

[2008][06-1]



Computer aided ship design

Part 2. Ship Motion & Wave Load

October 2008

Prof. Kyu-Yeul Lee

Department of Naval Architecture and Ocean Engineering,
Seoul National University of College of Engineering

Advanced
Ship
Design
Automation
Laboratory

Part II. Ship Motion & Wave Load

: 강의 개요

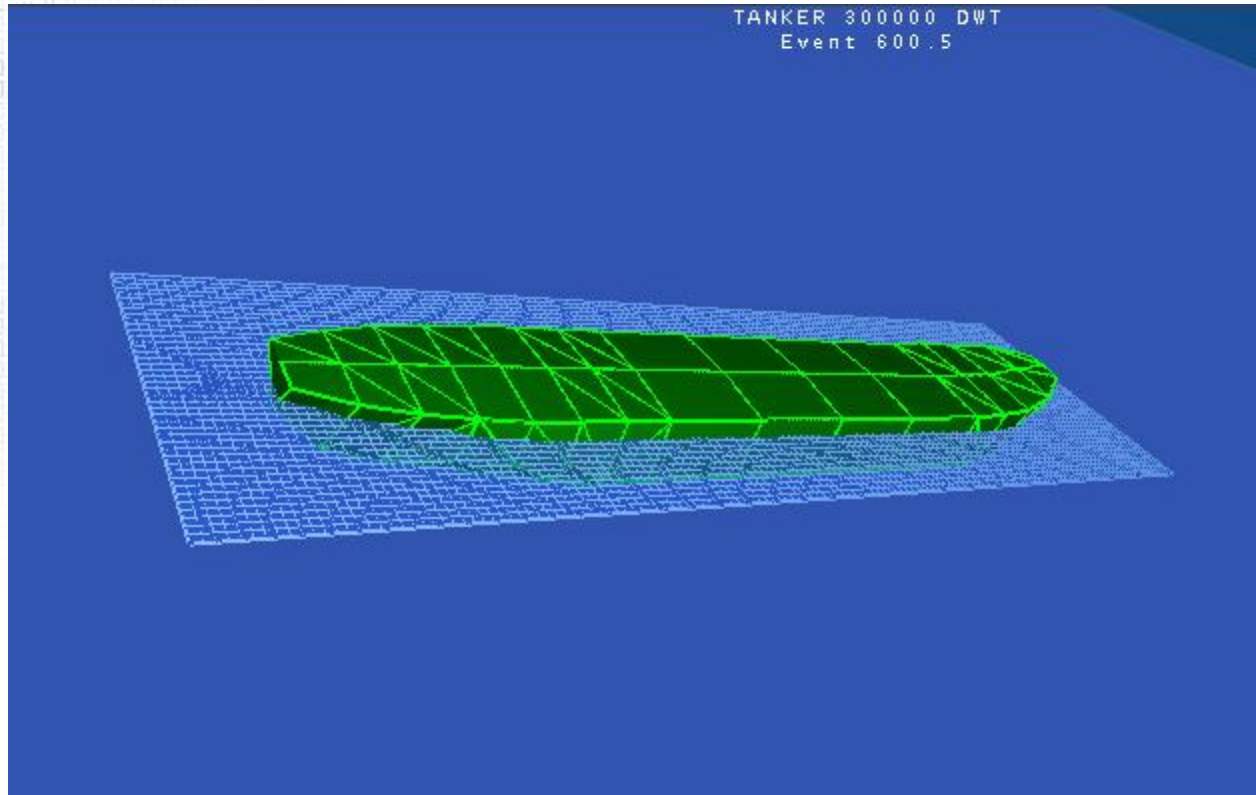
- 유체 역학(학부 2학년), 해양환경정보시스템(학부 3학년), 선박운항 제어론(학부 3학년), 선체구조설계시스템(학부 3학년)의 전공 과목과 공학 수학(학부 2학년), 기초구조정역학(학부 2학년), 동역학(학부 2학년)의 기초 교과목을 토대로 선박의 6자유도 운동 방정식의 유도 과정을 이해한다.
(Ship Motion)
- 선박의 6자유도 운동 방정식의 계산 결과를 구조 설계에 활용하는 방안에 대해 이해한다.
(Wave Load)

Part II. Ship Motion & Wave Load

: Term project (I) – 6DOF Ship motion simulation

- 선박의 6자유도 운동 방정식을 수치적으로 계산하여 시간에 따른 선박의 운동을 구하고, 이를 화면에 가시화 한다.

ex) MOSES (Multi-Operational Structural Engineering Simulator)
: Ultramarine 에서 개발(미국, 휴스턴)



