

다차원 최적열수력계통분석코드 MARS 1.3과 3차원 노심 핵설계 코드
MASTER 2.0을 통합한 MARS/MASTER 코드의 시험버전 개발

Development of a Draft Version of MARS/MASTER; A Coupled Code of
MARS 1.3 and MASTER 2.0

정재준, 주한규, 정법동, 하귀석, 이원재, 조병오, 지성균

한국 원자력 연구소
대전시 유성구 덕진동 150

요약

다차원 최적 열수력계통 분석코드 MARS 1.3과 3차원 노심 핵설계 코드 MASTER 2.0의 3차원 동특성 모듈을 통합한 MARS/MASTER 코드의 시험버전을 개발했다. PC Windows 운영체제에서 작동하는 통합코드를 효율적으로 개발하기 위해, Linkmaster라는 연계모듈을 만들어 두 코드로 하여금 공유하도록 하고 이를 통해서 두 코드간의 정보전달이 이루어지게 했으며, MASTER 코드의 3차원 동특성 모듈은 MARS에서 필요시에 호출할 수 있도록 수정한 다음, 동적 연계 라이브러리(Dynamic Link Library) 형태로 변환했다. 이렇게 함으로써 두 코드의 독립적인 유지 및 개선이 가능하도록 했다.

간단한 예시 계산을 통해 MARS/MASTER 코드 통합의 건진성 및 실효성을 확인했고, 실제 발전소에 적용할 경우의 계산속도도 현실적으로 사용이 가능한 범위에 있는 것으로 평가되었다.

Abstract

The draft version of MARS/MASTER code, a coupled code of MARS 1.3 and MASTER 2.0, has been developed. To efficiently couple the two codes on a personal computer with Windows system, an interface module named "linkmaster.f90" is programmed, through which the data communication between MARS 1.3 and MASTER 2.0 can be established, and the three-dimensional kinetics module of MASTER 2.0 in type of dynamic link library is modified so that it can be called by MARS.

From the results of a simplified test calculation, the validity of the coupling algorithm and the feasibility are confirmed. In addition, it is found that the computational speed of MARS/MASTER for a realistic plant application is tolerable.

1. 서론

근래 들어 열수력 계통 분석 코드와 3차원 중성자 동력학(Neutron Kinetics, 이후 '중성자'는 생략) 코드를 통합하여 보다 정확한 원전 과도 해석 코드를 구축하려는 연구가 세계적으로 활발히 진행되고 있다. 과거에는 두 코드의 통합이 각각의 코드에서 요구되는 막대한 계산량 때문에 비현실적이었으나, 이제는 급속한 전산기 능력의 향상에 힘입어 점차 현실화되고 있기 때문이다.

한국 원자력 연구소에서는 최적열수력계통 분석코드 개발의 일환으로 다차원열수력계통 분석 코드인 MARS 1.3[1]을 3차원 핵설계 코드 MASTER 2.0[2]의 3차원 동특성 모듈과 통합함으로써 각 코드의 장점을 최대한 활용한 새로운 통합코드체계 개발을 추진하고 있다[3]. MARS 1.3은 RELAP5와 COBRA-TF를 통합한 COBRA/RELAP5 코드를 근간으로 하여 코드 구조조정 및 현대화를 통해 고유의 코드로 새롭게 개발한 다차원 최적 열수력 계통분석 코드이다. 이 코드는 이상유체모델, 열전도체 모델, 점동특성 모델, 기타 밸브, 펌프 등의 가압경수형 원전의 계통과도분석에 필요한 모델을 거의 모두 가지고 있다. 특히 COBRA-TF를 다차원 이상열유동 모듈 형태로 통합함으로써 원자로 압력용기 등에서 발생하는 주요 다차원 이상열유동 현상을 최적 모의할 수 있는 장점이 있다. 또한, 코드 구조조정 및 현대화를 통해, 코드의 해독성, 보수유지성, 이식성 등을 크게 향상시켰으며, Windows 운영체제에서 Windows Programming의 적용을 통해 사용자 편의성을 대폭 향상시킨 코드이다.

현재 MARS 1.3의 노심 동특성 모델로 점동특성모델(Point Kinetics Model)이 사용되고 있다. 따라서, 운전 혹은 사고진행 도중에 노심의 출력분포가 변화하는 경우를 모델 하는 데는 한계가 있다. 본

연구에서는 이러한 한계성을 개선하기 위해 기존의 점동특성모델을 MASTER의 3차원 동특성 모듈로 대체/통합하고자 한다. MASTER 코드는 노심 핵설계 분석에 사용되는 코드로서 광범위한 계산 기능을 지니고 있다. 대표적인 기능은 정상 상태에 대한 노심 고유치 계산, 초단시간 과도 상태 해석, 핵중별 연소 계산, 지능 과도 상태 해석 등을 들 수 있다.

MARS와 MASTER 코드의 통합을 통해 노심의 다차원 동특성 해석이 중요한 계통과도현상, 혹은 계통과도현상이 노심 동특성에 중요한 영향을 미치는 사고 등에 대한 최적모의가 가능함으로써 기존 계산방식의 비효율성, 부정확성 및 보수성을 일거에 제거하는 장점을 얻게 된다.

