의 요구를 반영하여 열교환기 차수별 검사정보 데이터베이스를 구축하고, 현재 수작업에 의존하여 처리한 Tubesheet Drawing작업 및 검사

정보 Mapping처리 프로그램을 개발중에 있다. 본 논문에서는 이러한 열교환기 전열관 Tubesheet Map 자동 Drawing 기능, 편집기능, 저장기능 그리고 데이터베이스화 되어 있는 검사정보를 해당 Tubesheet에 Mapping 기능 등을 포함한 시스템 구현과정을 설명한다. 본 논문의 구성은 사용자 요구사항분석 내용, Tubesheet Map Drawing 기능 설계내용, 데이터베이스 관계 및 스키마정의 내용을 소개하며 마지막으로 주요 구현방법에 대해 설명한다.

2. Tubesheet 자동 Drawing 시스템 설계

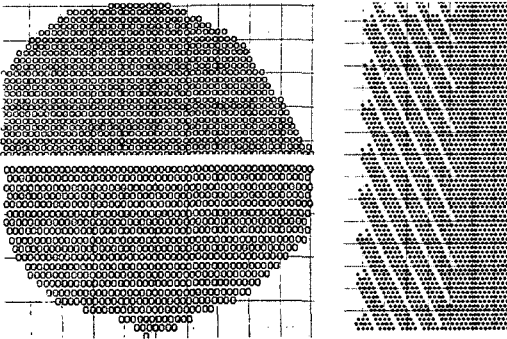
2.1 시스템 요구기능 분석

TubeSheet Map 자동 Drawing 시스템은 다음과 같은 사항을 고려하여 구현한다.

첫째, [그림2-1]과 같이 다양한 모양과 많은 종류(약300~400여개)의 열교환기 TubeSheet Map을 Drawing하기 위해 사용자가 쉽게 접근할 수 있는

[그림 2-1] 다양한 열교환기 Tube 모양

직관적인 Interface를 제공하고, Tube를 하나의 객체

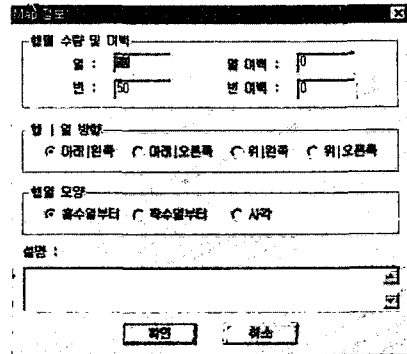


로서 인식할 수 있는 행/열(Row/Column) 번호를 부여할 수 있는 기능도 포함하도록 구현한다. 또한 여러 가지 Tube 배열이 있는 점을 감안하여 모양 선택기능을 사용자가 정의할 수 있도록 한다. 두 번째로 생성된 Tubesheet 모양의 영역별(Zone) 분할기능을 제공하도록 한다. Tube 전열관 개수가 10,000개 이상인 복수기 등과 같은 열교환기는 6개의 영역으로 분리하여 관리할 수 있도록 한다. 세 번째는 편리한 수정 편집 기능이 Toolbar 형태로 제공되도록 한다. 대칭된 Tube 편집을 위한 분할 선택기능, Box 및 다각형 영역 선택 기능, 선택/비선택 Toggle 기능 등이 포함되도록 한다. 또한 영역별 분할하여 Zone명 부여기능과 해제기능도 역시 포함하도록 한다. 네 번째로 완성된 Map은 데이터베이스로 저장기능이 필요하다. 저장은 오라클 DBMS를 사용하고, 각 Tube 행/열 번호를 Key로 하여 검사정보 Tubesheet Map 상에 Mapping시 사용할 수 있도록 구현한다. 마지막으로 검사정보 데이터베이스에서 해당 Tube의 결함현황을 읽어 생성된 Tube Map 정보와 일치되는가를 확인한다. 이미 검사정보는 행/열 번호를 기준으로 데이터베이스로 구축되어 있다고 전제한다.

