

범용 동역학 모듈과 가시화 모듈을 이용한 조선 블록 탑재 시뮬레이션

차주환*, 노명일**, 이규열***

Block Erection Simulation in Shipbuilding Using the Open Dynamics Module and Graphics Module

Ju-Hwan Cha*, Myung-IJ Roh** and Kyu-Yeul Lee***

ABSTRACT

The development of a simulation system requires many sub modules such as a dynamic module, a visualization module, etc. If a different freeware is used for each sub modules, it is hard to develop the simulation system by incorporating them because they use their own data structures. To solve this problem, a high-level data structure, called Dynamics Scene Graph Data structure (DSGD) is proposed, by wrapping data structures of two freeware; an Open Dynamics Engine (ODE) for the dynamic module and an Open Scene Graph (OSG) for the visualization module. Finally, to evaluate the applicability of the proposed data structure, it is applied to the block erection simulation in shipbuilding. The result shows that it can be used for developing the simulation system.

Key words : Modeling and Simulation(M&S), Dynamics Module, Graphics Module, Block Erection Simulation, Shipbuilding Process Simulation

1. 서 론

조선소에서는 해양 구조물(offshore structure)의 상부(topside) 블록(block)을 탑재하거나 턴 오버(turn-over) 작업(Fig. 1 참조)을 할 경우, 대상 블록 또는 장비와 선행 탑재된 블록 또는 장비의 복잡성으로 인해 Fig. 2와 같은 간섭(interference)이 빈번히 발생되고 있다. 이와 같은 간섭에 대해 현재는 현장에서의 사후 조치 중심으로 이루어지고 있다. 따라서 재 작업에 따른 자재 및 인력의 추가 투입, 그리고 후행 작업 증가로 인한 생산성 저하가 발생하고 있다. 만약 시뮬레이션을 통해 해양 구조물 상부 블록의 탑재 및 턴 오버 시 발생할 수 있는 간섭 문제를 사전에 파악할 수 있다면 위와 같은 시간 및 비용에서의 추가 손실을

최소화 할 수 있을 것이다.

이처럼 조선 공정에서의 다양한 시뮬레이션 요구에 부응하기 위해 현재 많은 시뮬레이션 시스템이 개발되고 있다. 이를 개발하기 위해서는 많은 요소 모듈들이 필요하다. 예컨대, 물체간 간섭 여부를 검사하고 간섭 후의 자세 변화를 계산하기 위해 동역학 모듈이 필요하고, 시뮬레이션 과정을 가시화하기 위해 가시화 모듈이 필요하다. 시뮬레이션 시스템 개발 시 이러한 요소 모듈들은 직접 개발할 수도 있고, 개발 시간을 단축하기 위해 공개된 모듈들을 활용할 수도 있다. 본 저자들은 후자의 방법 즉, 범용 동역학 모듈과 가시화 모듈을 이용하여 시뮬레이션 시스템을 개발하고자 하였다. 현재까지 개발된 공개용 동역학 모듈로는 Open Dynamics Engine (ODE)^[1], Wild Magic^[2], DynaMechs^[3], DsTool^[4] 등이 있고, 공개용 가시화 모듈로는 Open Scene Graph (OSG)^[5], Visualization Library^[6], OGRE 3D^[7] 등이 있다. 각 모듈에 대한 기능(강제 및 탄성체에 대한 계산 가능 여부, 사용 편리성 등) 및 성능(계산 속도 등) 테스트 결과, 동역학 모듈로는 ODE를, 가시화 모듈로는 OSG를 활용하기

*정회원, 서울대학교 공학연구소

**교신저자, 정회원, 울산대학교 조선해양공학부

***중산회원, 서울대학교 조선해양공학과 및 해양시스템 공학연구소

- 논문투고일: 2007. 07. 31

- 논문수정일: 2008. 06. 21

- 심사완료일: 2008. 07. 04

로 결정하였다. 하지만 각각의 모듈이 독자적으로 개발된 것이라 통합하여 사용하는 데에는 한계점이 존재하였다. 예컨대 블록 탑재 시 간섭 검사의 결과는 블록을 구성하는 수많은 부재 중 간섭을 일으킨 부재가 무엇이고, 어떤 상위 부재에 소속된 것이며, 어느 위치에 있는 것인지를 알려주는 것이다. 하지만 동역학 모듈인 ODE는 모든 부재들을 병렬적인 연결 관계를 가지고 표현하기 때문에 간섭을 일으킨 부재가 어디에 속해 있는 부재인지를 확인하기가 어렵다. 다행히 가시화 모듈인 OSG는 모든 부재들을 장면 그래프(scene graph)라는 것을 이용하여 계층적으로 표현한다. 따라서 간섭 검사, 간섭 후 자세 등과 같은 동역학 계산과 가시화를 유기적으로 하기 위해서는 이 두 모듈의 자료 구조를 통합한 새로운 자료 구조가 필요하였다.