Part I. 곡선, 곡면
모델링의 결과 사용

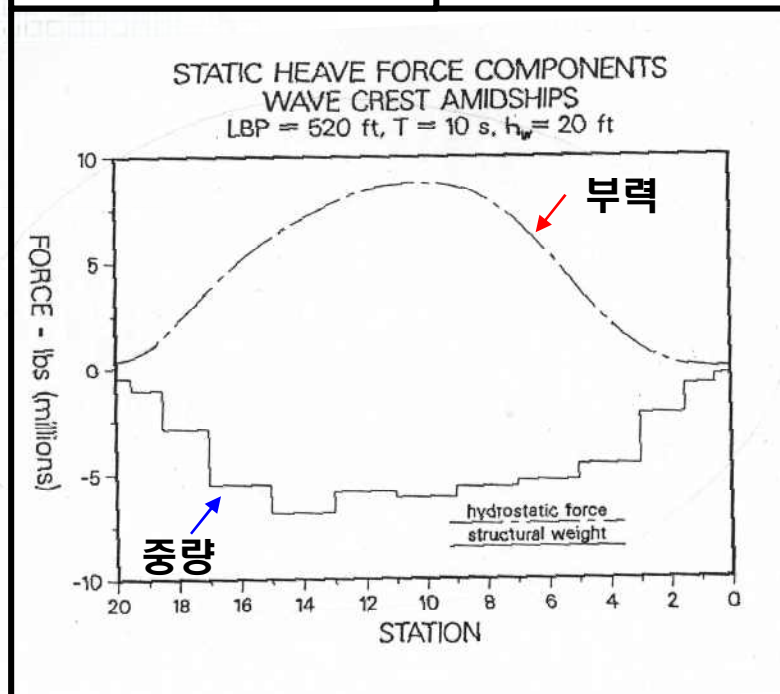
Part II. Ship Motion & Wave Load

: Term project (II) – Wave Load에 의한 VWBM¹⁾ 계산

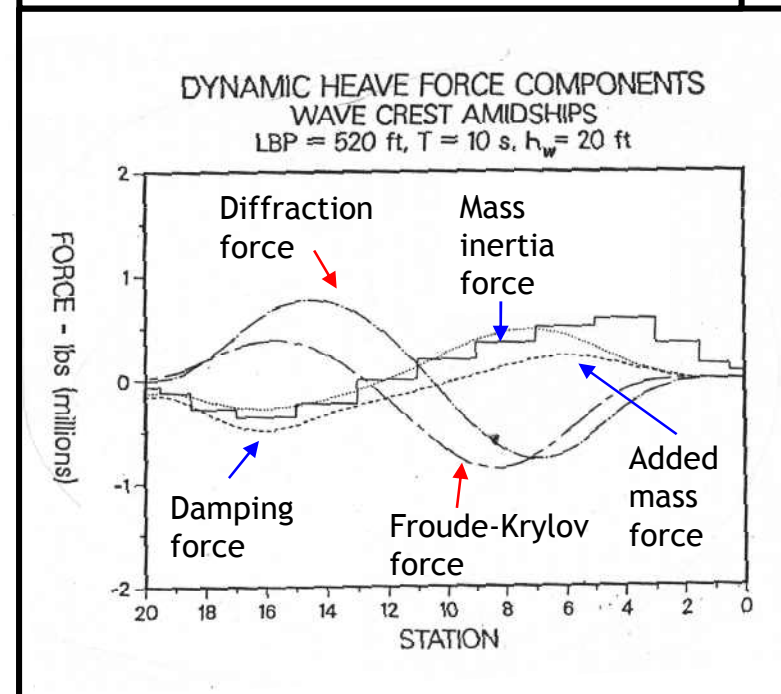
- 1) VWBM : Vertical Wave Bending Moment
- 2) SWBM : Still Water Bending Moment

- 6자유도 운동 방정식의 풀이 결과를 사용하여 선박에 작용하는 Wave Load를 계산하고, Shear force 및 Bending moment를 계산한다.

중량-부력 하중 곡선



Wave와 선박 운동에 의한 하중 곡선



Part II. Ship Motion & Wave Load

: 참고 자료

■ 참고 서적 및 자료

- 1) 小山健夫 著, 구종도 역, "선박과 해양구조물의 운동학", 연건문화사, 1997
- 2) Bhattacharyya, R. , "Dynamics of Marine Vehicles", John Wiley & Sons, 1978
- 3) Faltinsen, O.M. , "Sea loads on ships and offshore structures", Cambridge Univ. Press, 1998
- 4) Dean, R.G. , "Water wav mechanics for engineers and scientists", Prentice-Hall, Inc , 1984
- 5) Newman, J.N. , "Marine Hydrodynamics", The MIT Press, Cambridge, 1997
- 6) 이승건, "선박운동 조종론", 부산대학교 출판부, 2004
- 7) Journee, J.M.J. , Massie, W.W. , "Offshore Hydrodynamics", Delft University of Technology, 2001 (<http://www.shipmotions.nl/index.html>)
- 8) Journee, J.M.J. , Adegeest, L.J.M. , "Theoretical Manual of Strip Theory program“ Seaway for Windows”", Delft University of Technology, 2003 (<http://www.shipmotions.nl/index.html>)
- 9) Tommy Pedersen, "Wave Load Prediction - a Design Tool", PhD thesis, Department of naval architecture and offshore engineering, 2000
- 10) Cengel & Cimbala, "Fluid Mechanics", Mc Graw Hill, 2005
- 11) Erwin Kreyszig, "Advanced Engineering Mathematics", Wiley, 2005
- 12) Falnes, J. , "Ocean waves and oscillating systems", Cambridge Univ. Press, 2002



Term Project #4

Ship motion simulation

프로그램 소개

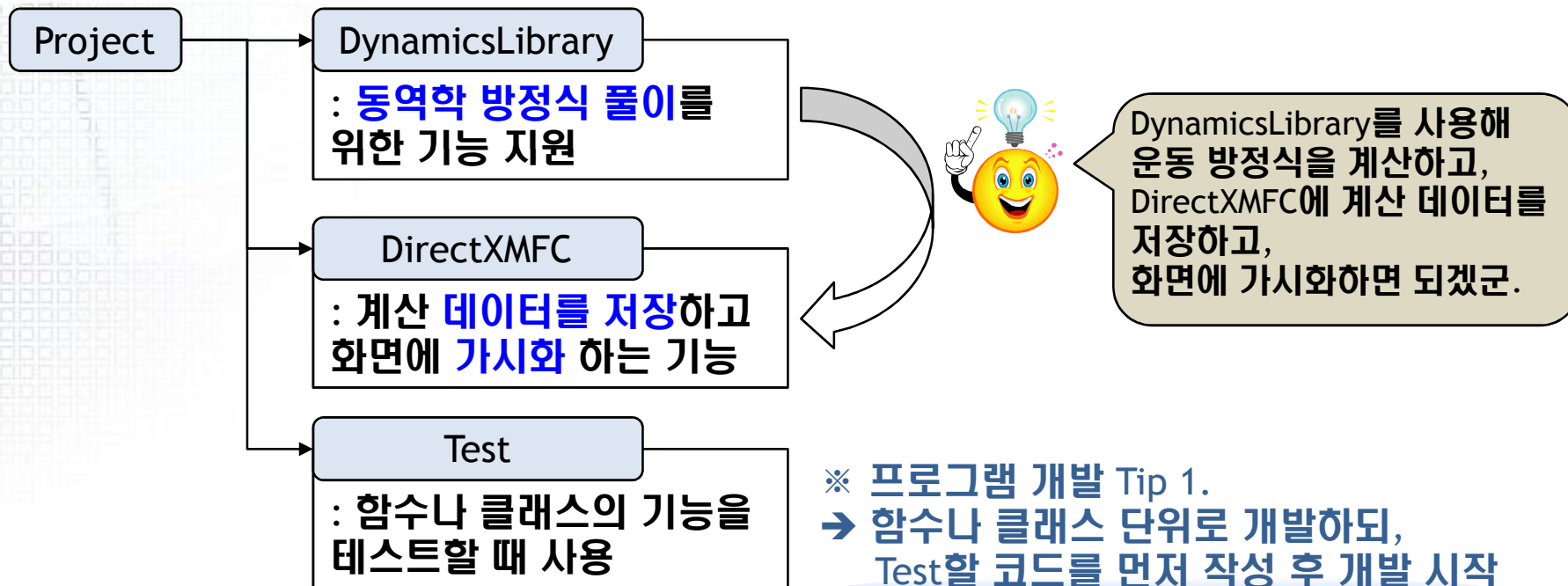
Advanced
Ship
Design
Automation
Laboratory

