

실시간 시뮬레이터와 연계된 3차원 가시화 프로그램 개발

이지우*, 이명수, 서인용, 홍진혁, 이승호, 서정관

Development of 3D Visualization Program Connected with Real-time Simulator

Ji-woo Lee, Myeong-soo Lee, In-yong Seo, Jin-huck Hong, Seung-Ho Lee, Jeong-Kwan Suh

Abstract

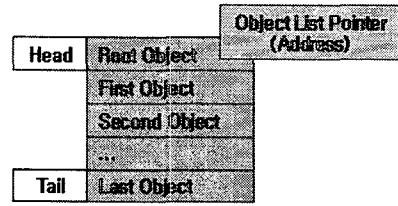
Each 3D visualization program has its own different structure as for the purpose. This paper describes the design and development of an on-line 3D core data visualization program, RocDis™, for the nuclear simulator. It is possible to analyze the inside of the core status including neutron flux, relative power, moderator and fuel temperature in 3D distribution. Some of other essential information, axial flux distribution etc. could also display in 2D graphs. This program would be design, tuning and training for the simulator core model.

Key Words: 3D Visualization, Simulator, Core Kinetics

* Korea Electric Power Research Institute.

1. 서론

3차원 가시화 프로그램은 목적에 따라 서로 다른 구조를 가지게 된다. 본 논문에서 다루고 하자 하는 가시화의 목적은 서버측(시뮬레이터)에서 시뮬레이션 모델이 기동되어 중요한 변수의 값을 통신 및 로컬 메모리를 통하여 3차원 가시화 프로그램과 연계되어 동적인 가시화를 보여주는 것이다. 사용자에게는 3D GUI(Graphic User Interface)를 제공 할뿐만 아니라 3차원 객체의 변화에 빠른 이해를 도와 주게 된다. 서버측의 시뮬레이터와 클라이언트 측의 가시화 프로그램을 연계하기 위해서 TCP/IP통신을 이용하였으며, 통신을 통하여 전달된 데이터는 3차원 가시화를 위한 객체의 속성과 연결된다. 데이터의 값이 변화함에 따라 3차원 객체의 속성이 실시간으로 변화함으로써 사용자에게 원하는 정보를 보여준다. 실제 시뮬레이터(영광1호기) 서버측에서 계산된 노심 일부의 데이터인 3차원 중성자속 및 핵연료 온도에 대하여 개발된 가시화 프로그램(RocDis™ : Realtime On-line Core Display) 도구를 이용하여 연계하여 보았다. 연계된 클라이언트측 RocDis™를 통하여 서버측 시뮬레이터 모델의 상태를 확인할 수 있었다.

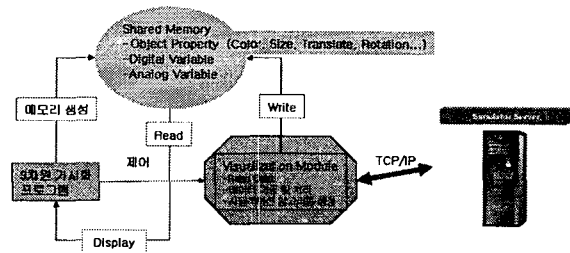


[그림 1]. 리스트 구조

생성된 객체는 리스트의 구조로 관리되며, 생성된 기본요소의 객체는 여러 가지 속성을 가지게 된다. 기본적으로 객체의 색(Red, Green, Blue, Alpha), 객체의 사이즈(Width, Height, Depth), 좌표 이동(Translate X, Y, Z), 좌표 회전(Rotation X, Y, Z)등의 요소를 가지게 된다. 이러한 요소는 동적인 표현을 위한 3차원 가시화에 사용되는데, 데이터 값의 변화에 따라 특정 객체의 색에 대한 변화를 주고자 할 경우에는 색 속성과 연결함으로써 동적인 표현이 가능하게 되는 것이며, 동적인 움직임을 주고자 할 경우에는 객체의 좌표 이동 및 회전 변환을 줌으로서 다양한 표현이 가능하도록 하였다. 동적인 표현을 위한 속성들은 데이터베이스로 관리되며, 공유메모리에 전역변수로 할당함으로써, 모든 변수의 값을 공유메모리로부터 읽고 쓰기가 가능하도록 하였다.