따라서 본 연구에서는 평행적 연결 관계를 가지고 있는 동역학 모듈 ODE와 계층적 연결 관계를 가지고 있는 가시화 모듈 OSG를 효과적으로 연동할 수 있는 새로운 자료 구조인 DSGD(Dynamics Scene Graph Data structure)를 제안하였다. 또한 상용 그래픽 모델링 시스템 3D Studio MAX로부터 자동으로 DSGD를 생성할 수 있는 방법에 대해서도 연구하였다. 마지막으로 본 연구에서 제안한 DSGD의 효율성을 검증하기 위해, 조선 공정 시뮬레이션 중 하나인 블록 탑재 시뮬레이션에 적용하였다.

본 논문의 구성을 간략히 설명하면, 2장에서는 동역학 모듈 ODE와 가시화 모듈 OSG의 자료 구조에 대해 간략하게 설명하고, 3장에서는 본 연구에서 제안한 DSGD에 대해 기술한다. 4장에서는 이를 블록 탑재 시뮬레이션에 적용한 예를 보여주고, 마지막으로 5장에서는 결론 및 향후 연구 계획을 기술한다.

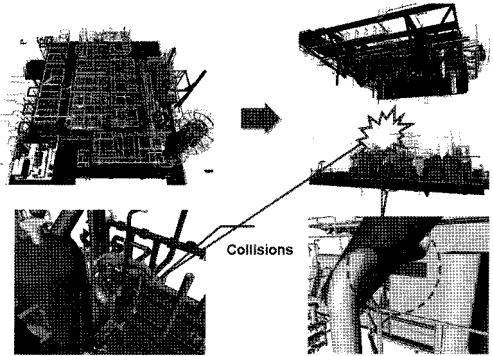


Fig. 2. 해양 구조물 블록간의 간섭 발생 예.

2. 범용 동역학 모듈과 그래픽스 모듈의 소개

2.1 동역학 모듈 Open Dynamics Engine (ODE)의 자료 구조

본 연구의 적용 대상이 조선 공정 시뮬레이션 중 하나인 블록 탑재 시뮬레이션이기 때문에, 본 연구에서는 선박의 블록을 예로 들어 각 모듈의 자료 구조를 설명하기로 한다.

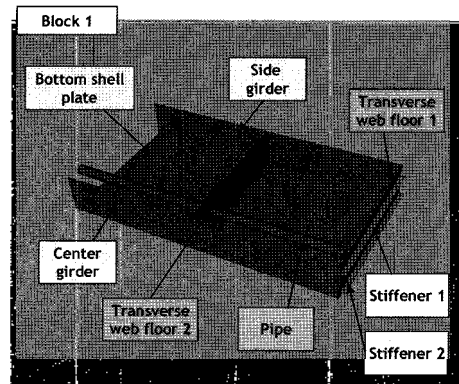
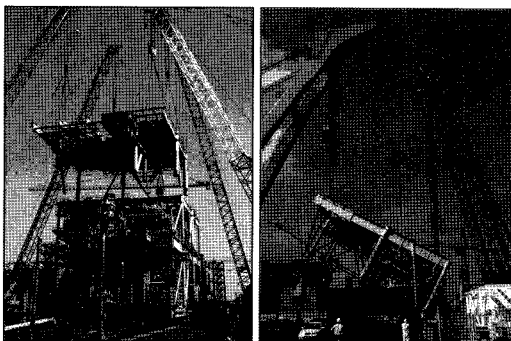


Fig. 3. 선박의 단순화 된 블록의 형상 예시.

선박의 단순화 된 블록의 형상을 Fig. 3과 같이 정의하였다. 선박의 단순화 된 블록은 총 8개의 부재로 이루어져 있다. 두 개의 거더(girder) ‘Center girder’, ‘Side girder’와 두 개의 웹 플로어(transverse web floor) ‘Transverse web floor 1’, ‘Transverse web floor 2’, 이 중에서 ‘Transverse web floor 1’에 두 개의 보강재(stiffener) ‘Stiffener 1’, ‘Stiffener 2’가 부착되어 있다. 그리고 ‘Bottom shell plate’와 ‘Transverse web floor 2’를 관통하는 하나의 파이프(pipe) ‘Pipe’가 있다.