Term Project#4 Ship motion simulation 프로그램 소개

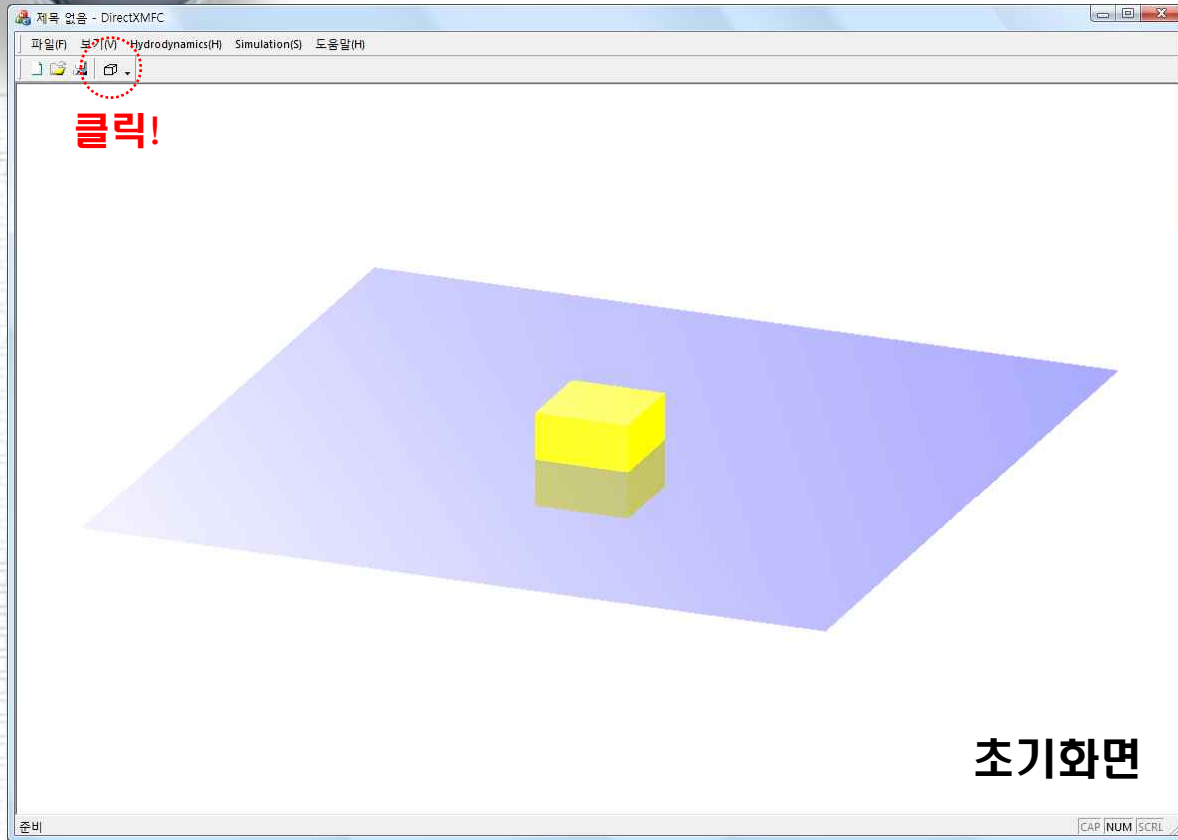
프로그램 실행 환경 및 프로젝트의 구성

- ✓ 프로그램 실행 환경 : Visual Studio 2008 (**반드시 Service pack 1을 설치해야 함**)
DirectX SDK 9.0 이상 버전

✓ 프로젝트의 구성

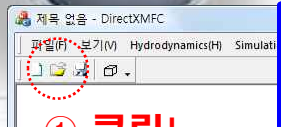


Term Project#4 Ship motion simulation 프로그램 소개

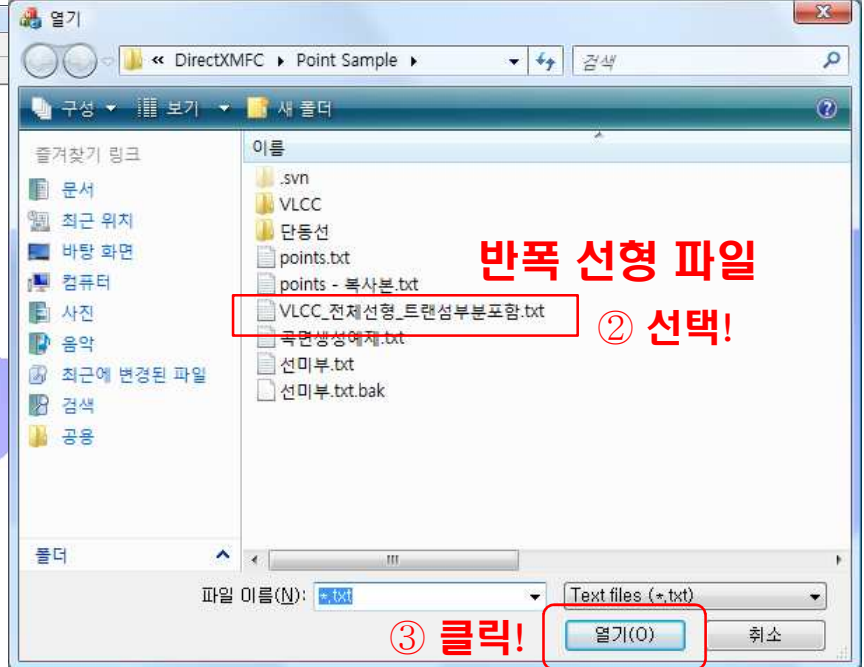
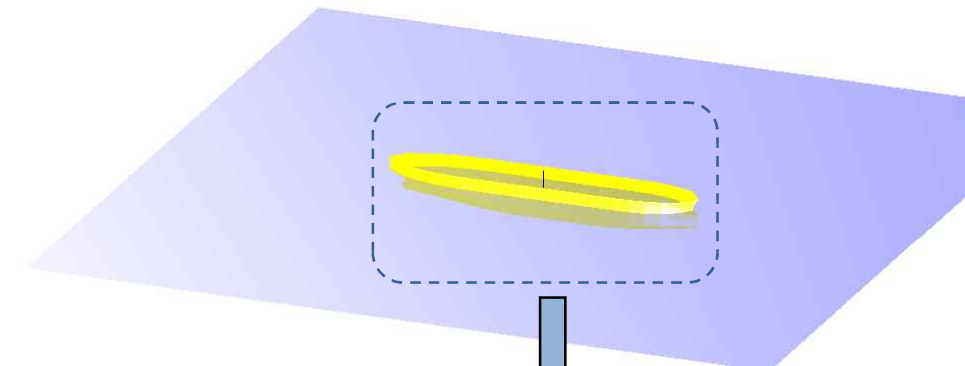