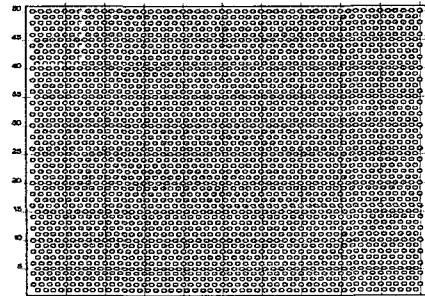
2.2 User Interface 구성

이 시스템 사용자는 설비도면상에서 전체 Tube 개수 및 열교환기 형상을 인식하고 있는 것으로 간주한다. 우선 필요한 Drawing 대상 행 방향 Tube 개수와 열 방향 Tube 개수를 입력받는다. Tube 배열시 간격을 조절하기 위한 Number형 입력 Text Item를 포함하도록 한다. 그리고 행/열 진행방향을 [그림2-2]과 같은 4가지 항목을 라디오 그룹에서 선택할 수 있도록 하며, 행/열 모양도 3가지로 선택할 수 있도록 User Interface를 구현한다. 입력받은 결과는 행 * 열 개수로 사각형 Box안에 원(O) 모양 형태로 [그림2-3]과 같이 화면에 Display 되도록 한다. 정확한 Tube 편집 작업을 위해 가로 세로 Grid를 표시하고, 일정한 간격으로 행/열 번호를 표시한다. 또한 마우스 이동시 행/열 번호를 확인할 수 있도록 화면하단에 Status Line를 사용해 Tube 번호를 표시하도록 한다. 이상과 같은 작업이 완료되면 사용자는 [그림2-4]와 같은 편집 Toolbar를 이용해 원하는

형태의 Tubesheet Map을 점진적으로 Drawing 할 수 있도록 한다.



[그림 2-2] 기본정보 입력화면



[그림 2-3] 편집 작업전의 TubeSheet Map

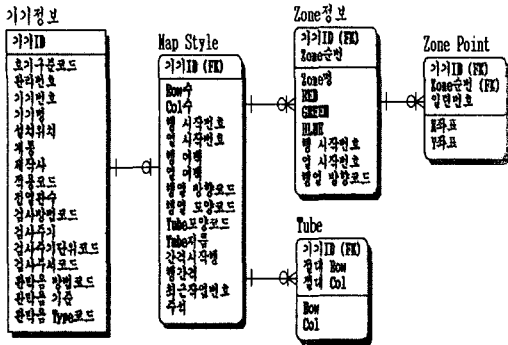
[그림 2-4] 편집 Toolbar

2.3 DB 모델링 및 스키마 정의

이 데이터베이스는 관계형 DBMS인 오라클을 사용하며, 기기 정보 Table의 기기ID를 Primary Key 하여 Map Style, Zone 정보, Zone Point Table 정보와 관계가 이루어지도록 한다. Map Style Table에서 각 행/열 번호의 절대, 상대 번호를 가지고 있도록 하여 검사정보와 Mapping시 활용토록 한다. 기기정보 Table은 열교환기에 대한 주요사양 정보를 관리하는 Table로서 사용자에게 의해 입력받도록 한다. Map Style 정보에서는 Drawing 된 Map 정보를 가지고 있다. Zone 정보는 영역별로 분할된 Map 정보 및 Zone별 행/열 시작번호와 색상 및 라인과 같은 속성 정보 등을 가지고 있다. [그림2-5]에 DB 모델링 결과 생성된 개체관계도를 표시하였다.

[그림 2-5] 개체 관계도

3. 시스템 구현



```

for(i=1;i<=m_nRow;i++){
    for(j=1;j<=m_nCol;j++){
        if(m_nGrid==0 && i%2==0){
            if(j==m_nCol) continue;
            CreateTube(i,j,i*r,j*r+r/2,TRUE);
        }
        else if(m_nGrid==1 && i%2==1){
            if(j==m_nCol) continue;
            CreateTube(i,j,i*r,j*r+r/2,TRUE);
        }
        else CreateTube(i,j,i*r,j*r,TRUE);
    }
}
    
```

