

<기술논문>

OOP 개념에 기초한 유동해석용 전처리 프로그램 개발

명 현 국[†] · 안 종 기^{*}

(2007년 7월 25일 접수, 2007년 10월 15일 심사완료)

Development of a Pre-Processing Program for Flow Analysis Based on the Object-Oriented Programming Concept

Hyon Kook Myong and Jong Ki Ahn

Key Words: Pre-Processing(전처리), OOP(Object-Oriented Programming; 객체지향프로그래밍), Class (클래스), GUI(Graphic User Interface; 그래픽 사용자 인터페이스), Flow Analysis(유동해석), VTK(Visualization ToolKit)

Abstract

A pre-processing program based on the OOP(object-oriented programming) concept has been developed. The program consists of the input of a 2D or 3D flow problem to a CFD program by means of an user-friendly interface and the subsequent transformation of this input into a form suitable for the solver(PowerCFD) using unstructured cell-centered method. User-friendly GUI(graphic user interface) has been built on the base of MFC(Microsoft Foundation Class). The program is organized as modules by classes based on VTK(Visualization ToolKit)-library, and these classes are made to function through inheritance and cooperation which is an important and valuable concept of object-oriented programming. The major functions of this program are introduced and demonstrated, which include mesh generation, boundary settings, solver settings, generation of grid connectivity and geometric data etc.

1. 서 론

복잡한 열/유체 유동현상을 수치해석방법으로 계산하고자 하는 경우, 문제의 형상에 대하여 격자를 구성하고 경계조건을 입력하는 전처리기(pre-processor), 유동현상을 지배하는 지배방정식을 적절한 수치해석방법을 사용하여 계산결과를 얻기 위한 솔버(solver), 솔버로부터 얻어진 결과를 이해 및 분석하기 용이하도록 컴퓨터그래픽으로 표현하여 주는 후처리기(post-processor)가 필요

하게 된다. 이중 솔버를 위한 수치해석방법에 대한 연구는 국내 대학 및 연구소에서 활발하게 진행되어 비교적 많은 양의 연구결과가 축적되어 있다. 그러나 전처리기와 후처리기에 대한 국내 기술개발은 외국에 비해 상대적으로 매우 미흡한 실정으로, 특히 전처리기에 대한 국내 기술개발은 거의 전무한 상태이다.^(1,2)

이와 같이 국내의 전처리 및 후처리 기술개발 수준이 외국에 비해 매우 낙후된 주요 원인은 CFD 해석 소프트웨어의 핵심엔진인 솔버에 대한 기술개발이 완성되지 않고, 또한 상업적으로 연계되지 못하기 때문으로 사료된다. 특히 전처리기는 계산에 필요한 계산격자를 생성하기 위하여 복잡한 형상을 입력하여 필요한 블록으로 분할한 뒤 격자를 생성하는 것으로 격자생성 기술이 중