Term Project#4 Ship motion simulation 프로그램 소개

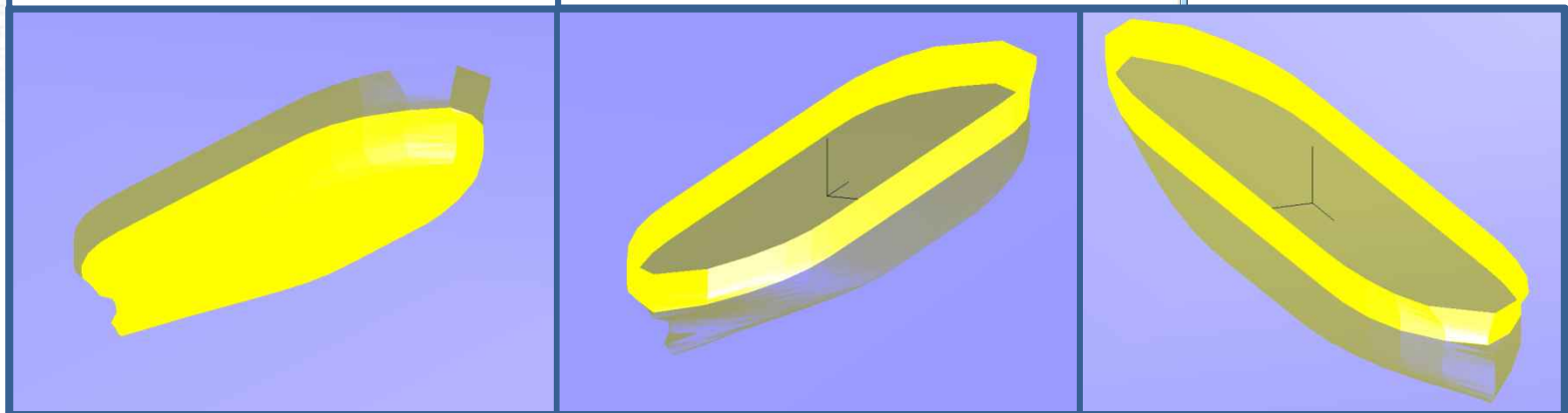
: 선박 형상 가시화



✓ Term Project #1,#2의 곡선,곡면 모델링 결과를 사용함

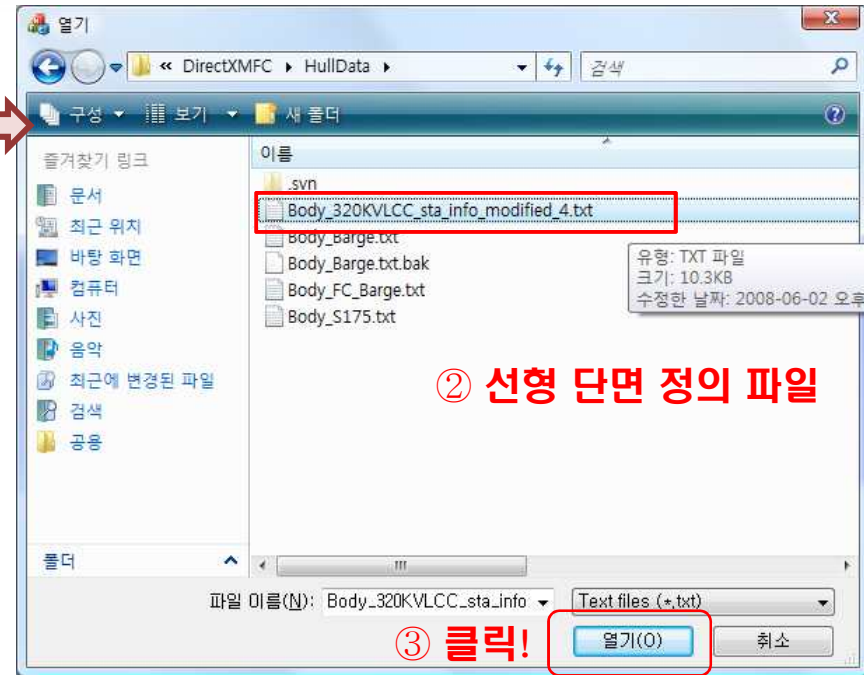
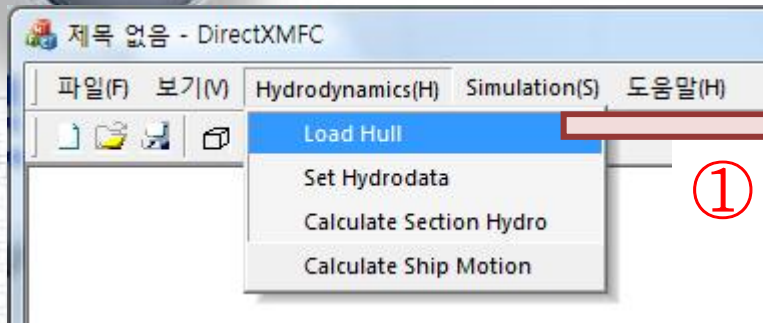