3.1 Tubesheet Map 정보 구조

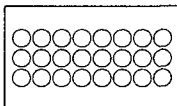
```

int m_Arow;           //절대 열 번호
int m_Acol;           //절대 행 번호
int m_Arowp;         //절대 화면위치
int m_Acolp;         //절대 화면위치
int m_row;           //상대 열 번호
int m_col;           //상대 행 번호
int m_Orgrow;        //Zone전 열 번호
int m_Orgcol;        //Zone전 행 번호
BOOL m_live;         //Tube Flag
BOOL m_select;       //Tube 선택
int m_zoneno;        //Zone 번호
int m_Status;        //검색정보 표현 Index
    
```

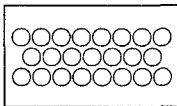
3.2 주요 Tubesheet Map 생성방법

열교환기 Tubesheet Map은 증기발생기와 일반적인 열교환기 형태로 구분할 수 있다. 즉 Map 형태의 행/열 번호 구조가 상이하서 서로 구분하여 구현하여야 한다.

○ 증기발생기 Map구조



○ 열교환기 Map구조

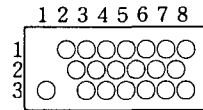


위 그림과 같이 열교환기의 경우 두 번째 열이 첫 번째 열보다 안쪽으로 구성되며 첫 번째 열에 비해 Tube 개수가 하나 적음을 알 수 있다. 따라서 형태 구분을 위해 m_nGrid로 구성하고 m_nGrid에 따라서 전체의 Map 구조를 만든다. 아래 모듈의 CreateTube 함수는 위의 Tubesheet Map구조를 생성하는 모듈이다.

3.3 Tubesheet Map 편집기능

3.3.1 열교환기와 증기발생기 구조이해

○ 열교환기 구조

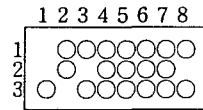


위 그림에서 좌우측 숫자는 절대 행/열 번호라고 할 때 편집과정에서 Tubesheet가 없는 정보는 열번호에 적용이 안된다. 따라서 위 구조의 행/열 번호는

- (1,1)(2,1)(3,1)...
 - (1,2)(2,2)(3,2)...
 - (1,3)(2,3)(3,3)...
- 형태로 입력된다.

즉 Tubesheet가 없는 행/열 번호는 적용하지 않는다.

○ 증기발생기 구조



증기발생기는 절대번지구조로 Mapping 된다. 따라서 위 구조의 행/열 번호는

- (2,1)(3,1)(4,1)...
- (2,2)(4,2)(5,2)...
- (1,3)(3,3)(4,3)...

즉 Tubesheet의 행/열 번호는 절대 행열값과 동일한 번호가 부여된다. 또한 열교환기는 증기발생기보다 약 2배정도 더 많은 수의 Tube를 가지고 있다. 따라서 많은 정보를 관리하기 위해 Zone이라는 구조를 약 8개 정도를 가지고 있어 각각 Zone 별로 행/열 값을 따로 관리해야 하는 문제가 있다. 따라서 편집후 Re-Ordering 기능 모듈에서 이 문제를 해결하도록 하였다. Re-Ordering 기능은 사용자에 의해 Zone 영역으로 분할 과정에서 수행된다.

3.4 편집후 행/열 재정렬

아래 모듈은 위에서 열거한 Zone 영역 분할후 Re-Ordering 하여 행/열 번호를 다시 부여하는 부분이다.

```

old = row=0,no=1;
for(j=0;j<m_DocTube->GetSize();j++){
    ptube=(CTube*)m_DocTube->GetAt(j);
    if(!ptube->m_live) continue;
    if(ptube->m_zoneno!=pzone->zone_no)
    continue;
    if(old!=ptube->m_Arow){
        if(pzone->m_ZOption%2!=0 && row!=0){
            for(k=os;k<m_nRow*m_nCol;k++){
                ptube2=(CTube*)m_DocTube->GetAt(k);
                if(ptube2==NULL) continue;
                if(old!=ptube2->m_Arow) break;
                if(!ptube2->m_live) continue;
                if(ptube2->m_zoneno!=pzone->zone_no)
                continue;
                ptube2->m_col = no-ptube2->m_col;
            }
        }
        if(pzone->m_ZOption<2) row = bottom-top-
        ptube->m_Arow+pzone->m_row_st_no+1;
        else row = ptube->m_Arow-top
        +pzone->m_row_st_no;
        old = ptube->m_Arow;
        no=pzone->m_col_st_no;
        os=j;
    }
    ptube->m_row = row;
    ptube->m_col = no;
    no++;
}
if(pzone->m_ZOption%2!=0 && row!=0){
    for(k=os;k<m_nRow*m_nCol;k++){
        ptube2=(CTube*)m_DocTube->GetAt(k);
        if(ptube2==NULL) continue;
        if(old!=ptube2->m_Arow) break;
        if(!ptube2->m_live) continue;
        if(ptube2->m_zoneno!=pzone->zone_no)
        continue;
        ptube2->m_col = no-ptube2->m_col;
    }
}
    
```