[†] 책임저자, 회원, 국민대학교 기계자동차공학부

E-mail : myong@kookmin.ac.kr

TEL : (02)910-5030 FAX : (02)910-4839

^{*} 회원, 국민대학교 대학원 기계공학과

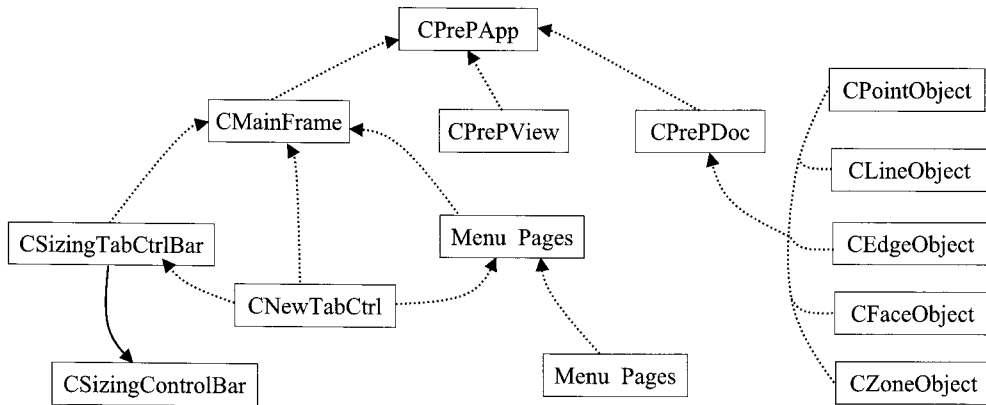


Fig. 1 Class inheritance and collaboration diagram

요하다. 또한 전처리기에는 격자생성과 함께 물성치 정의, 경계조건 설정, 물리 현상 모델 설정 등 많은 복잡한 기술이 복합적으로 포함되기 때문에 기술개발에 어려움이 따른다. 따라서 현재는 주로 외국에서 개발된 상용 CFD 코드의 전처리 장치를 사용하거나 전문 격자생성 프로그램을 사용하여 격자를 생성한 후 별도의 변환 프로그램을 사용하여 해석코드인 솔버에서 받아들이 수 있는 파일형식으로 변환하여 사용하고 있다. 이런 방법은 여러 프로그램을 사용하여 번거로운 뿐만 아니라 외국 상용 CFD 코드의 많은 설정과 기능들로 인하여 사용법을 익히기도 어렵고 외화도 낭비되는 문제점을 가지고 있다. 따라서 국내에서도 외국 상용 CFD 코드의 전처리기와 유사한 유동해석용 전처리 프로그램 개발이 절실히 요청되고 있다.

최근 저자는 비정렬 셀 중심 방법(cell-centered method)에 기초한 3차원 유동해석 솔버인 PowerCFD 코드^(5,6)를 개발하면서, 동시에 GUI(Graphic User Interface) 환경하에서 OOP(Object-Oriented Programming; 객체지향 프로그래밍) 개념에 기초하여 비정렬격자를 생성하고 경계조건 등을 입력할 수 있는 전처리 프로그램(PowerCFD/PreP v1.0)을 일련의 연구^(3,4)를 통해 개발하였다. 이 전처리 프로그램은 VTK(Visualization Toolkit)⁽⁷⁻⁹⁾라는 OOP 개념으로 설계·구현된 3차원 그래픽 라이브러리를 사용함으로써 보다 강력하고 편리한 기능구현이 가능하고 모듈(module)화를 이용한 단순한 구조의 프로그램을 통해 변경이나 확장에 유연하다. 또한 자유로운 형태의 격자생성이

용이하며, 생성된 격자의 연결정보와 경계조건을 파일로 생성해 솔버(PowerCFD)가 계산을 위해 받아들일 수 있다.

본 논문에서는 OOP 개념에 기초하여 개발된 유동해석용 전처리 프로그램(PowerCFD/ PreP v1.0)의 구조, 주요 기능을 소개한다.

2. 프로그램의 구조

2.1 개발 목표

PowerCFD/PreP v1.0은 다음과 같은 목표로 개발되었다. 첫째, 특화된 프로그램 개발로, 프로그램 사용방법을 익히는데 들어가는 시간을 최소한으로 줄여주기 위해 외국 상용 CFD 소프트웨어들의 기능들 중에서 많이 쓰이면서도 중요한 기능을 구현하여 원하는 형상의 격자를 보다 쉽고 간편하게 생성하는 것이다. 둘째, 사용자에게 친숙한 환경을 제공해줄 수 있는 편리한 GUI 환경을 구축하는 것으로, 이것은 사용자들을 확보하고 프로그램을 성장시킬 수 있는 가장 기본적인 면서 중요한 사항이다. 셋째, 확장성 및 관리의 용이성을 위해 OOP 개념에 기초하여 프로그램 구조를 설계하는 것이다.

2.2 클래스

Fig. 1은 클래스(class) 협력관계를 도표로 나타낸 것이다. 여기서 각각의 박스들은 클래스를 의미하고, 실선 및 점선은 각각 상속(inheritance) 및 협력(collaboration) 관계를 나타내고 있다. VTK에

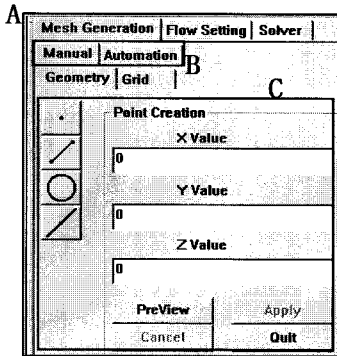


Fig. 2 Structure of menu page

서 제공하는 클래스는 상당히 많기 때문에 도표에서 생략하였다. 클래스 협력관계 도표에 나타난 클래스들 중에서 주요 기능을 담당하는 클래스는 다음과 같다.