본 논문에서는 MARS/MASTER 통합코드개발의 첫번째 단계 결과물인 시험버전의 통합방법론을 기술하고, 예시계산의 결과를 평가하며, 이를 토대로 MARS/MASTER의 완성도를 높이기 위해 필요한 향후 연구항목을 기술하였다[3].

2. 코드통합방법

FORTRAN으로 작성된 독립적인 두 코드를 통합할 경우, 한 코드를 다른 코드의 서브루틴화하여 단일 코드로 만드는 것이 일반적이다. 이 경우 두 코드내의 부 프로그램 이름이나 Common Block 이름이 중복되지 않아야 한다. 이러한 중복을 피하기 위해서는 두 코드 모두에 많은 양의 수정이 필요하고, 추후 각 코드의 지속적인 개선이 독립적으로 이루어지지 못한다는 단점이 있다. 이러한 단점을 피할 수 있는 코드 통합 방법은 별도의 연계코드를 작성하는 방법과 동적 연계 라이브러리 (Dynamic Link Library, DLL)을 사용하는 방법[4, 5]이 있다. 전자는 UNIX 운영체제에서 프로세스 제어를 통해 가능하고, 후자는 Windows 운영체제에서 객체 지향적 (Object Oriented) 계산 수행을 통해 가능하다.

DLL을 사용할 경우, 두 코드중 하나를 주 코드와는 독립적으로 작동하는 DLL 형태로 생산하는데, 통합된 코드가 실행될 때 DLL내의 코드가 사용하는 기억영역과 주 코드가 사용하는 기억영역은 별도로 분리되기 때문에 기억영역의 상충이 원천적으로 배제된다. 두 코드간의 자료교환을 위해 제한적인 기억영역의 공유가 허용되는데, 이는 DLL내에 Entry Point로 지정되어 있는 서브루틴의 매개인자 (Arguments)들을 통해 가능하다. 이 같은 장점으로 인해, DLL을 사용하는 코드통합은 별도의 연계 코드를 작성하는 것에 비해 작업량이 적고 전체 알고리즘도 단순해진다. 따라서 MARS/MASTER 통합코드의 개발에는 DLL 방법을 적용했다.

MASTER의 3차원 동특성 모듈을 DLL 형태로 변환하고, DLL내에 Entry Point로 지정된 서브루틴의 매개인자를 연계모듈(Interface Module)로 만들어 주 코드 MARS와 공유하도록 하고 이를 통해서 두 코드간의 정보전달이 이루어지게 하며, MARS 코드는 필요시에 부 코드, 즉, MASTER DLL을 호출하도록 수정함으로써 두 코드의 통합을 자연스럽게 이루었다. 이 통합 방식의 가장 큰 장점은 두 코드의 독립적인 유지 및 개선이 가능하다는 점이다.

3. MARS/MASTER 통합코드 개발

통합코드 개발을 위해서 MARS의 동특성 관련 부프로그램을 수정하고, MASTER의 3차원 동특성 모듈을 수정후 DLL로 변환해야 하며, 매개인자를 정확하게 정의해야 한다. 본 절에서는 위의 각 단계에 대해 간략히 설명한다.

3.1 MARS 1.3의 통합관련 수정

두 코드의 통합에는 기존의 MARS 코드 입력자료 사용시 통합으로 인한 영향이 나타나지 않아야 하며, 두 코드의 계산은 상호 독립적으로 수행되면서도 동기화(Synchronization)되어야 한다는 점이 중요하다. 또한, MARS의 열수력 모델(Hydrodynamic Model) 및 열전도체 모델 (Heat Structure Model)의 계산결과가 MASTER의 열케환 입력으로 전달/이용되며, MASTER의 계산결과, 즉, 노심 총출력 및 출력 분포는 MARS의 열전도체 모델에서 열원(Heat Source)으로 사용되도록 해야 한다. 그림 1은 MARS 코드내에서 각 모델간의 정보전달 내용을 보여준다.

MARS 1.3은 과도현상 모의시 매 시구간 마다 열전도체 모델, 열수력 모델, 노심 동특성 모델의 순으로 계산한다. 이와 같은 순서는 상호반복계산이 없다는 전제하에서 일반적인 시간상수가 긴 쪽을 먼저 풀고 짧은 쪽을 나중에 풀어서 계산의 정확성을 상대적으로 증대시키기 위한 것이다.

MARS/MASTER 코드개발과 관련된 MARS 코드의 수정은 기존 점동특성 모델을 MASTER의 3차원 동특성 모듈로 대체하는 관점에서 이루어졌다. 즉, MARS에서 점동특성모델 계산을 위해 필요한 초기화, 열수력 및 열전도체 모델의 정보전달, 점동특성 모델의 계산, 계산결과의 전달 등 각 단계를 MASTER의 3차원 동특성 모듈로 대체하도록 MARS 코드를 수정했다.