전쪽으로 확장하여 가시화



Term Project#4 Ship motion simulation 프로그램 소개

: 유체력 계수 계산 및 선박의 운동 계산



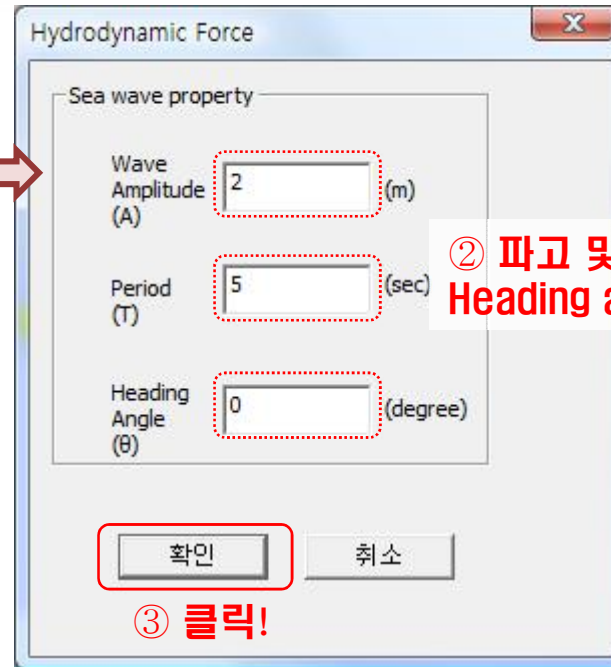
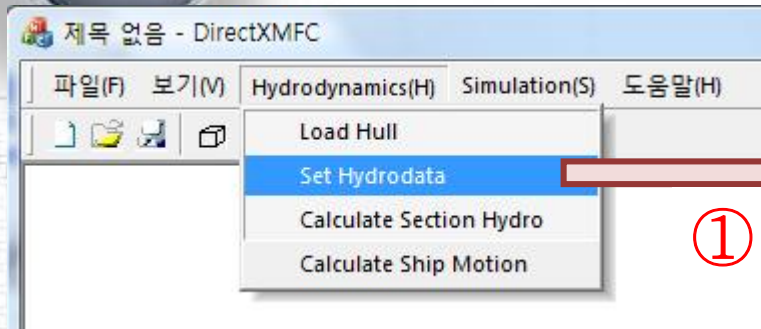
✓ 2차원 단면에 대한 유체력을 계산하기 위해 Station별 선형 정의 파일을 연다

② 선형 단면 정의 파일

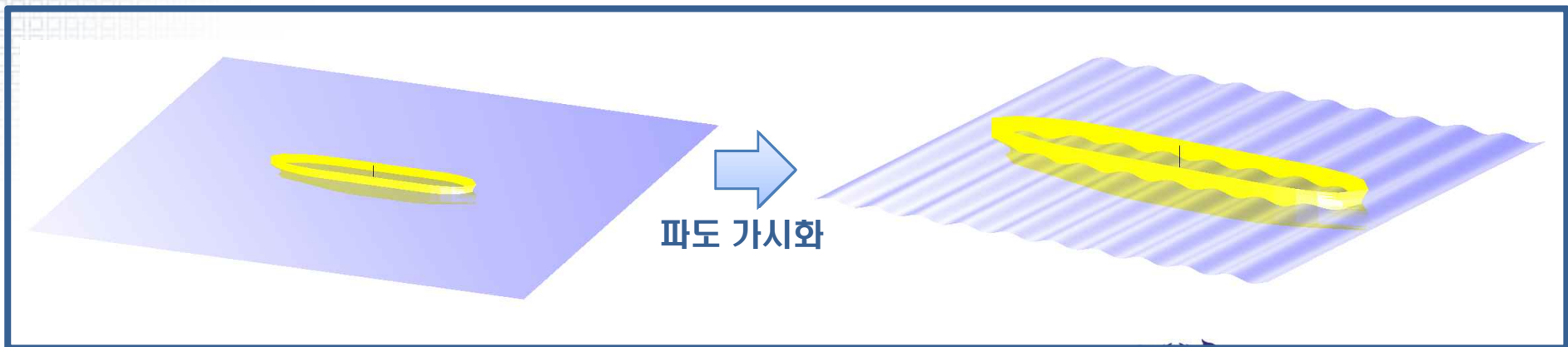
③ 클릭!

