

COSMOS 코드에서의 사용자 친화적인 인터페이스 구축

Construction of the User-friendly Interface for the COSMOS code

오제용, 이병호, 구양현, 천진식, 손동성

한국원자력연구소
대전광역시 유성구 덕진동 150

요 약

컴퓨터의 눈부신 발달에 의하여 프로그램의 생산성은 단순한 계산 속도뿐만이 아니라 얼마나 간편하고 사용자 친화적인 Man-Machine Interface를 구축하여 하느냐에 크게 영향을 받게 되었다. 이런 Interface의 대표적인 것이 윈도우(window) 환경이나 메뉴(menu), 버튼(button), 아이콘(icon) 등을 사용한 GUI (Graphical User Interface)이다. 따라서 각종 인자들의 변화를 메뉴에서 간편하게 입력하기 위하여 COSMOS 코드에 GUI를 도입하였으며, 화면과 Postscript 형식 파일로 각각 결과 그래프를 출력하여 직관적이며 편리한 결과 검토가 이루어질 수 있게 하였다. 한편 그래프 라이브러리를 제작하여 손쉬운 그래프 출력이 가능하도록 하였으며, 이 라이브러리의 높은 이식성으로 다른 코드 개발 시에 재활용될 수 있을 것이다.

Abstract

Due to the dramatic improvement of the computer technology, the productivity of the program depends on not only the speed of calculations, but also on the convenience and user-friendliness of Man-Machine Interface. Graphic user interface (GUI) is one of the those user-friendly Man-Machine Interface, which consists of windows, menus, buttons, icons, and so on. As the GUI was introduced to the nuclear fuel performance code COSMOS, various parameters can be input conveniently by menu structure and output to the result graphs on screen and postscript files. This makes it easy to compare results intuitively. Because the graphic library used in COSMOS was made with Fortran and had good portability, it can be reusable for other Fortran codes with little efforts.

1. 서론

1990년대 이후로 컴퓨터 특히 PC는 발전 속도는 눈부시다. 불과 10년만 IBM AT와 펜티엄4 PC간의 속도 차이는 비교할 수 없을 정도가 되었다. 따라서 전에는 대형 컴퓨터에서 실행되던 프로그램들이 PC에서 실행될 수 있게 되었으며, 이런 배경으로 COSMOS가 개발되었다. 현재는 아무리 복잡한 입력이라고 COSMOS에서는 5분 이내에 계산을 완료할 수 있게 되었다. 그러나 초기 COSMOS가 DOS 기반으로 개발된 관계로 입력파일의 작성과 구성에 어려움이 있었으며, 입력파일을 구성하는데 걸리는 시간이 계산에 걸리는 시간보다 훨씬 길게 되었다. 따라서 코드의 생산성은 단순한 계산 속도뿐만이 아니라 얼마나 편리하게 입력파일을 구성하고 여러 인자들을 입력할 수 있으며, 결과를 한눈에 이해할 수 있게 하는냐에 달려있다. 이런 간편하고 직관적인 Man-Machine Interface의 대표적인 것이 윈도우(window) 환경이나 메뉴(menu), 버튼(button), 아이콘(icon) 등을 사용한 GUI (Graphical User Interface)이다. 따라서 본 연구에서는 각종 인자들의 변화를 메뉴에서 간편하게 입력하기 위하여 COSMOS 코드에 GUI를 도입하였다.