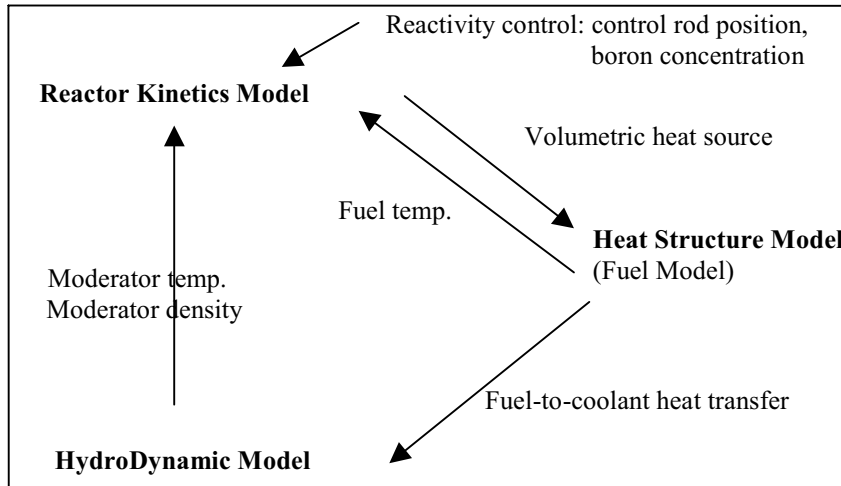


그림 1. 열수력 모델, 열전도체 모델 및 노심동특성 모델간의 정보전달

3.2 MASTER 코드 수정 및 DLL 생산

MASTER 코드에는 열수력 모듈로 부수로 해석코드인 COBRA-IIIC/P가 내장되어 있으며, 핵연료의 열전도체 모델도 들어있다. 통합코드에서는 이 부분들을 MARS로 모의하므로 기존 MASTER에서 제외하고 나머지 모든 MASTER 서브루틴을 DLL 형태로 생산했다. 다만, 현재 DLL 형태의 통합은 Windows 운영체제에서만 생산과 사용이 가능하고, 기존의 MASTER는 UNIX 체계의 워크스테이션에서 사용되어 왔기 때문에, MASTER를 Windows 운영체제용으로 전환하기 위한 다소의 수정은 불가피했다. 이 과정에서 특정 전산기에 대한 의존성을 탈피하기 위해 프로그램 전체를 표준 FORTRAN90으로 변환했다.

통합코드를 사용한 과도 해석에서 계산 수행 과정은 크게 세 단계로 분류할 수 있다. 첫 번째가 초기화 단계로 이 때는 입력의 처리 및 변수 초기화가 이루어진다. 두 번째는 정상 상태 계산 단계로 과도 계산의 초기조건을 결정하는 초기 정상상태의 형성을 위한 계산이 진행된다. 마지막 세 번째는 각 시구간마다 한 번씩 반복되어 진행되는 과도계산 상태이다.

MASTER코드가 독립적으로 수행될 때는 MASTER의 주 프로그램이 세 계산 단계를 일관되게 제어하지만, MASTER가 MARS에 종속되어 수행될 때는 MARS 코드가 각 계산 단계를 통제하여야 한다. 이를 위해 MASTER의 주 프로그램을 서브루틴으로 전환했고, 서브루틴 매개인자를 통해 세 단계 중 어느 단계에 속하는지 판단하고 단계별로 적절히 MASTER내 해당 서브루틴을 호출하는 방식으로 계산이 수행될 수 있도록 일부 서브루틴을 수정했다. 그림 2는 수정된 MASTER DLL의 연계계산 유통도를 나타낸다.

3.3 매개인자 정의 및 세부계산절차

세부 연계 방식을 정립하는 과정에는 매개인자의 선정, 통상적으로 서로 다른 구조를 가질 수 있는 3차원 노심 동특성 계산과 열수력 계산 격자간의 대응방식 설정, 연계계산 진행 논리구성 등이 포함된다.

3.3.1 매개인자

열 교환 효과를 결정하는 열수력장 변수들은 핵연료 온도, 냉각재 온도 및 밀도이고, 이들 열수력장 변수들은 열수력 계산의 결과로 동특성 계산에 공급되어야 한다. 동특성 계산은 각 격자별로 주어진 열수력장 조건에 따라 균정수를 계산하는 것으로 출발하여 출력분포를 구하는 것으로 끝난다. 동특성 계산에 의해 얻어진 출력분포는 열수력 계산 코드로 전달되어 다음 단계의 열수력 계산에 사용된다.

열 교환 처리 목적 이외에 MASTER 계산조건을 지정하기 위한 연계변수로 봉산농도와 각 제어봉군의 삽입위치가 필요하다. 봉산 농도의 경우, 계통에서 유입되는 냉각재내 봉산농도가 노심내 위치에 따라 다를 수 있으므로 MARS에 의해 그 값이 지정되어야 한다. 제어봉군의 위치도 마찬가지로, MARS에서 출력감발 제어나 운전정지 신호 등에 의해 결정되므로 MARS에 의해 MASTER에 제공되어야 한다.

이와 같이 MARS에서 MASTER로, 혹은 그 반대로 전달되는 모든 변수를 매개인자로 정의하고 이들을 모아서 하나의 연계모듈로 만들어 통합관리하도록 했다.