먼저, PowerCFD/ PreP의 전체적인 GUI를 구성하는 역할을 담당하는 것이 CMainFrame, CSizingTabCtrlBar, CSizingControlBar 및 CNewTabCtrl 클래스이다. 간단히 각 클래스의 역할을 설명하면, CSizingControlBar 클래스는 모든 설정과 프로그램의 진행이 이루어지는 메뉴 페이지(menu pages)의 기본 틀을 구성하며, 이 클래스를 상속받아 CNewTabCtrl 클래스를 부착할 수 있도록 설계된 것이 CSizingTabCtrlBar 클래스이다. 이 클래스는 각각의 Tab에 메뉴 페이지를 부착할 수 있도록 설계되었으며 이것을 가능하게 하는 것이 CNewTabCtrl 클래스이다.

Fig. 2는 이런 구조로 생성된 메뉴 페이지이다. 메뉴 페이지도 CNewTabCtrl 클래스를 멤버(member)로 가질 수 있기 때문에 하위에 또 다른 메뉴 페이지를 부착시킬 수 있다. 참고로 그림상에 나타난 A, B 그리고 C에 대해 간략히 설명하면 다음과 같다.

A : CSizingTabCtrlBar 클래스로 전체적인 틀을 형성한다. 각각의 탭에는 서로 다른 메뉴 페이지가 붙어있다.

B : 그림의 모든 탭은 CNewTabCtrl 클래스로 구현된 것이다. 각각의 Tab에 메뉴 페이지가 부착될 수 있도록 실제적인 역할을 한다.

C : Tab에 부착된 메뉴 페이지이다.

CPrePDoc 클래스는 점(point), 선(line), 모서리

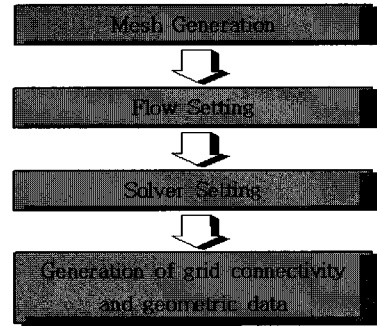


Fig. 3 Flow chart of present pre-processing program(PowerCFD/PreP v1.0)

(edge), 면(face) 및 존(zone) 등 모든 객체(object)를 생성하는 역할을 한다. 예를 들어 모서리 객체를 생성하는 경우 선 객체를 일정비율로 나누어 각각의 위치에 점을 생성하는 알고리즘이 구현되는 곳이 CPrePDoc 클래스이다.

CPrePView 클래스는 생성된 모든 객체들을 화면에 나타내거나 감추는 일과 마우스의 선택(pick) 기능을 사용하여 선택된 객체의 색깔을 붉은색으로 바꾸는 것과 같이 그래픽관련 역할을 하는 클래스이다.

2.3 프로그램의 순서도

프로그램이 실행되는 기본 절차를 간단하게 표현하면 Fig. 3과 같다. 즉, 먼저 수동(manual) 또는 자동(automation)으로 계산격자를 생성한 후, 유동 설정(flow setting)에서 생성된 격자에 경계조건, 초기조건 및 필요한 물성값 등을 설정한다. 다음에 솔버 설정(solver setting)에서 솔버에 관한 여러 가지 옵션 등을 설정하고, 마지막으로 격자 연결정보와 격자 파일을 생성하여 솔버로 넘겨주는 것이다.

2.4 GUI 구성

본 연구를 통하여 개발된 프로그램의 화면 구성은 Fig. 4와 같이 이루어져 있으며 각 영역의 기능은 다음과 같다.

A) Toolbar : On/Off가 가능한 단추로 구성되어 있어 Graphic Viewport에서 객체들을 보이거나 감추는 기능을 한다.

B) Graphic Viewport : 3차원 그래픽 처리 영역.