해상 Topside 탑재 작업 해상 Topside Turn-Over 작업

Fig. 1. 해양 구조물의 상부 블록의 탑재 작업 및 턴오버 작업의 예.

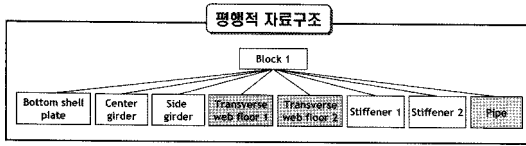


Fig. 4. 선박의 단순화 된 블록의 평행적 자료 구조.

이를 평행적 자료 구조로 나타내면 Fig. 4와 같다. 즉, 모든 부재가 상하위 관계 없이 모두 평행적으로 나열되어 있는 구조이다. 동역학 모듈 ODE는 이러한 평행적 자료 구조를 가지고 있다. 선박의 단순화 된 블록을 동역학 모듈 ODE의 자체적인 평행적 자료 구조로 나타내면 Fig. 5와 같다.

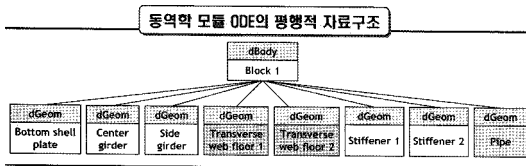


Fig. 5. 동역학 모듈 ODE의 평행적 자료 구조로 나타낸 선박의 단순화 된 블록.

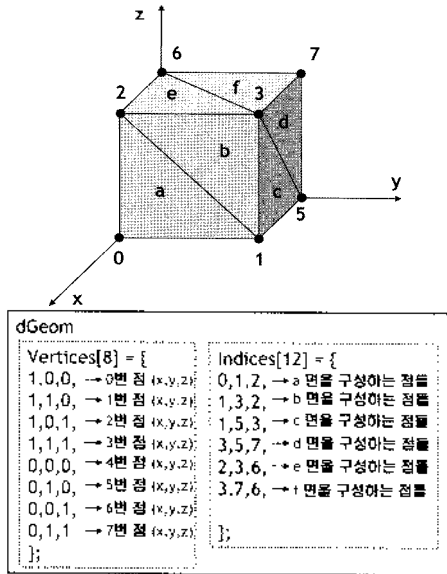


Fig. 6. 동역학 모듈 ODE의 자료 구조 중 하나인 'dGeom'의 직육면체에 대한 예.

동역학 모듈 ODE의 자료 구조는 크게 'dBody'와 'dGeom'의 두 가지로 구성되어 있다. 'dBody'는 독립된 물체를 표현하는 추상적인 것이고, 'dGeom'은 형상을 표현하는 메쉬 정보(메쉬 선들과 이들간의 연결 관계)를 저장하는 것이다^[8]. 여기에서 독립된 물체라는 것은 따로 떨어져서 움직일 수 있는 것을 말한

다. 선박의 단순화 된 블록의 예를 들면, 하나의 블록은 독립된 물체인데, 블록의 형상은 여러 부재의 형상이 합쳐진 것이다. 따라서 'Block 1'이라는 'dBody'에 'Bottom shell plate', 'Center girder' 등 8개 부재의 형상 정보를 담고 있는 8개 'dGeom'이 연결되어 있으면 된다. 'dGeom'의 직육면체에 대한 예가 Fig. 6에 나타나 있다.

2.2 가시화 모듈 Open Scene Graph (OSG)의 자료 구조

선박의 단순화 된 블록을 계층적 자료 구조로 나타내면 Fig. 7과 같다. 최상위에는 전체 블록을 나타내는 'Block1'이라는 노드(node)가 있고, 그 아래에 크게 'Bottom shell plate', 'Girders', 'Transverse web floors', 'Pipe'라는 하위 노드가 있다. 'Girders'는 'Center girder'와 'Side girder'를 묶는 그룹 노드이고, 'Transverse web floors'는 'Transverse web floor 1'과 'Transverse web floor 2'를 묶는 그룹 노드이다. 그리고 'Transverse web floor 1' 아래에는 'Stiffener 1'과 'Stiffener 2'를 묶는 그룹 노드인 'Stiffeners'가 있다. 이러한 계층적 자료 구조를 이용할 경우, 부재간의 상하위 관계들 명확하게 파악할 수 있다. 예를 들면, 'Stiffener 1'은 'Transverse web floor 1'에 속한 부재라는 것을 알 수 있다.