2. 3차원 가시화 프로그램 구현

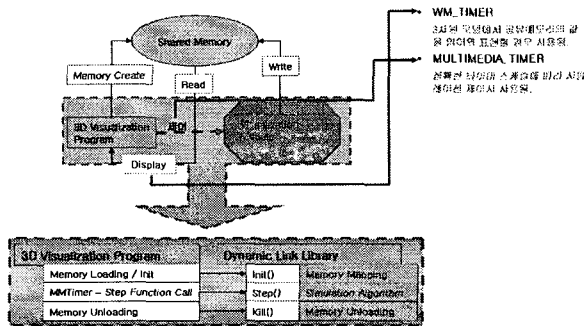
3차원 시뮬레이션을 위해서는 3차원 라이브러리에 대한 습득이 필요하다. 하지만, 사용자에게 객체의 속성 적용만으로 가시화가 가능하도록 하였다. 사용자의 마우스 및 키보드의 이벤트에 따라 3차원 기본객체의 동적생성이 가능하도록 하여 객체의 추가 그리고 제거 및 수정이 가능하도록 에디터 기능을 추가하였다.



[그림 2]. 가시화 프로그램의 구조

3차원 가시화 프로그램은 [그림 2]와 같이 Display하기 위한 타이머와 Visualization Module을 제어하는 타이머 두개를 동작시키게 된다. 데이터값을 처리하여 객체의 속성 메모리에 값을 채워주는 Visualization Module타이머 와 결과를 리드로우(Redraw)하는 타이머이다. 리드로우에 대한 타이머는 낮은 우선순위

를 갖는 타이머를 이용하였다. Visualization Module처리부분은 우선순위가 높은 타이머를 사용하여, 데이터값을 처리하는 객체의 속성 메모리를 기록하는 일뿐만 아니라 자체적인 3차원 시물레이션이 가능하도록 한 사이클에 대한 루프타임을 적용할 수 있도록 하였다. Visualization Module은 dll(Dynamic Link Library)로 생성되며, [그림 3]에서처럼 Init(), Step(), Kill()함수로 구성되어 있다.

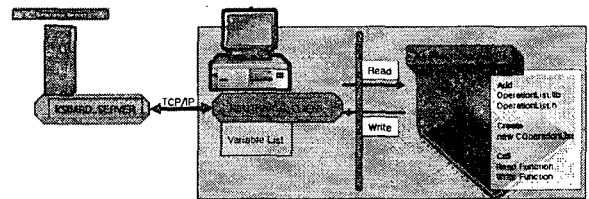


[그림 3]. 3차원 가시화 프로그램과 시물레이션 모듈과의 관계

생성된 Visualization Module은 3차원 가시화를 위한 메모리 매핑 및 처리를 담당한다. 이 함수들은 3차원 가시화의 메인 프로그램에서 호출되며, 3차원 가시화 프로그램과 생성된 모델의 관계의 구조를 보면 [그림 3]의 아래 부분과 같다. 3차원 가시화 프로그램에서 공유 메모리 생성 후 모델파일의 Init()함수를 호출하여 로드 및 초기화 작업을 수행하게 된다. 이때 Init()함수가 호출됨에 따라 데이터베이스 정보를 이용하여 공유메모리와 매핑이 이루어지고 환경설정에 의한 시물레이션 스케줄에 따라 모델파일에 있는 Step()함수를 타이머에 의해 호출된다. Step()함수는 데이터를 읽어오는 모듈과 데이터 처리 그리고 알고리즘 구현부분이며, 시물레이션시 공유메모리에 연산된 결과를 기록한다. 시물레이션 종료시 Kill()함수가 호출이 되며, 메모리 로드 및 초기화 작업으로 생성된 메모리가 모두 제거된다. 3차원 가시화 프로그램은 빌더와 뷰어, 개발된 모듈(dll)파일과 3차원 객체의 속성 파일로 이루어져 있다.

3. 시물레이터(영광1호기)와 3차원 가시화 프로그램의 연계

발전소 원자로 노심설계코드의 출력(Output)으로부터 3차원 형상을 만들어 주는 프로그램과 연계하기 위해서 TCP/IP통신을 사용하였다. 노심설계코드에서는 매우 많은 양의 데이터를 생성하는데, 이중 핵연료온도와 중성자속에 대한 데이터는 3차원 노심 모델을 가시화하는데 사용하였다. 통신을 통하여 데이터가 전달되기 때문에 빠른 전송을 위하여 3차원 가시화에 꼭 필요한 변수만을 이용하여 통신하도록 구현하였다. 이러한 변수는 중성자속의 변수인 "crsflux2j(15,15,12)"과 온도의 변수인 "crstfuelj(15,15,12)"은 총 5400개의 실수형 데이터로 이루어져 있다. 서버와 클라이언트용 컴퓨터 사이에 통신을 하기 위해서 개발된 KSIMRD_SERVER(서버측) 프로그램과 NEUTRINCS_CLIENT(클라이언트측) 프로그램은 원활한 통신을 위하여 통신하고자 하는 변수들의 목록을 기록하도록 개발하였으며 서버측과 클라이언트측 사이의 통신을 최적화시키기 위해 3차원 노심 모델 가시화에 필요한 변수들만 추출하여 기록하였다.