C) InfoView : Graphic Viewport에 생성된 여러

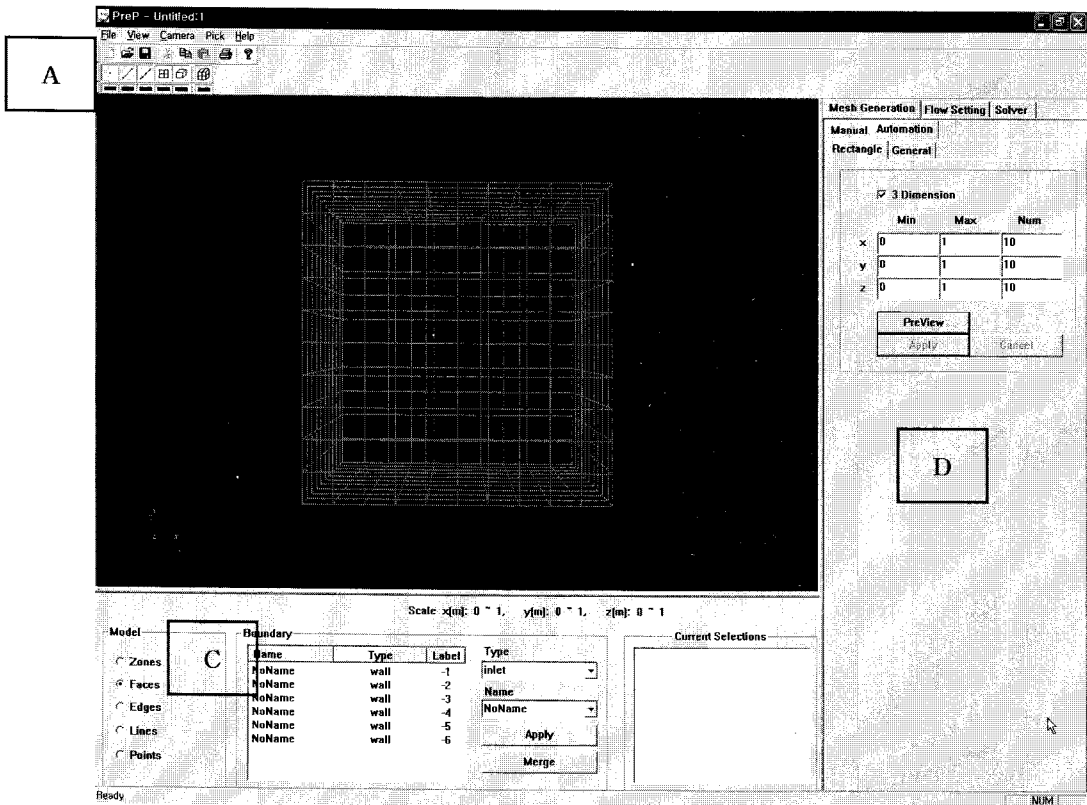


Fig. 4 Main window of GUI

객체들의 정보를 보여준다.

D) 메뉴 페이지 : 격자를 생성하고 격자에 경계 조건을 부여하는 등의 프로그램 전반적인 진행을 담당한다.

3. 격자 생성

본 연구를 통하여 개발된 프로그램의 격자생성은 자동격자(automation mesh)와 수동격자(manual mesh) 방법이 있다.

3.1 자동격자(Automation Mesh)

자동격자 방법은 계산영역이 사각형(2차원인 경우) 또는 직육면체(3차원인 경우)인 경우 각 좌표축 방향으로의 길이와 등분 개수를 정해서 정렬격자를 생성하는 것과 같은 방법으로 격자를 생성하는 것이다. 형상이 정해져 있는 단점이 있지만 간편하게 격자를 생성할 수 있어 교육용으

로 적합하다. Fig. 4는 자동격자 방법으로 생성된 격자와 메뉴 페이지를 나타내고 있다.

3.2 수동격자(Manual Mesh)

수동격자 방법은 일반적인 CFD 코드의 전처리기와 마찬가지로 원하는 형상을 자유롭게 만들 수 있는 방법으로써 점, 선, 모서리의 순서로 각 요소를 수동적으로 생성한 후, 격자생성(mesh generation)을 자동으로 수행하여 완성된 격자를 만들어 가는 방식이다. 각 요소의 생성 방법을 살펴보면 다음과 같다.