Term Project#4 Ship motion simulation 프로그램 소개

: 유체력 계수 계산 및 선박의 운동 계산

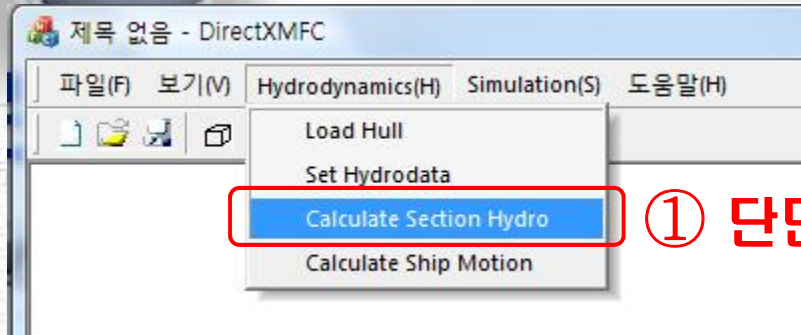


✓ 유체력 계산을 위해 파고, 파의 주기 Heading angle 입력

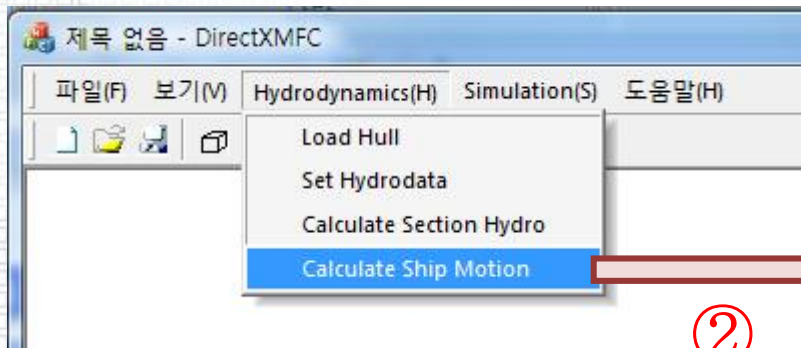


Term Project#4 Ship motion simulation 프로그램 소개

: 유체력 계수 계산 및 선박의 운동 계산

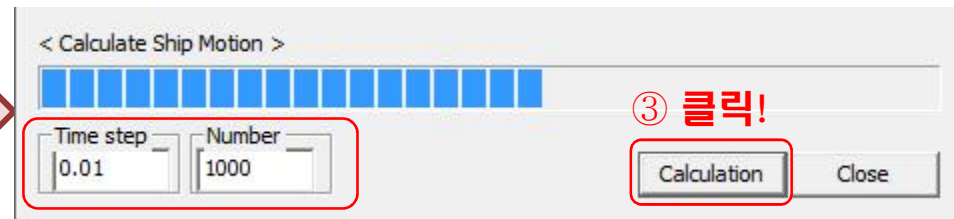


① 단면별 유체력 계수 계산



②

Progressbar에서 계산이 진행됨을 확인

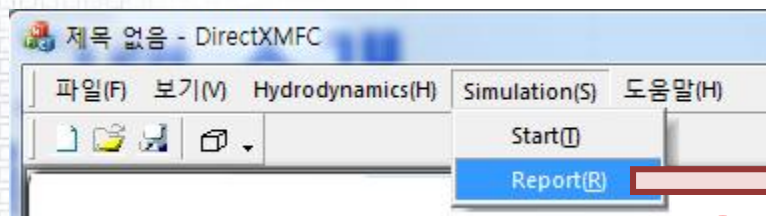


③ Simulation time step 및 계산 횟수 정의

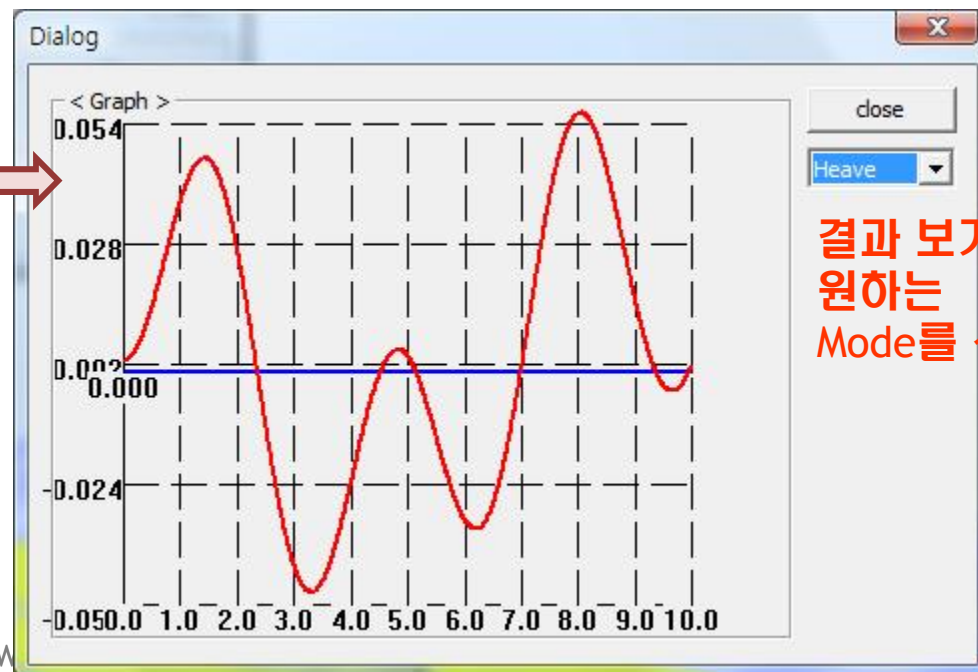
✓ 배의 운동 계산

Term Project#4 Ship motion simulation 프로그램 소개

: 시뮬레이션 가시화 및 결과 Report



①



결과 보기를
원하는
Mode를 선택



Term Project #4

Ship motion simulation

프로그램 과제

Term Project#4 Ship motion simulation 프로그램 과제

과제 1. CShipMotion 클래스에 equation 함수 및 CalcHydroCoeffAndExciting 함수 구현

과제 2. Gaussian quadrature 적분 방법을 이용한 면적 적분 함수 CalcArea 구현

과제 3. 선박 운동을 Runge-Kutta 방법을 사용하여 수치 적분하기 위해
CIMRunge-Kutta 클래스의 Integration 함수 구현

과제 4. 메뉴의 Simulation -> Report 클릭 시 그래프로 결과를 출력하는 Dialog 창 만들기

과제 5. Dialog 창을 만들 때, 모달리스(Modeless)로 만들기

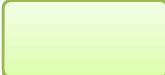
과제 6. 가시화할 때, 선박의 자세 값을 화면에 나타냄

과제 7. 파고와 주기를 입력하여 해수면을 가시화하고 시간에 따라 운동시킴

과제 8. Dialog 창을 사용하여 선박의 초기 위치를 설정

과제 9. 선박의 Deck 부분도 가시화

과제별 세부 내용은
이론 설명 다음에 언급함

 : 프로그램 내에 함수 제공
기본 점수 배점

 : 추가 구현할 부분 (함수 제공 안함)
구현 시 추가점수 부여

Term Project#4 Ship motion simulation 프로그램 과제

- 과제 5~9의 경우, 과제 1~4를 완료한 이후 진행한다.
- 과제 5~9는 이외에도 개인이 창의적으로 개발한 내용이 있을 시에는 추가 점수로 인정을 받는다.
[ex 선박 또는 해수면의 색상 변경, 광원 추가 등 DirectX를 추가로 사용할 경우 등]
- 점수의 배점은 과제 1~4까지 완성했을 경우를 75로 하고, 과제 5~9를 완성할 경우 하나당 +5의 배점을 추가한다.
[배점의 경우 제출 상황을 봐서 달라질 수 있음]
- 보고서는 표지를 포함하여 5~6 page로 작성하고, 자신이 직접 개발했다는 것을 표현할 수 있도록 한다.
[단순히 코드만 붙여 넣으면 카피로 간주함]

Term Project#4 Ship motion simulation 프로그램 과제

과제 1. CShipMotion 클래스에 equation 함수 및 CalcHydroCoeffAndExciting 함수 구현

→ DirectXMF C 프로젝트의 ShipMotion.cpp에 정의되어 있는 다음 두 함수를 구현 한다.

// ShipMotion.cpp 파일

```
void CShipMotion::Equation( double t,Matrix motc,Matrix velc,Matrix* accn )
{
    // x''=(M+A)-1 * (Fext -Bx' -Cx) 구현
}
void CShipMotion::CalcHydroCoeffAndExciting( CSectionHydro* section )
{
    // Gaussian Quadrature 적분을 통해 다음 계수를구함

    // 구해야할값
    double A22,A24,A26,A33,A35,A44,A46,A55,A66;
    double B22,B24,B26,B33,B35,B44,B46,B55,B66;
    double C33,C44,C55;
    Complex F2,F3,F4,F5,F6;
    m_WaveAmp = section->waveAmp;
    m_Freq = section->freq;
    ...

    // M,A,B,C,Fext Matrix의 성분을 채워줌
}
```