2. 개발환경

초기 버전의 COSMOS코드는 PC의 DOS 환경에서 돌아가는 것이었다. 따라서 코드 재개발의 노력을 줄이기 위하여 윈도우에서 Compaq Visual Fortran(CVF)을 사용하여 프로그래밍을 하는 경우에도 DOS기반인 Fortran Console Application이라는 프로젝트 형태를 사용하였다. 그러나 이러한 상태로는 GUI와 그래픽 출력부분의 프로그램이 어렵기 때문에 GUI부분을 COSMOS 코드에 추가하기 위해서 윈도우상에서 돌아가는 구조를 갖는 프로젝트를 선택하여야 한다. 이 형태의 프로젝트로는 Fortran QuickWin Application과 Fortran Windows Application이라는 두 종류가 존재한다 [1]. Fortran QuickWin Application 프로젝트의 경우는 DOS 프로그램의 구조를 그대로 사용할 수 있는 장점이 있으나 윈도우 프로그램으로서는 기능이 다소 제한적이다. 반면 Fortran Windows Application 프로젝트에서는 윈도우 프로그램으로서의 모든 기능을 사용할 수 있으나 프로그램 구조자체를 윈도우 환경에 맞도록 모두 고쳐하는 부담이 있다. COSMOS에서 필요한 기능의 수준과 프로그래밍상의 용이성을 검토한 결과, Fortran QuickWin Application 프로젝트를 선택하였다.

3. GUI 구현

3.1 메뉴

기본 메뉴 구조는 Fortran QuickWin Application 프로젝트에서는 자동적으로 만들어지는 메뉴를 사용했으며, 몇몇 필요한 메뉴를 추가하는 방식으로 만들어 졌다. 새

로운 메뉴를 만들었을 경우에는 그 메뉴가 선택되었을 경우 작동될 서브루틴을 만들고 메뉴와 연결시켜 주어야 한다. 메뉴와 관련된 명령은 표 1에 간단히 정리되어 있다 [1].

한편 입력을 위한 다이얼로그 창 제작은 다음과 같다. CVF의 Insert메뉴에서 Resource를 선택하면 편집화면이 나온다. 이 상태에서 원하는 구성을 만들면 프로그램에서 사용할 수 있는 Dialog 화면이 완성된다. 버튼이나 Edit box 같은 것들은 고유의 ID를 갖는데 이 ID를 이용하여 프로그램 상에서 원하는 제어를 수행할 수 있다. Dialog 관련하여 많이 사용하는 명령어들을 간단히 정리하면 표 2와 같다 [1].

3.2 그래프 라이브러리

그래프 라이브러리는 그래픽에 대한 상세한 지식과 복잡한 절차를 사용할 필요가 없이 간단한 몇 개의 명령어로 그래프를 그리기 위하여 만들어졌다. 이 그래프 라이브러리를 이용하면 화면 구조에 대한 깊은 지식이 필요없이 간단히 그래프를 만들 수 있으며 이식성이 좋아서 다른 코드에 이식하여 그래프를 구현하는 것이 용이하다. 그래프 라이브러리를 사용하여 그래프를 그리는 경우는 다음 세가지 과정을 거친다.

3.2.1 자료 저장

라이브러리를 사용하여 그래프를 그리기 위해서는 X축 자료와 Y축 자료에 대한 자료가 배열에 들어 있어야 한다. 자료는 REAL(4) 형이어야 한다.

3.2.2 그래프 정보 설정

그래프를 그리기 위해서는 그래프, X-Y 축, 자료에 대한 정보를 설정해야 한다.

- 그래프 위치, 타이틀, 종류, 배경색 등 그래프 자체에 대한 설정
- 자료에 대한 타이틀, 자료를 어떤 식으로 표현할지(선, 바, marker), 자료의 숫자, 자료의 범위 등에 대한 설정
- 축에 대한 설정 : 그래프의 scale, 그래프에 표시할 자료 범위, 축의 위치, 축을 Logarithmic 또는 Linear로 표시할 것인가, grid에 대한 설정

위의 설정을 모두 끝내면 GetDataDefaults와 GetAxisDefaults를 호출해 준다. 이 호출에 의하여 설정 내에서 화면에 출력할 최적 조건을 찾아서 저장한다.

3.2.3 그래프를 화면에 표시

앞의 두과정이 끝나면 실제로 그래프를 그리기 위하여 다음과 같은 절차를 거친다.

- PlotGraph를 호출한다. 이 함수는 열려진 윈도우 상에 3.2.2 항에서 설정한 대로 그래프의 배경, 경계, 축, 각종 타이틀들을 그린다.
- PlotData를 호출한다. 자료들을 앞에서 그려진 그래프 위에 표시한다.