점 생성(point generation)

점 생성은 좌표로 x, y, z 값을 입력하여 생성하는데 원점의 위치는 초기에는 Graphic Viewport의 중심에 위치해 있으나, 상황에 맞추어 유동적으로 변하게 하여 생성된 객체들이 화면을 벗어나 생성되어도 좌표축이 재설정되어 모든 객체들을 한눈에 볼 수 있다.

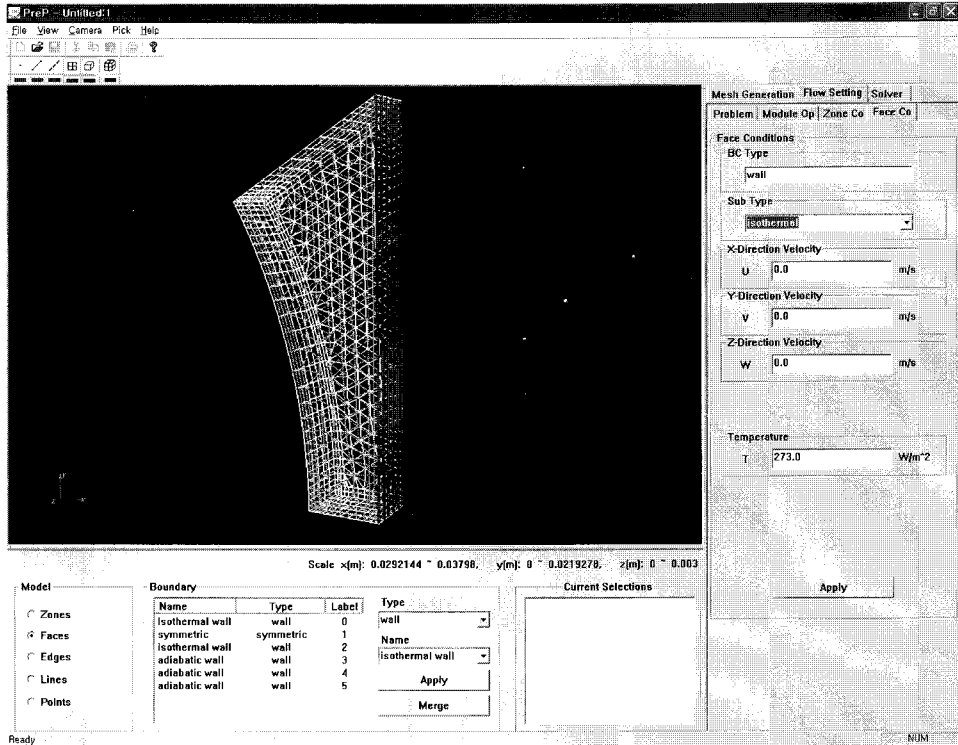


Fig. 5 An example of manual mesh generation and flow setting

선 생성(line generation)

선 생성은 두 개의 점을 연결하여 생성할 수 있으며, 점은 마우스를 이용하여 직접 선택할 수 있다. 먼저 선택된 점이 시작점이 되고 나중에 선택된 점이 끝점이 된다. 원이나 호를 그리기 위해서는 먼저 중점이 되는 점이 생성되어 있어야 한다. 생성방법은 먼저 관련 메뉴 페이지를 열고 마우스를 이용하여 중점을 선택한다. 그러면 초기에 비활성 상태인 메뉴 페이지가 활성화 되어 반지름과 시작 각도와 끝 각도를 입력할 수 있다. 이렇게 중점의 좌표와 시작 각도와 끝 각도를 이용하여 호를 생성한다. 또한, 선을 생성하는 중요 옵션으로 선의 한 점을 기준으로 선을 두 부분으로 나누는 기능이 있다. 이러한 기능은 격자생성을 보다 자유롭게 해준다.

모서리 생성(edge generation)

모서리 생성은 마우스로 선을 선택하면 미리 설정해 놓은 비율로 선을 따라서 점이 생성되어 만들어진다. 비율은 0에서 2까지의 값을 가지는 확대비(expansion factor)로 설정한다. 선과 마찬가지로 모서리 객체도 모서리 위의 한 점을 이용하

여 두 부분으로 나눌 수 있다.