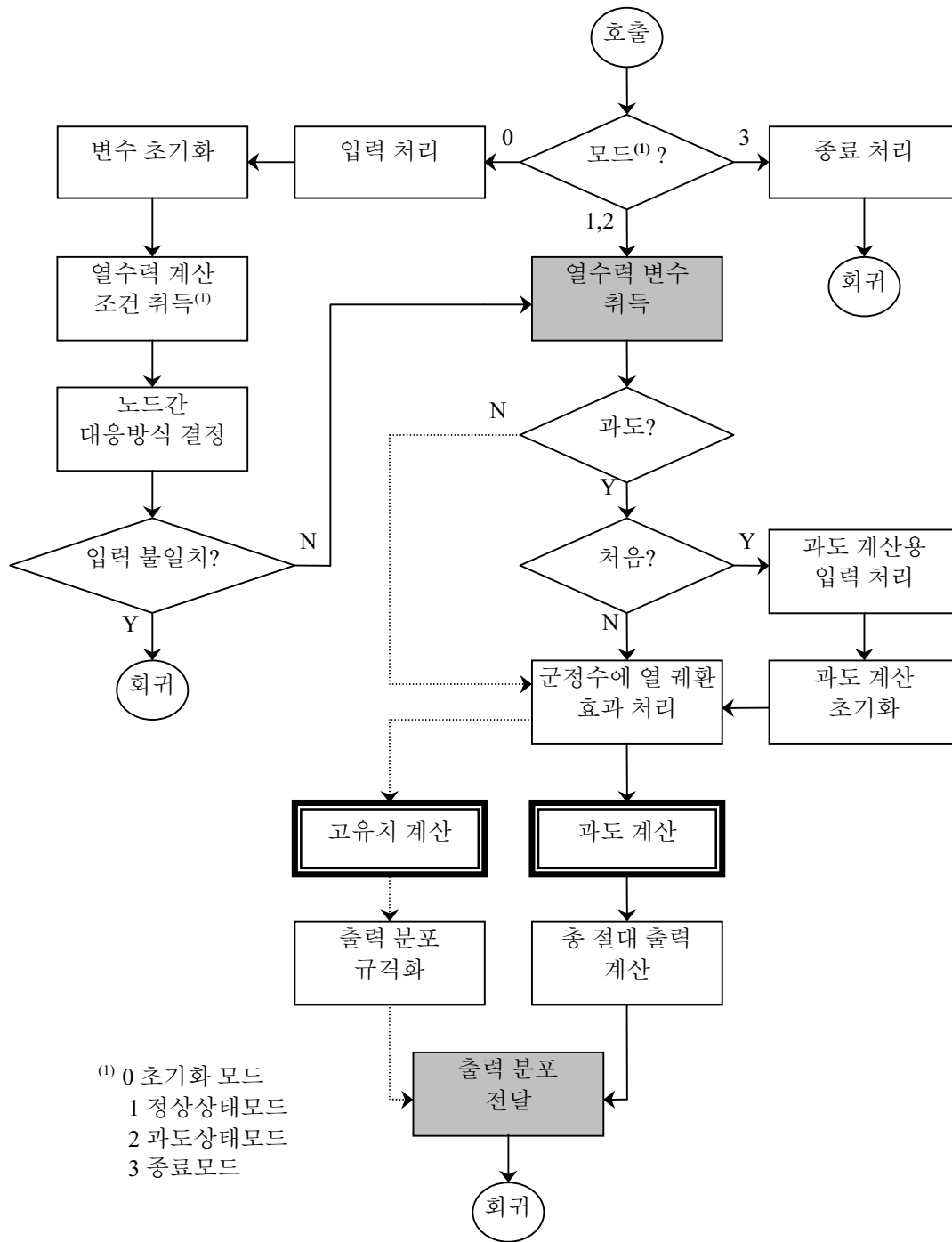


그림 2. 수정된 MASTER DLL의 연계 계산 유통도

3.3.2 동특성 계산 및 열수력 계산 격자간 대응 방식

일반적으로 노심 열수력 계산에 사용하는 격자의 크기는 동특성 계산용 격자에 비해 크다. 이것은 중성자의 수송에 기인한 격자간 연계성이 냉각재 유동에 의한 연계성 보다 강하기 때문이다. 두 계산 모듈간에 격자 구조가 다를 경우에 연계 변수를 상호 교환하기 위해서는 각 격자간 대응 관계를 설정해야 한다. 열수력 격자의 경계면과 동특성 계산 격자의 경계면도 일반적으로 일치하지 않을 수 있으나, 대응관계 설정에 있어서 불필요한 복잡성을 피하기 위해 통합코드 사용시 두 격자 구조내 경계면 만은 서로 일치하도록 제한을 두는 것이 바람직하다.

본 연구는 타당성 평가차원에서 수행된 것이므로, 서로 다른 격자간의 일반적인 Data Mapping 문제는 우회하고, 미리 정해진 격자에 대해서만 Data Mapping이 가능하도록 했다 (제 4절의 예시계산 참조).