3.3 Postscript 출력

COSMOS에서 계산이 종료된 후에 결과는 화면에 그래프로 표시되며, 동시에 text 파일로 출력된다. 화면에 출력된 그래프를 프린터로 출력하여 자료로 활용할 수 있으나 프로그램이 종료된 후에는 결과 그래프를 사용할 수 없기 때문에 화면과 동일한 그래프를 파일로 출력하도록 함수를 만들었다. 상용이나 공개 프로그램 중에 이런 기능을 하는 것들도 존재하나 나중에 라이선스 문제가 있을 수 있으므로 직접 제작하는 방식을 택하였다. 파일 형식으로는 기본적인 구조와 명령어가 공개되어 있는 Postscript를 사용하였다. Postscript는 다른 포맷들과는 달리 그림의 비트맵 이미지를 갖고 있는 것이 아니라 각각의 그래픽 요소들을 정의하는 명령어들로 구성되어 있으므로, 확대와 축소를 하여도 이미지의 손실이 적으며, text 편집기 등을 사용하여 직접 편집이 가능하다. 특히 Postscript에서 점과 선, 문자 등을 그리는 명령어들은 기본적으로 화면에 그래픽을 출력하는 함수들과 매우 유사한 구조를 갖고 있으며, 표 3에서와 같이 Postscript 명령어과 거의 일대일 대응이 되는 그래픽 명령어들이 존재한다 [2]. 따라서 COSMOS에서는 화면에 그래픽이 출력됨과 동시에 그에 상응하는 postscript 명령어가 text 형식으로 출력되어 postscript 파일이 만들어지도록 하였다.

4. 결과

그림 1 - 6은 COSMOS를 실행하여 입력파일을 열고 계산을 수행한 후에 결과를 출력하는 모습을 차례대로 캡처한 그림이다. COSMOS를 실행하면 그림 1에서와 같이 윈도우 창내에 도스창 같은 것이 들어있는 초기화면이 나온다. 이 초기화면은 Fortran QuickWin Application 프로젝트로 제작된 프로그램의 특징으로 도스창에는 도스 상에서 실행되는 명령어들의 결과가 출력되며, 그래프들은 별도의 창이 열려서 실행된다. 입력 파일을 선택하기 위해서는 그림 2에서처럼 File 메뉴를 클릭한 후 Open 메뉴를 선택하면 그림 3과 같이 원하는 입력파일을 선택할 수 있는 창이 열리게 된다. 이 동작은 표준 윈도우 함수를 사용하였으므로, 다른 프로그램과 동일한 조작법을 갖는다.

입력 파일이 선택되면 자동적으로 그림 4의 다이얼로그 창이 떠서 계산에 필요한 인자들을 받아들이도록 한다. 여기서 입력받는 인자들은 기본조사/재조사의 선택, 관찰할 노드 번호, 각 변수들을 관찰할 범위 등이며 입력이 끝난 후 "Calculation" 버튼을 누르면 계산이 실행된다. 모든 계산이 완료된 후에는 화면상에서는 그림 5와 같이 연소도에 대한 선출력, 핵연료 중심온도, 산화막 두께, 핵분열 기체 방출도, 봉내 압력에 대한 그래프가 그려지고 이것과 동일한 그래프가 그림 6과 같이 Postscript 파일로도 출력된다.

5. 결론

각종 인자들의 변화를 메뉴에서 간편하게 입력하기 위하여 COSMOS 코드에 GUI를 도입하였으며, 화면과 Postscript 형식 파일로 각각 결과 그래프를 출력하여 직관적이며 편리한 결과 검토가 이루어질 수 있게 하였다. 한편 그래프 라이브러리를 제작하여 손쉬운 그래프 출력이 가능하도록 하였으며, 이 라이브러리의 높은 이식성으로 다른 Fortran 코드 개발 시에 재활용되어 적은 노력으로 그래프 기능을 구현할 수 있을 것이다.

감사의 글

본 연구는 과학기술부의 중장기 원자력 연구개발 사업의 일환으로 수행되었음.