위의 모듈과 같이 열교환기의 행/열 번호는 각각의 Zone별 구조와 병행하여 행/열 번호를 재부여 할 수 있는 기능이 요구되어진다.

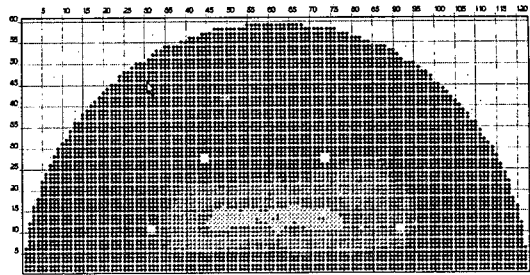
3.5 Tubesheet Map 저장 및 확인

이상과 같은 Drawing 과정을 거쳐 Map이 완성되면 저장기능을 이용해 Map을 데이터베이스에 저장하고, 실제 검사정보 데이터베이스에서 Mapping이 되는가를 확인한다. 검색된 정보의 Row/Column

Number와 저장된 Tubesheet Map의 행/열 번호가 정확히 일치해야 아래 [그림3-1]과 같은 Mapping이 이루어진다.

[그림 3-1] Mapping 결과

4. 결론 및 향후 연구과제



본 논문에서는 현재 추진중인 원자력발전소의 주요 설비인 열교환기 및 증기발생기의 검사정보 관리를 위한 “열교환기 검사정보 관리시스템” 개발에 주요 기능중의 하나인 Tubesheet Map 자동 Drawing 시스템 구현 방법에 대해 설명하였다. 이 검사정보 관리시스템은 현재까지 수행된 전 검사정보를 데이터베이스로 구축하고, 현업 담당자들에게 다양한 형태의 Tube 현황이력 및 통계분석 정보를 제공하기 위한 전산시스템 개발을 목표로 현재 진행중에 있다. Tubesheet Map 자동 Drawing 시스템은 발전소 별로 300~400여개에 달하는 다양한 종류의 열교환기 Tubesheet Map을 컴퓨터를 이용하여 쉽게 작성할 수 있도록 도와주기 위한 시스템이다. 이렇게 Drawing된 Tubesheet Map상에 각 Tube의 결합, Dent, 슬러지 등과 같은 검사정보를 데이터베이스 읽어 행/열 번호에 따라 Mapping하여 준다. 이 시스템을 활용하여 현업부서 담당자들은 Tube 이력정보를 시각적으로 파악할 수 있을 것으로 판단되고, 방대한 분량의 검사정보를 체계적으로 관리할 수 있을 것으로 판단된다. 향후 이 시스템은 열교환기 및 증기발생기 사양/특성관리 시스템, 검사결과 자료관리 시스템, 통계/보고서 처리 프로그램과 함께 개발되어 현장에 직접 활용될 예정이다.

참고문헌

- [1] 한전 원자력연수원, “가동중검사”, 1997
- [2] 한전, 원자력연구소, “고리 4호기 제12차 가동 검사 보고서”, 2000. 3
- [3] 이상엽, “Visual C++ Programming Bibl 4.x”, 영진출판사, 1997. 2
- [4] 대한전기학회, “’99 하계학술대회” 논문지 컴퓨터 및 인공지능, CAP 18, 1999. 7
- [5] 양수정, “오라클 SQL과 ProC”, 해지원, 1998. 7