3.3.3 동특성 계산과 열수력 계산 교차 수행 방식

과도상태에 대한 동특성 계산과 열수력 계산에 있어서 각각에 적합한 시구간 크기는 서로 다를 수 있다. 그러나 여기서는 최적화된 시구간 조절 방식은 차후 개발과정에 고려한다는 전제하에 동특성 계산과 열수력 계산간의 시구간 크기의 차이 가능성을 일단 배제하고 동일한 시구간 크기를 사용하여 통합코드 계산이 진행되게 하였다. 따라서 과도계산시 매 시구간 마다 열수력 계산과 동특성 계산이 교차로 수행된다. 원칙적으로 동특성 계산과 열수력 계산의 밀접한 연관성으로 인해 한 시구간 내에서도 동특성 계산과 열수력 계산의 교차수행을 반복할 필요가 있다. 이러한 필요성은 시구간 크기가 클수록 더 커진다. 그러나 개발 초기에는 계산의 편의성을 고려하여 반복 수행을 배제했다.

3.3.4 연계계산 진행논리

과도상태 계산에는 초기 정상상태에 대한 계산이 선행되어야 한다. 동특성 계산에 있어서 정상상태와 과도상태에 대한 계산 방법은 고유치 계산과 고정선원 계산으로 확연히 구별된다. 따라서 연계 계산시 가장 먼저 지정해야 할 것은 MASTER의 계산 모드이다. 정상상태 계산 모드와 과도 상태 계산 모드 이외에 MASTER.DLL에는 MASTER 자체의 입력처리 및 변수지정을 위한 초기화 모드가 있다. MASTER 계산모드는 MARS가 MASTER의 주 루틴을 호출할 때 전달되어야 하고 MASTER 주 루틴은 계산 모드에 따라 적절히 계산의 흐름을 제어하여야 한다.

위와 같은 고려 사항을 토대로 MASTER.DLL은 그림 2에 도시한 계산 진행 논리를 구현하도록 생산되었다. 이 그림에서 회색으로 칠해진 부분은 매개인자를 교환하기 위한 것이고 점선으로 표시된 경로는 고유치 계산을 위한 것이다. 이 계산 유통은 MARS가 매 시구간 마다 계산모드를 지정한 후 MASTER 주 루틴을 한번 호출함으로써 시작되고 MASTER가 한 시구간에 대한 계산을 완료하고 MARS로 회귀함으로써 종료된다.

4. 예시계산 및 결과

MARS/MARS 통합코드의 시험 버전에 대한 논리의 건전성 및 실효성을 보기위해 간단한 예시계산 문제를 만들었다. 여기에서 사용된 노심의 구성은 다음과 같다.

- 총 핵연료 봉 수: 총 46,990개
- 핵연료봉 재원: LOFT 핵연료봉 사용
- 핵연료봉길이: 3.6 m
- 노심유면적: 4 m²
- 노심출력: 2775 MW
- 압력: 159.27 MPa
- 정격유량: 15,000 kg/s
- 입구온도: 571.2 K
- 기타 제원: 고리 3/4호기 자료이용

상기 노심을 MASTER의 노심 동특성 계산에서는 그림 3에 나타난 바와 같이 1/4 노심에 대해 64(radial) x 18 (axial) 격자를 사용했다. MARS에서는 열수력 관점에서 노심전체를 1차원의 12개 격자로 나누되 다차원열수력 모듈로 모의했다. 노심 입출구의 공동을 모의하기 위해 상하부에 각각 6개의 열수력 격자로 구성된 파이프를 설치했다 (그림 3에는 표시되지 않음). MARS의 핵연료의 열전도 모델도 열수력 모델과 동일하게 높이 방향의 격자 12개를 이용했으며, 핵연료핀은 원주 방향으로 4개 격자로 나누었다. 서로 다른 격자간의 Data Mapping은 격자체적을 기준으로 선형 근사했다.

예시계산의 시나리오는 다음과 같다. 초기유량을 영(Zero)으로 두고 0초에서 50초 사이에 유량을 선형적으로 정격유량까지 증가시킨 다음 정격유량을 유지한다. 이와 유사하게 0초에서 50초까지