격자생성(mesh generation)

현재 격자생성은 삼각형 격자와 사각형 격자 두 종류를 생성할 수 있다. Fig. 5는 두 종류의 격자가 혼합된 격자생성의 예를 보여주고 있다. 또한, 2차원 격자는 3차원으로 확장할 수 있는데, 만일 격자가 x-y 평면상에 존재한다면 z축 방향으로 격자를 적층시키는 방법으로 2차원 격자를 3차원으로 확장할 수 있다. 여기서 사각형 격자는 육면체가 되고 삼각형 격자는 프리즘으로 바뀐다. 또한, 생성된 격자는 존으로 확정할 수 있으며, 이때 격자를 구성하는 모서리 객체는 면으로 자동적으로 설정된다.

4. 파일 가져오기 및 저장

4.1 파일 가져오기(file import)

이 기능은 격자를 프로그램에서 직접 생성하지 않고 외부 파일을 읽어서 격자 객체를 생성하는 것으로, 다른 상용 소프트웨어의 격자생성 프로

그램을 이용하거나 CAD와 같은 격자생성 전문 프로그램을 이용하여 격자를 생성했을 경우라도 본 프로그램의 파일 가져오기 기능을 이용하여 기하학 객체들을 생성할 수 있도록 하기 위한 것이다. 현재 개발단계에 있으며 가져오기 가능한 파일은 상용 소프트웨어인 CFD-ACE의 전처리를 이용하여 생성한 MFG 파일과 본 프로그램으로 생성한 격자파일 등이 있다.

4.2 저장(save)

이 기능은 격자 생성 도중이나 완성 후에 현재의 격자 정보를 파일로 저장하기 위한 것으로, 복잡한 형상의 격자를 생성할 경우나 혹은 필요에 의해 현재의 격자 정보를 파일로 저장하여 나중에 파일 가져오기 기능을 이용하여 격자를 생성하기 위한 것이다. 저장되는 객체는 격자점(vertex), 선(line), 모서리(edge), 면(face) 그리고 존(zone) 등이 있다.

5. 유동 설정(flow setting)

유동 설정은 크게 Problem Type, Module Option, Zone Conditions, Face Conditions 및 Initial Conditions로 나뉘어져 있다.

Problem Type

Problem Type은 해석하고자 하는 문제의 유동을 정의하는 것으로 압축성 유동인지, 난류유동인지, 점성을 고려할 것인지 그리고 상변화를 고려할 것인지 등을 결정한다.

Module Option

Module Option에서는 축대칭성, 체적력(body force), 대표길이(reference length), 비정상 조건(transient condition) 및 기타 여러 가지 세부옵션을 설정하는 부분인데 사용자가 잘 알지 못하는 여러 변수들은 특별히 고려하지 않아도 되도록 기본값을 설정해 두었다.

Zone Conditions

Zone Conditions는 생성된 존이 유체인지 혹은 고체인지를 정하고 그에 따라 밀도, 점성계수, 비열 그리고 열전도 계수 등 여러 물성 값들을 설정하는 부분이다.

Initial Conditions

Initial Conditions는 각 존의 물성 값을 설정하면 대화상자(dialog)가 자동으로 생성되어 속도와,

압력 그리고 온도의 초기값을 설정할 수 있다.

Face Conditions

Face Conditions는 경계조건을 설정하는 것으로 크게 입구, 출구, 벽면, 계면(interface), 압력 및 대칭 경계조건 등으로 나누어지는데 유동의 상태 및 주어진 조건에 따라 다시 Sub Type을 정해 최종적으로 key값으로 저장된다(Fig. 5 참조).

예를 들어 입구 경계조건에서 비압축성 유동인 경우 Sub Type은 두 가지, 즉 고정 속도(fixed velocity)와 고정 정체압력(fixed stagnation pressure)로 나누어진다. 고정 속도는 경계조건 값으로 속도성분을 받아들이며 고정 정체압력은 정체압력 값을 입력받는다.

본 연구에서 개발된 프로그램은 경계조건의 입력 시에 사용자의 혼동을 방지하기 위해서 유동의 종류와 같은 Problem Type을 정하면 그것에 따라 입력항목이 바뀌어 사용자가 문제에 필요한 조건만을 입력할 수 있게 하였다.

6. 솔버 설정(solver setting)

솔버 설정은 솔버의 전반적인 옵션들을 설정하는 것으로 Iteration, Spatial, Solver, Under Relaxation Factor, Advanced, Output 그리고 Run 페이지로 나누어진다.