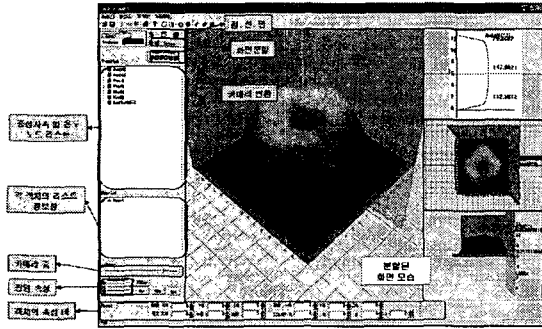
[그림 4] .3D 노심 모델 가시화 프로그램 연계

통신 연결의 구조를 보면 [그림 4]와 같다. 서버측에서는 모델이름과 패스워드를 이용한 방법으로 개별적인 클라이언트 접속이 가능하도록 하였으며, 클라이언트 프로그램에서는 서버측의 컴퓨터와 연결하기 위해 IP주소와 모델이름과 패스워드를 통하여 접속하도록 하였다. 또한, 클라이언트 프로그램은 외부 프로세서와 연계하여 동작하여야 한다. 클라이언트 프로그램은 Read() 및 Write()함수를 제공하는 동적 라이브러리(OperationList.dll)를 제공하여, 생성

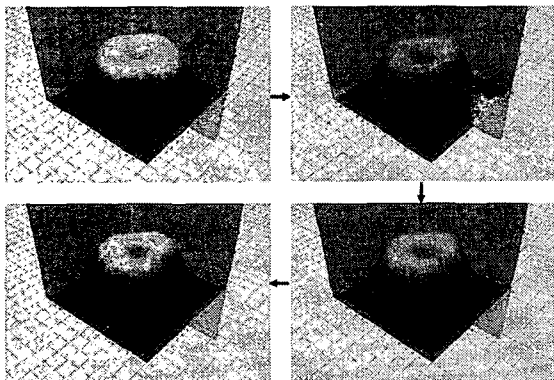
하고자하는 외부 프로세서에 라이브러리파일과 헤더파일을 추가한 후 통신 클래스 객체를 생성하여 Read()와 Write()함수를 호출함으로써 서버측의 데이터를 읽거나 쓰기가 가능하도록 개발하였다.

3. 3D노심모델 가시화 프로그램의 기능

3차원 노심 모델 가시화 프로그램의 목적으로 12개의 노드에 대하여 핵연료온도와 중성자속에 대한 값을 실시간으로 모니터링이 가능하도록 하였다. 실제 구현되어 동작되는 프로그램은 다음과 같다.



[그림 5] 가시화 프로그램 구성



[그림 6] 제어봉 Drop에 의한 노심 상태 변화

앞의 [그림 5]와 같이 노심 모델 가시화 프로그램은 다양한 기능을 갖추고 있다. 왼쪽에는 중성자속 및 핵연료온도에 대한 노드별 리스트가 나타나며, 리스트 클릭시 각각의 노드별 현재 상태를 실시간으로 확인 가능하도록 구현하였다. 3차원 객체의 표현은 데이터의 값에 따

라 색과 높이의 변화를 주어 사용자에게 빠른 이해를 도모할 수 있도록 하였으며, 3차원 그래픽의 속성(광원 속성의 변경, 카메라의 줌인아웃, 화면분할, 카메라의 빠른 변환, 점선 면 등)을 변경함으로써 다양한 관찰이 가능하도록 하였다.

4. 결론

실시간 시뮬레이터와 연계된 3차원 가시화 프로그램을 개발하였다. 개발된 가시화 프로그램은 TCP/IP통신을 통하여 데이터를 받아 3차원 가시화에 이용하였다. 지금까지 개발된 3차원 가시화 프로그램은 기본요소(Primitive)의 개수가 한정적이기 때문에, 복잡한 형태의 가시화는 표현하기에는 아직까지 미흡한 실정이며, 현재까지 개발된 RocDis™는 시뮬레이터의 상태를 표현하는데 그쳤지만, 시뮬레이터 서버에 데이터값을 조작할 수 있는 인스트럭터(Instructor)에 대한 기능이 추가적으로 요구된다.

본 논문은 과학기술부 원자력연구개발사업인 “원전 시뮬레이터 노심-계통 연계기술 개발”과제의 연구비 지원에 의하여 연구되었음.

참고문헌

- [1] 이지우, 이창훈, 이덕재, 이명수, 3차원 모델빌더와 실시간 시뮬레이션 환경 개발, 대한전기학회 추계학술대회, 2004
- [2] 홍진혁, 이명수, 이용관, 서인용, Development of the Core-Model Implementation Technology for YGN1 Simulator, 한국원자력학회 춘계학술대회, 2004
- [3] GSE, SimSuitePower3.3 Training Reference Manual