Term Project#4 Ship motion simulation 프로그램 과제

과제 1. CShipMotion 클래스에 equation 함수 및 CalcHydroCoeffAndExciting 함수 구현

→ A22 구하는 과정

```
void CShipMotion::CalcHydroCoeffAndExciting( CSectionHydro* section )
```

```
{
```

```
    // Gaussian Quadrature 적분을통해각계수를구함
```

```
    // 구해야할값
```

```
    double A22,A24,A26,A33,A35,A44,A46,A55,A66;
```

```
    double B22,B24,B26,B33,B35,B44,B46,B55,B66;
```

```
    double C33,C44,C55;
```

```
    Complex F2,F3,F4,F5,F6;
```

```
    m_WaveAmp = section->waveAmp;
```

```
    m_Freq = section->freq;
```

```
    Vector* tmpVec = new Vector[section->m_NumOfStation]; ← Station 개수만큼 Vector를 동적 할당
```

```
    // A22
```

```
    for (int i=0;i<section->m_NumOfStation;i++)
```

```
    {
```

```
        double x = section->SectionHydro[i]->m_X;
```

```
        tmpVec[i].Set(x,section->SectionHydro[i]->CallFunc(0),0);
```

```
    }
```

```
    A22 = CalcArea(tmpVec,section->m_NumOfStation);
```

```
    ...
```

```
}
```

global.cpp에 정의된 함수 - Gaussian Quadrature로 적분 수행(과제2)

Term Project#4 Ship motion simulation 프로그램 과제

과제 1. CShipMotion 클래스에 equation 함수 및 CalcHydroCoeffAndExciting 함수 구현

→ 단면별로 계산된 유체력 계수(added mass, damping coefficient), Froude-Krylov force Diffraction force를 받아오는 방법

※ 계산 결과 받아오는 방법

(ex) i번째 station의 a22 값 : $a22 = \text{section} \rightarrow \text{SectionHydro}[i] \rightarrow \text{CallFunc}(0);$

CallFunc에서 리턴해주는 값

CallFunc(0) → a22; CallFunc(1) → a24; CallFunc(2) → a33; CallFunc(3) = a44

CallFunc(4) → b22; CallFunc(5) → b24; CallFunc(6) → b33; CallFunc(7) = b44

CallFunc(8) → f2_real; CallFunc(9) → f3_real; CallFunc(10) → f4_real;

CallFunc(11) → h2_real; CallFunc(12) → h3_real; CallFunc(13) → h4_real;

CallFunc(14) → f2_image; CallFunc(15) → f3_image; CallFunc(16) → f4_image;

CallFunc(17) → h2_image; CallFunc(18) → h3_image; CallFunc(19) → h4_image;

f2, f3, f4 : 단면의 sway, heave, roll 방향의 Froude-Krylove force (Complex)

h2, h3, h4 : 단면의 sway, heave, roll 방향의 Diffraction force (Complex)

※ 각 계수별 적분하는 방법은 18~20 page를 참고할 것

Term Project#4 Ship motion simulation 프로그램 과제

(x_C, y_C, z_C) : 선박의 무게 중심의 좌표 (from body-fixed coordinate)

과제 1. CShipMotion 클래스에 equation 함수 및 CalHydroCoeffAndExciting 함수 구현

→ CalHydroCoeffAndExciting 함수에서 다음 매트릭스의 성분을 모두 채워 줌

$$\mathbf{M} = \begin{bmatrix} M & 0 & 0 & 0 & Mz_C & -My_C \\ 0 & M & 0 & -Mz_C & 0 & Mx_C \\ 0 & 0 & M & My_C & -Mx_C & 0 \\ 0 & -Mz_C & My_C & I_{xx} & 0 & -I_{xz} \\ Mz_C & 0 & -Mx_C & 0 & I_{yy} & 0 \\ -My_C & Mx_C & 0 & -I_{zx} & 0 & I_{zz} \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} B_{11} & 0 & B_{13} & 0 & B_{15} & 0 \\ 0 & B_{22} & 0 & B_{24} & 0 & B_{26} \\ B_{31} & 0 & B_{33} & 0 & B_{35} & 0 \\ 0 & B_{42} & 0 & B_{44} & 0 & B_{46} \\ B_{51} & 0 & B_{53} & 0 & B_{55} & 0 \\ 0 & B_{62} & 0 & B_{64} & 0 & B_{66} \end{bmatrix}$$

$$(\mathbf{M} + \mathbf{A})\ddot{\mathbf{x}} + \mathbf{B}\dot{\mathbf{x}} + \mathbf{C}\mathbf{x} = \mathbf{F}_{exciting}$$

$$\mathbf{F}_{exciting} = \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \\ F_5 \\ F_6 \end{bmatrix}$$

$$\mathbf{A} = \begin{bmatrix} A_{11} & 0 & A_{13} & 0 & A_{15} & 0 \\ 0 & A_{22} & 0 & A_{24} & 0 & A_{26} \\ A_{31} & 0 & A_{33} & 0 & A_{35} & 0 \\ 0 & A_{42} & 0 & A_{44} & 0 & A_{46} \\ A_{51} & 0 & A_{53} & 0 & A_{55} & 0 \\ 0 & A_{62} & 0 & A_{64} & 0 & A_{66} \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & C_{33} & 0 & C_{35} & 0 \\ 0 & 0 & 0 & C_{44} & 0 & 0 \\ 0 & 0 & C_{53} & 0 & C_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

surge	Heave	Pitch
↓	↓	↓
$\mathbf{x} = [\xi_1, \xi_2, \xi_3, \xi_4, \xi_5, \xi_6]^T$		
↑	↑	↑
Sway	Roll	Yaw

Term Project#4 Ship motion simulation 프로그램 과제

과제 1. CShipMotion 클래스에 equation 함수 및 CalcHydroCoeffAndExciting 함수 구현

DRAFT MOULDED(M)	19.983
DRAFT EXTREME(M)	20.000
TRIM(M)	0.000
HEEL(DEG.)	0.000
VOLUME MOULDED	321512.2
DISPL. TOT. S.W.	329986.6
DISPL. TOT. F.W.	321938.2
DISPL. AFT. BODY	150089.9
DISPL. FORE BODY	179460.1
L.C.F. FROM MIDSHIP	1.021
L.C.B. FROM MIDSHIP	11.142
L.C.B. AFT. BODY	-63.919
L.C.B. FORE BODY	73.919
V.C.B. ABOVE B.L.	10.348
T.P.C.(TON/1CM)	179.9
M.T.C. 0.01M(T-M)	4148.7
K.M.T.(M)	25.169
K.M.L.(M)	412.664
I LONG.*1/100(M4)	1295207.6
I TRAN.*1/100(M4)	47716.2
W.P.A. MOULDED(M2)	17554.7
WETTED SURFACE(M2)	28398.8
BLOCK COEFF.(Cb)	0.8380
PRISM. COEFF.(Cp)	0.8401
W.P.A. COEFF.(Cw)	0.9143
MIDSH. COEFF.(Cm)	0.9975