Iteration Page

Iteration Page에서는 한 시간 스텝에서의 최대 반복계산 횟수와 수렴판정 기준 그리고 발산 판단기준 등을 설정한다.

Spatial Page

Spatial Page에서는 대류항에 대한 차분법을 선택하고, 1차와 2차 정확도를 가지는 차분법들을 혼합하는 브랜딩계수(blending factor)를 설정할 수 있다.

Solver Page

Solver Page에서는 이산화된 지배방정식에 대한 행렬솔버(matrix solver)의 선택과 행렬솔버의 수렴판정을 위한 오차감소 비율의 설정 그리고 행렬솔버의 최대 반복회수를 설정하고 운동량방정식의 하향 완화계수(under relaxation factor)를 설정한다.

Advanced and Output Pages

Advanced Page에서는 계산에 필요한 Solution Gradient를 구하는 방법을 선택할 수 있으며,

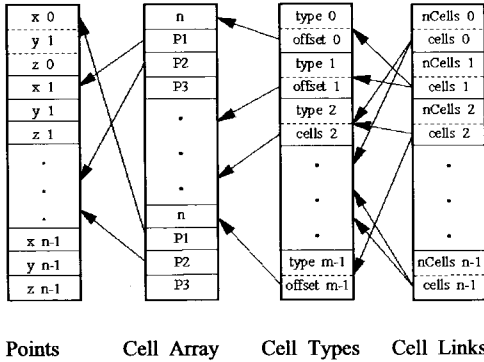


Fig. 6 The data structure of the class vtkUnstructuredGrid⁽²⁾

Output Page는 솔버의 계산결과를 모니터링하기 위한 옵션을 설정하는 부분이다.

Run Page

Run Page는 유동 설정과 솔버 설정에서 작성한 모든 데이터와 격자데이터 및 격자 연결 자료를 파일로 만들어 솔버가 받아들일 수 있도록 연결하는 역할을 한다.

7. 격자 연결자료

본 연구에서 프로그램의 패키지 모듈로서 개발된 3차원 유동해석 코드(PowerCFD)⁽¹⁾는 대부분의 일반 상용 CFD 코드와 마찬가지로 비정렬 셀 중심 방법(cell-centered method)에 기초하고 있다. 그런데 정렬격자에서는 기본적인 정의에 의해 어느 임의의 셀에 대해 이웃하는 셀의 번호를 알고 있기 때문에 격자의 연결자료에 대한 특별한 정보가 없이도 유한체적법을 적용할 수 있으나, 비정렬격자에서는 모든 셀에 대해 이웃하는 셀, 격자점, 모서리, 면 등의 정보를 필요로 한다.

비정렬 격자에서 격자의 구성요소는 점(vertex or node), 선(edge), 면(face), 셀(cell) 등이며, 이들은 서로 간에 연결되어 있다. 이 연결성에 대한 정보를 제공하는 것이 격자 연결자료로 격자점 기준 연결자료, 선(edge)기준 연결자료, 면(face)기준 연결자료, 셀 기준 연결자료 등이 있는데 본 연구에서는 Fig. 6에 나타난 것과 같이 VTK에서 제공하는 클래스 vtkUnstructuredGrid의 데이터 구조를 사용하여 각 구성요소간의 관계를 다음과 같이 정의하였다.

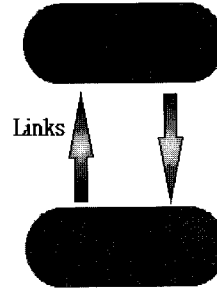


Fig. 7 Full unstructured data representation

먼저 Cell Array의 첫 번째 칸의 n은 셀을 구성하는 점의 개수를 의미하며 이하 n개의 칸에는 각각의 point id가 저장된다.

Cell Types에는 각 셀의 Type이 저장되어 Cell Array에 저장된 셀들의 형상을 명확히 하고 Offset이라는 변수가 함께 저장되어 임의의 Cell을 검색하고자 할 때 사용된다. 이것은 Cell Array만으로는 각 셀들 간의 경계가 명확하지 않기 때문이다.