참고문헌

1. Help Files in Compaq Visual Fortran Professional Edition 6.1.A
2. <http://www.math.ubc.ca/~cass/courses/ps.html>

표 1. 메뉴 관련 명령어 요약

명령어	설명
INSERTMENUQQ	메뉴 리스트 중 지정된 위치에 메뉴 항목을 하나 추가하고 관련 서브루틴을 등록한다.
APPENDMENUQQ	메뉴 리스트의 끝에 메뉴 항목을 하나 추가하고 관련 서브루틴을 등록한다.
DELETEMENUQQ	지정된 메뉴 항목을 제거한다.
MODIFYMENUSTRINGQQ	메뉴 항목의 이름을 바꾼다.
MODIFYMENUROUTINEQQ	메뉴 항목에 관련된 서브루틴을 바꾼다.
MODIFYMENUFLAGSQQ	메뉴 항목의 상태를 바꾼다.

표 2. Dialog 관련 명령어 요약

명령어	설명
DLGINIT	선택한 Dialog를 초기화하여 화면에 표시할 준비를 한다.
DLGSETSUB	구성요소가 클릭되었을 경우 실행할 subroutine를 설정한다.
DLGSET	구성요소를 원하는 값으로 설정한다.
DLGMODAL	Dialog를 표시하며 설정된 행동을 수행한다.
DLGGET	구성요소에서 값을 읽어 들인다.
DLGEXIT	열린 Dialog를 닫는다.

표 3. Postscript 명령어와 그래픽 명령어의 비교

Postscript 명령어	그래픽 명령어	설명
moveto	MOVETO	현재 위치를 바꾼다
lineto	LINETO	선을 그린다
arc	ARC	호를 그린다
setfont	SETFONT	폰트를 설정
show	OUTGTEXT	문자열을 출력