✓ 운동 방정식을 세우는데 필요한 VLCC의 주요 제원 및 Hydrostatic 계산결과

L = 320 [m]
 B = 60 [m]
 D = 30 [m]
 T = 20 [m]
 질량 M = 329986.6 [ton]
 LCG = 11.142 [m]
 KG = 17 [m]
 밀도 = 1.025 [ton/m³]

※ 질량관성모멘트(I_{xx}, I_{yy}, I_{zz})는 선박을 직육면체로 가정하여 구함
 $C_{35}, C_{53}, I_{xz}, I_{zx}$ 는 0으로 가정함

← EzCompart에서 계산된 VLCC의 Hydrostatic table 중 발췌

Term Project#4 Ship motion simulation 프로그램 과제

과제 2. Gaussian quadrature 적분 방법을 이용한 면적 적분 함수 CalcArea 구현

→ DirectXMFC 프로젝트의 global.cpp 파일에 있는 CalcArea 함수 내부를 구현함

```
// global.cpp
```

```
// Gaussian Quadrature를 사용하여 xy 평면의 면적을 계산해줌.
```

```
double CalcArea(Vector* point, int num)
```

```
{
```

```
    double area = 0;
```

```
    double tmp_area = 0;
```

```
    ...
```

구현

```
    return area;
```

```
}
```

Term Project#4 Ship motion simulation 프로그램 과제

과제 3. CIMRunge-Kutta 클래스의 Integration 함수 구현

→ Rungge-Kutta 방법을 사용하여 적분을 수행하기 위해 DirectXMFC 프로젝트의 IMRungeKutta.cpp 파일에 있는 Integration 함수를 구현한다.

```
// 현재단계: n 현재시간: t
// 다음단계: n+1          다음시간: t+m_TimeStep
void CIMRungeKutta::Integration( CEquationOfMotion* moe, int n, double time_step )
{
    // Using Runge-Kutta Scheme
    if (moe->m_Mot.size() != n+1)
    {
        printf("Count Error in Integral method!!");
        ASSERT(0);
        return;
    }

    // 1st step
    ...
    // 2nd step
    ...
    // 3rd step
    ...
    // 4th step
    ...

    moe->SetPosition(motn);
    moe->SetVelocity(veln);
    moe->SetAcceleration(accn);
}
```


Term Project#4 Ship motion simulation 프로그램 과제

과제 3. CIMRunge-Kutta 클래스의 Integration 함수 구현

→ Integral하는 방법을 바꾸기 위해서는 DirectXMFCDoc.cpp에 있는 Initialize 함수의 다음 표시된 부분을 수정해 준다.

// DirectXMFCDoc.cpp 파일

```
void CDirectXMFCDoc::Initialize()  
{
```

```
    // 동역학계산을위한초기화과정
```

```
    m_Simulation = new CSimulation;  
    m_Simulation->SetIntegralMethod(new CIMRungeKutta);
```

```
    m_ShipMotion = new CShipMotion;  
    m_ShipMotion->Initialize(6,Matrix(6,1),Matrix(6,1),Matrix(6,1));    // 초기값으로모두0을입력, 변수는6개
```

```
    m_ShipMotion->SaveMotion("motion.txt");    // 변위,속도,가속도저장  
    m_ShipMotion->SaveVelocity("velocity.txt");  
    m_ShipMotion->SaveAcceleration("acceleration.txt");
```

```
    m_Simulation->SetMotionEqation(m_ShipMotion);  
}
```

Integral method를 설정해 주는 부분

Cf) Euler method 사용 시
m_Simulation->SetIntegralMethod(new CIMEuler);

Term Project#4 Ship motion simulation 프로그램 과제

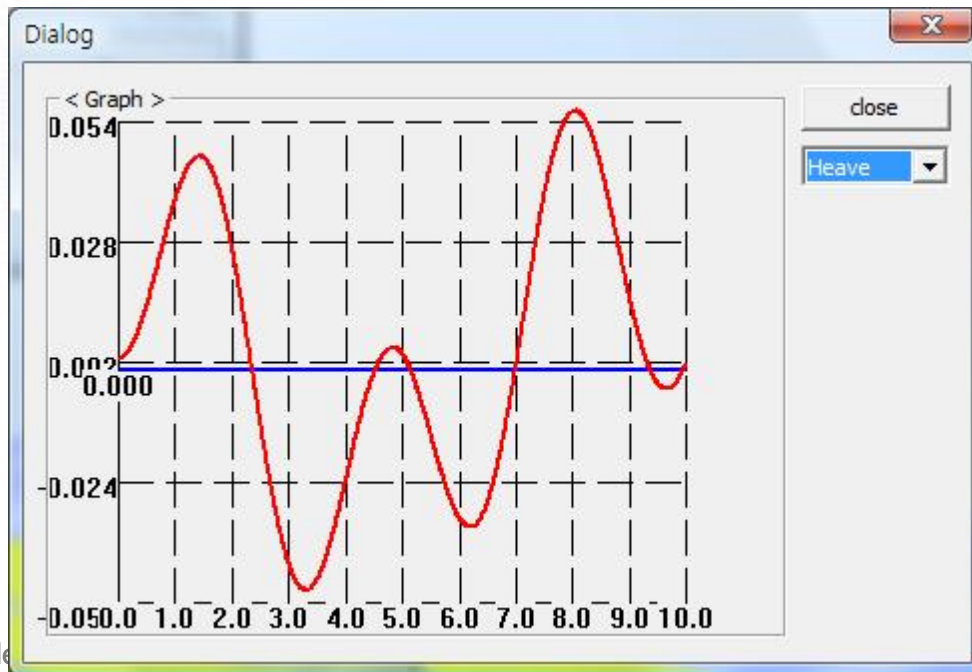
과제 4. 메뉴의 Simulation -> Report 클릭 시 그래프로 결과를 출력하는 Dialog 창 만들기

→ DirectXMFVC 프로젝트의 DirectXMFCDoc.cpp 파일에 있는 OnSimulationReport() 함수를 구현한다.

```
void CDirectXMFCDoc::OnSimulationReport()
{
    // TODO: 여기에 명령처리기코드를추가합니다.

    // Modal 또는 Modeless 대화상자를 만들어 결과를 Report 한다.
}
```

(ex)



Modeless로 만들 경우,
과제 4,5를 동시에 수행한
것으로 간주함