Cell Links는 각 격자점(vertex)을 공유하는 셀들의 배열을 저장해 놓은 배열이다. Cell Links의 첫 번째 칸의 nCells는 한 점을 공유하는 셀들의 개수이고 cells로 표현된 다음 칸은 셀들의 ids를 저장한 배열을 나타낸다. 만일 Cell Links가 없다면 임의의 점을 포함하는 셀을 검색하는 작업은 1번 셀부터 그 점이 나올 때까지 모든 셀을 차례로 찾아 봐야하는 비효율적인 일이 될 것이다. Fig. 7은 Cell Links의 기능을 간단한 도표로 나타낸 것이다. 또 모서리 및 면과의 연결은 셀 클래스의 함수 GetEdge() 및 GetFace()를 사용하여 구한다. 이것은 셀을 구성하는 점들이 일정한 규칙으로 정렬되어 있기 때문에 따로 메모리를 할당하지 않고도 연결 관계를 알아 낼 수 있기 때문이다.

한편, 셀 중심 유한체적법에서는 모든 종속변수를 셀의 중심에 저장하고 셀의 면을 따라 적분을 하기 때문에 셀의 제어체적, 셀 중심 좌표, 면 중심 좌표, 면의 단위 수직 벡터, 면의 면적 같은 기하학적 자료(geometric data)도 필요하다. 따라서 본 연구에서 개발한 프로그램은 격자 연결자료를 바탕으로 필요한 기하학적 데이터를 계산하여 솔버에 제공하고 있다.

8. 결 론

본 논문에서는 OOP(객체지향 프로그래밍) 개념에 기초하여 개발된 유동해석용 전처리기 (PowerCFD/PreP v1.0)의 구조, 주요 기능을 소개하였다. 이 전처리 프로그램은 GUI 환경하에서 VTK(Visualization Toolkit)라는 객체지향프로그래밍 개념으로 구현된 3차원 그래픽 라이브러리를 사용함으로써 보다 강력하고 편리한 기능구현이 가능하고 모듈화를 이용한 단순한 구조의 프로그램을 통해 변경이나 확장에 유연한 전처리기이다.

향후, 이 전처리 프로그램이 전처리기에 대한 국내 연구자들의 관심을 고취시켜 국내에서 경쟁력 있는 전처리가 개발되고, 또한 국내 대학에서 CFD 교육을 위한 전처리로 개발될 수 있도록, 다양한 실험적 기법들을 채용하고 테스트할 예정이다.

후 기

본 연구는 서울시 산학연 협력사업(2005년도 신기술연구개발 지원사업)의 연구비를 지원받아 수행된 연구이다.

참고문헌

- (1) Sah, J. Y. and Huh, J. S., 2003, "Code Development for Two-Dimensional Flow Visualization," *KSCFE, J. Computational Fluids Engineering*, Vol. 8, No. 1, pp. 30~37.
- (2) Na, J. S., Kim, K. Y. and Kim, B. S., 2004, "Study on a Post-Processing Program for Flow Analysis Based on the Object-Oriented Programming Concept," *KSCFE, J. Computational Fluids Engineering*, Vol. 9, No. 2, pp. 1~10.
- (3) Myong, H. K. and Ahn, J. K., 2006, "Development of a Pre-Processing Program for Analysis based on the Objected-Oriented Programming," *Proc. of the KSME 2006 Spring Annual Meeting*, June 7-9, Jeju, Korea, pp. 771~776.
- (4) Myong, H. K. and Ahn, J. K., 2006, "Development of a Pre-Processing Program for Flow Analysis," *Proc. of the 4th National Congress on Fluid Engineering*, Aug. 23-25, Kyungju, Korea, pp. 897~900.
- (5) Myong, H. K. and Kim, J., 2005, "Development of 3D Flow Analysis Code using Unstructured Grid System(1st Report, Numerical Method)," *Trans. of the KSME(B)*, Vol. 29, No. 9, pp. 1049~1056.
- (6) Myong, H. K., Kim, J. and Kim, J. E., 2005, "Development of 3D Flow Analysis Code using Unstructured Grid System(2nd Report, Code's Performance Evaluation)," *Trans. of the KSME(B)*, Vol. 29, No. 9, pp. 1057~1064.
- (7) <http://kitware.com>.
- (8) The VTK User's Guide, ver. 4.2, 2003, Kitware, Inc.
- (9) The Visualization Toolkit, An Object-Oriented Approach to 3D Graphics, 3rd ed., 2002, Kitware, Inc.