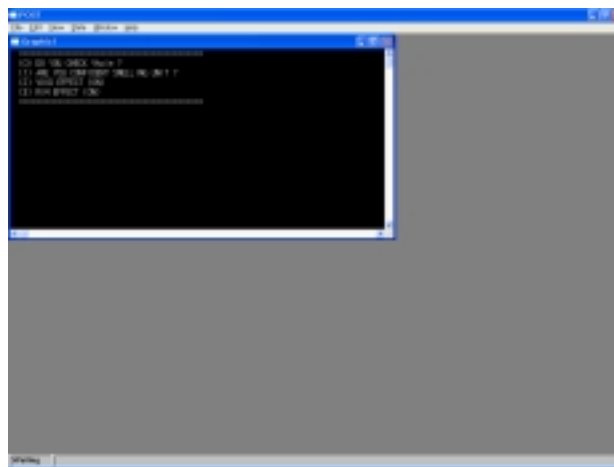


그림 1. GUI의 처음 실행 화면

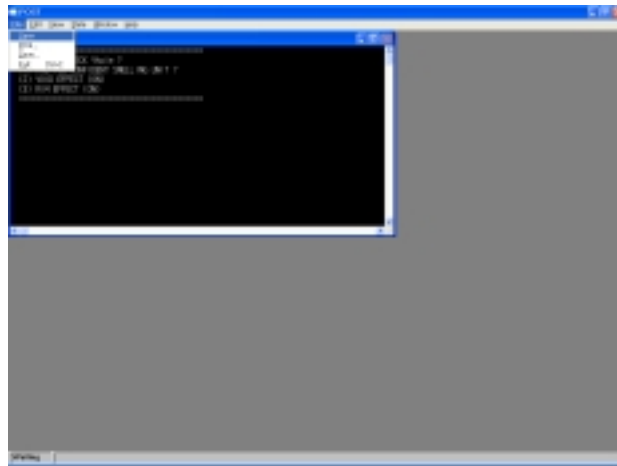


그림 2. Open 메뉴 선택

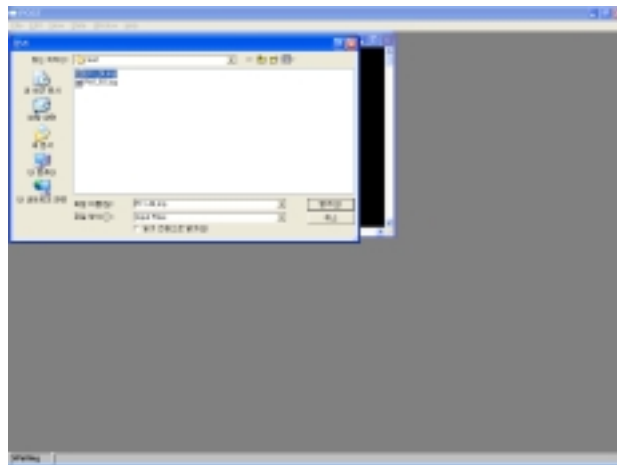


그림 3. 입력파일 선택창

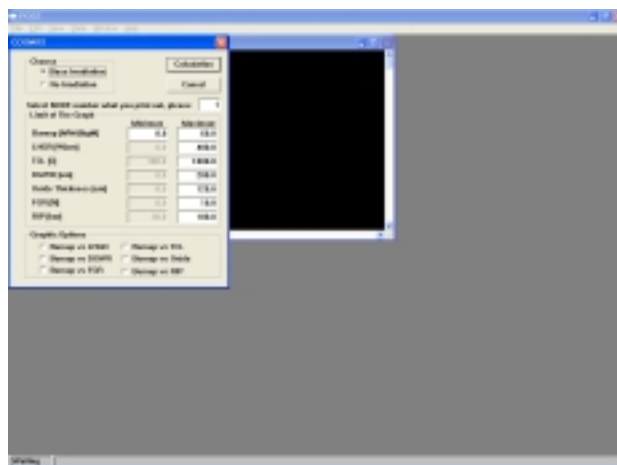


그림 4. 인자입력을 위해 열리는 다이얼로그 창

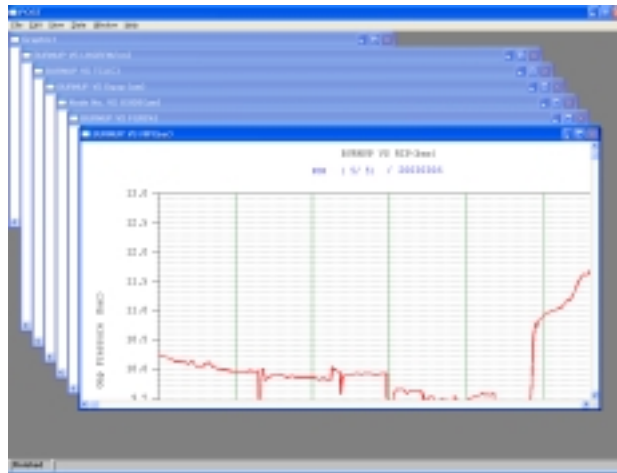


그림 5. 결과 출력 화면

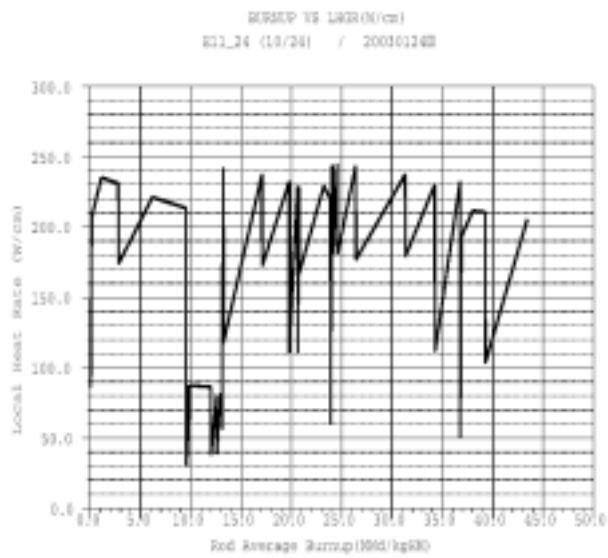


그림 6 Postscript 파일 출력 예