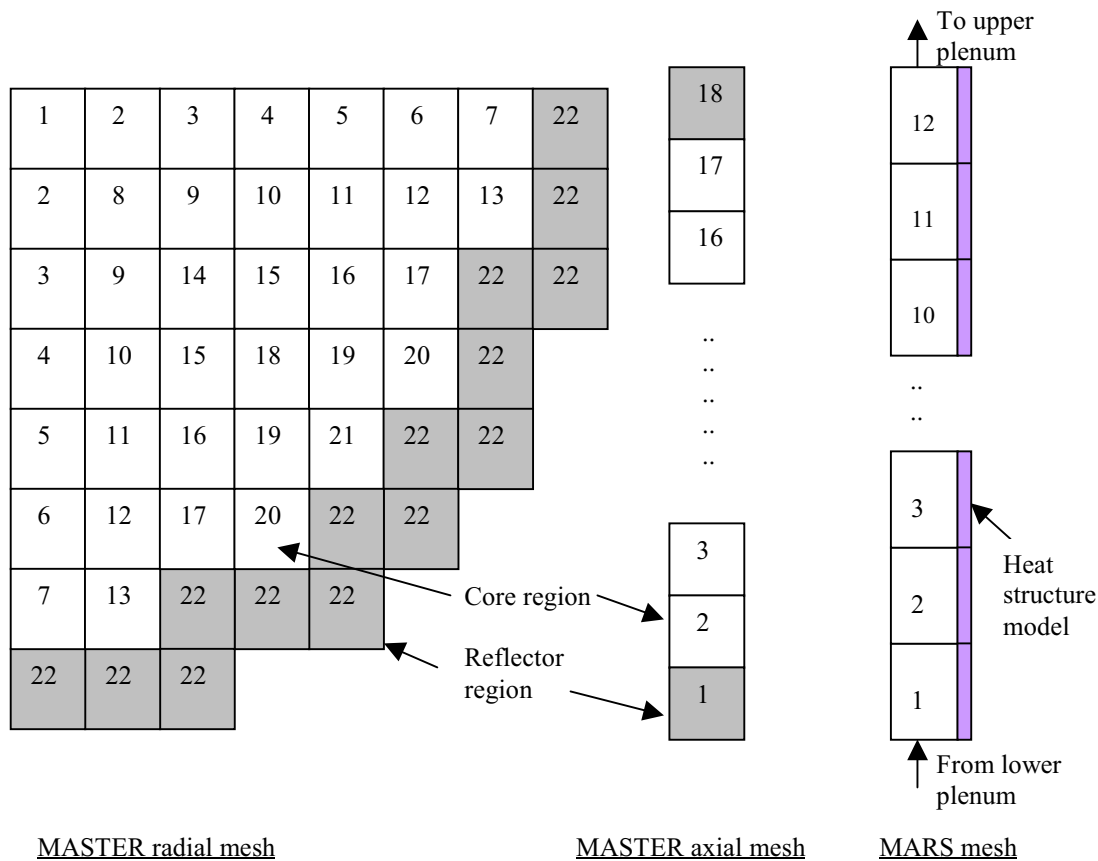


그림 3. 예시계산에 사용된 MASTER와 MARS 코드의 노심입력모델:
MARS는 1차원, MASTER 는 3차원으로 계산함.

노심출력은 영으로 두었다가, 50초에서 100초까지 노심출력을 선형적으로 정격출력 까지 증가시킨 다음 정격출력을 끝까지 유지한다.

이 계산을 기존의 MARS 코드로 우선 수행했다. 기존 MARS 코드에서는 노심 총출력 및 출력분포를 테이블 형태로 입력했으며 이때 출력분포는 편의상 노심전체에 걸쳐 균일하다고 가정했다. MARS/MASTER 계산에는 앞 절에서 설명한 바와 같이 정상상태 및 과도상태 계산양식이 있는데, 본 계산에서는 정상상태 양식을 사용했다 (과도상태계산을 위해서는 정상상태계산의 결과로부터 MARS와 MASTER가 함께 Restart 계산을 해야 하는데, 현재 통합코드의 Restart 기능이 완성되지 않았다). 정상상태 계산시에는 총출력이 주어지면 MASTER가 출력분포를 계산해서 MARS로 넘겨준다. 두 계산의 결과는 그림 4와 5에 각각 나와있다. 두 계산의 출력파일 및 그림 4와 5의 과도 현상 비교를 통해 MARS/MASTER 통합은 정성적, 정량적 관점에서 성공적이라 할 수 있었다. 단, 그림 4와 5의 우측 중간에 표시된 노심중간부위의 엔탈피 거동 차이는 노심출력분포를 MARS에서는 Uniform shape을 썼으나 MARS/MASTER에서는 Bottom Skewed shape으로 계산하였기 때문이다.

개인용 컴퓨터(300 MHz)에서 상기 예시계산 200초를 모의하는데 MARS는 544초를, MARS/MASTER는 1965초를 소요했다. 이와 같은 큰 차이는 MARS 계산의 열수력 격자 수가 24개에 불과한 반면, MARS/MASTER 계산에는 1152(64x18)개의 노심동특성 계산용 격자가 추가로 사용되었기 때문이다. 두 계산 모두 시구간의 크기는 0.01초로 거의 고정되었다. 이 계산을 근거로 각 코드의 Grind time을 산출하면 대략 다음과 같다:

MARS: 1.14 ms/hydrodynamic mesh/time step

MASTER: 0.0823 ms/kinetics mesh/time step

이를 바탕으로 실제 원자로계통 적용시의 계산시간을 추론해 볼 수 있다. 열수력 격자수를 약 500개, 노심동특성 격자를 약 2826(157x18)개로 보면, 한 시구간 계산에 요구되는 시간은 대략 다음과 같다:

$$1.14 * 500 + 2826 * 0.0823 = 570 + 232.6 \text{ (ms)}$$

즉, 계통 계산에서 MASTER의 계산시간이 약 29%에 이른다고 볼 수 있다. 이와 같은 방식으로 여러 가지 격자수를 조합해보면 MASTER 계산시간의 비율을 표 1과 같이 추정할 수 있다.

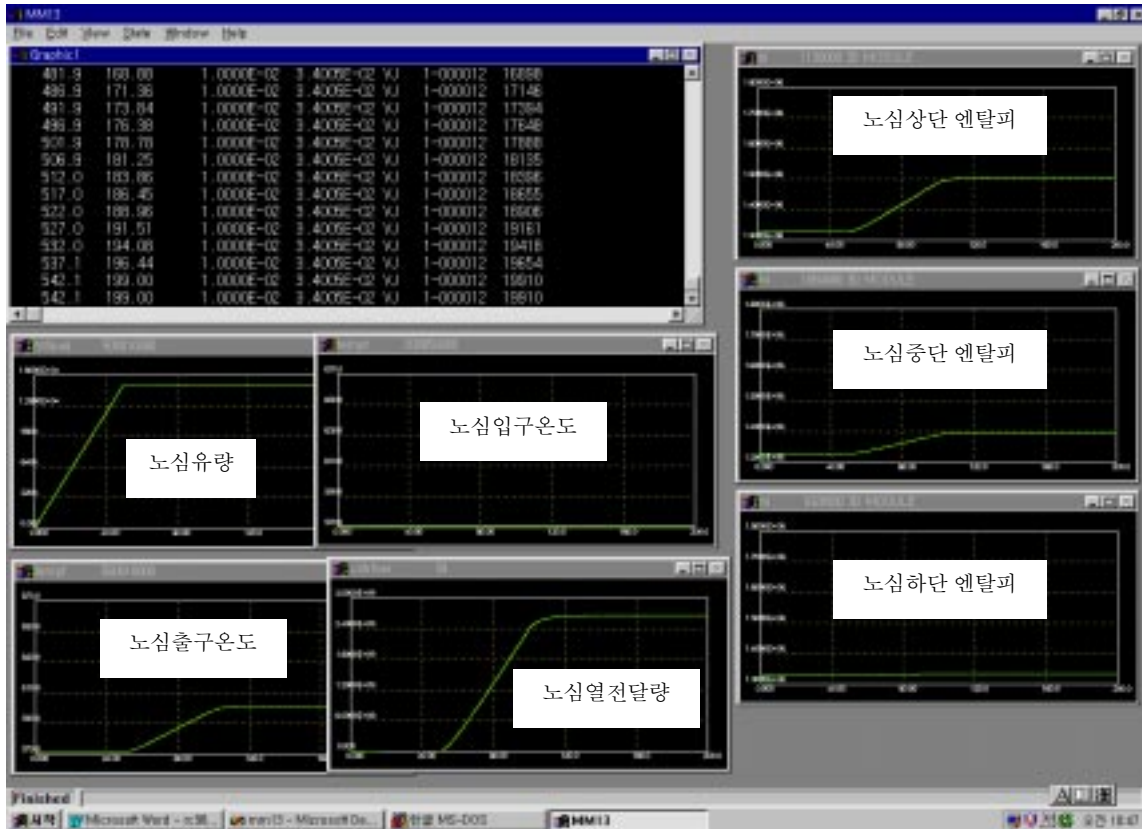


그림 4. MARS 계산결과: 좌측중단부터 노심유량, 노심출구냉각수온도, 노심입구냉각수온도, 노심열전달량, 노심 상,중,하단의 냉각수 엔탈피 거동 (0-200초)

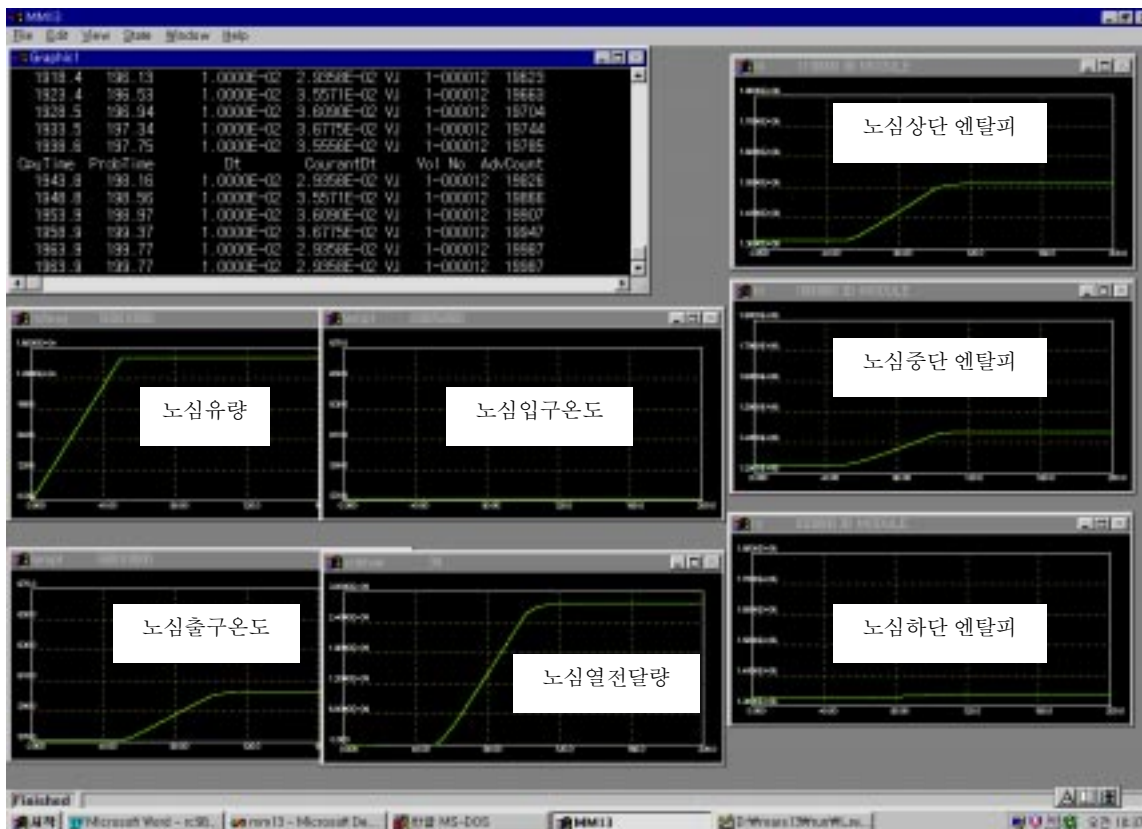


그림 5. MARS/MASTER 계산결과: 그림배열은 위와 동일함.

표 1. MARS/MASTER의 Mesh수에 따른 MASTER 계산시간 비 추정치

열수력 격자	노심동특성 격자	MASTER 계산시간 비 추정치
500	157 x 18	29.0 %
1000	157 x 18	16.9 %
500	230 x 18	37.4 %
1000	230 x 18	23.0 %
2000	230 x 18	13.0 %

실제 계산에서 열수력 격자수가 증가할수록 MARS의 Grind time은 더 크게 증가하기 때문에 표 1의 MASTER 계산시간 비 추정치는 다소 감소할 것으로 보인다. 현재 MARS 계산 그 자체만으로도 계산시간이 큰 편이기 때문에 MARS/MASTER 사용시의 최적의 격자 사용이 필요할 것으로 평가되지만, 개인용 컴퓨터에서도 현실적인 계산이 가능하다는 점은 분명하다.

그런데, 상기 계산은 정상상태 계산 모드로 계산된 것이다. 과도상태 계산모드를 시험하기 위해, 상기 계산을 재현하면서 디버거 상에서 계산 모드를 인위적으로 바꾸고 그 결과를 분석해보았다. 이 경우에도 통합논리의 건전성과 결과의 타당성이 확인되었는데, 계산시간은 다소(수 퍼센트) 커지는 것으로 나타났다.

5. 결론 및 향후 연구항목

다차원 최적계통분석코드 MARS 1.3과 3차원 노심핵설계코드 MASTER 2.0의 3차원 동특성 모듈을 통합한 MARS/MASTER 코드의 시험 버전을 개발하고 평가했다. MARS와 MASTER 코드의 통합시, MASTER 모듈은 MARS가 필요시에 호출할 수 있도록 수정한 다음, 동적연계 라이브러리 형태로 변환하여 활용함으로써 두 코드의 독립적 개발 및 개선이 가능하도록 하였다.

간단한 예시계산을 통해 MARS/MASTER 코드 통합의 건전성 및 실효성을 확인했고, 실제 발전소에 적용할 경우의 계산속도도 현실적으로 사용이 가능한 범위 내에 있는 것으로 평가되었다. 그러나 계산시간을 더욱 줄이기 위해 최적의 격자조합에 대한 연구가 필요한 것으로 보인다. 향후 이 코드의 완성도를 높이기 위해 수행되어야 할 연구항목은 다음과 같다:

- (1) MARS 격자와 MASTER 격자간의 대응관계를 일반화한 Data Mapping Scheme 개발.
- (2) MASTER의 과도 계산에 대한 Restart 기능 추가.
- (3) 효율적인 통합 코드의 실행을 위해 최적의 시구간 크기를 독자적으로 선택할 수 있는 기능 개발.
- (4) 잘 알려진 Benchmark problems을 이용한 통합코드 검증.

감사의 글

본 연구는 과학기술부의 원자력중장기개발계획의 일환으로 수행된 것이다.

참고 문헌

1. 이원재 외, "Development of a Multi-Dimensional Realistic Thermal-Hydraulic System Analysis Code, MARS 1.3 and its Verification," KAERI/TR-1108/98, 한국원자력연구소 (1998).
2. B. O. Cho et al., "MASTER 2.0 Methodology Manual," KAERI/TR to be published, KAERI (1998).
3. 정재준, 주한규, "MARS 1.3과 MASTER 3차원 동특성 모듈을 통합한 MAR/MASTER 코드의 Draft Version 개발 및 타당성평가", Internal Report, RTHS-M-98003, 한국원자력연구소 (1998).
4. Digital Fortran Language Reference Manual, Digital Equipment Cooperation (1997).
5. 정법동, "Code Coupling using a Dynamic Link Library," 열수력안전연구팀 내부 Seminar 자료, 1998. 8.
12. 발표, 한국원자력연구소 (1998).