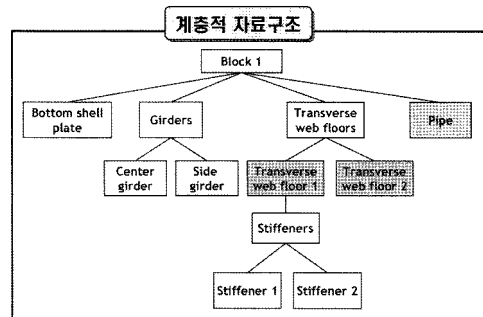


Fig. 7. 선박의 단순화 된 블록의 계층적 자료 구조.

가시화 모듈 OSG의 자료 구조는 위와 같은 계층적 자료 구조로서 Fig. 8은 가시화 모듈 OSG의 자료 구조를 나타낸 것이다.

가시화 모듈 OSG의 자료 구조는 크게 'osg::Group'과 'osg::Geode'의 두 가지로 구성되어 있다. 'osg::Group'은 계층 구조를 표현하기 위해 하위 노드를 묶어 놓은 추상적인 것이고, 'osg::Geode'는 형상을 표현하는 메쉬 정보를 저장하는 것이다. 'osg::Geode'는 동역학 모듈 ODE의 'dGeom'과 유사하게

메쉬 점들과 이들간의 연결 관계를 저장하고 있다(Fig. 6 참고).

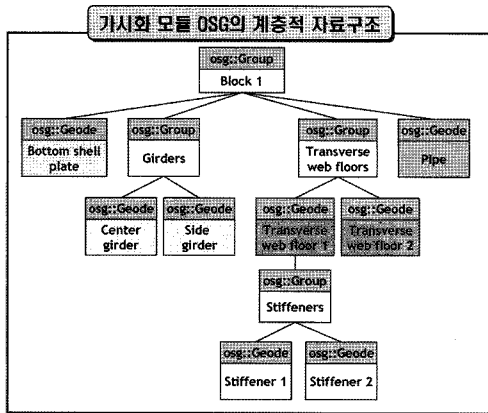


Fig. 8. 가시화 모듈 OSG의 계층적 자료 구조로 나타낸 선박의 단순화 된 블록.

3. Dynamics Scene Graph Data structure (DSGD)의 제안

3.1 동역학 모듈 ODE의 자료 구조와 가시화 모듈 OSG의 자료 구조 간의 연동에 대한 기본 아이디어

간접 검사 및 동역학 계산을 위해서는 동역학 모듈 ODE의 자료 구조가 필요하고, 계층적으로 표현하여 시뮬레이션 모델의 구성을 효과적으로 가시화하기 위해서는 가시화 모듈 OSG의 자료 구조가 필요하다. 그

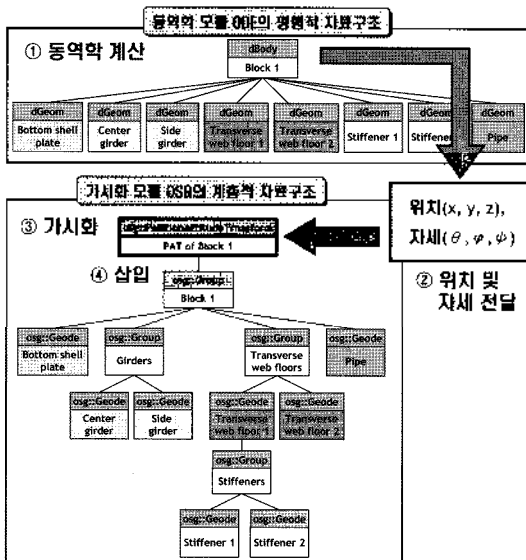


Fig. 9. 동역학 모듈 ODE 자료 구조와 가시화 모듈 OSG 자료 구조의 연동에 대한 기본 아이디어.

러나 두 모듈의 자료 구조의 형태가 다르기 때문에 이를 연동할 수 있는 방법이 필요하다.

연동에 대한 기본 아이디어는 다음과 같다. 우선, ODE에서 간접 검사 및 동역학 계산을 수행하면 대상 물체의 위치와 자세가 결정된다(Fig. 9 ①). 이 때 위치와 자세는 독립된 물체 단위로 구별되므로, 위치와 자세 정보는 'dBody' 내에 저장이 된다. 그 후에 이 위치와 자세 정보를 'OSG'에게 전달하여(Fig. 9 ②) 형상을 가시화 하도록 하는 것이다(Fig. 9 ③). 그런데 ODE의 'dBody'에서 전달된 위치와 자세 정보를 반영할 노드가 OSG 내에는 없다. 독립된 하나의 물체에서 위치와 자세 정보가 전달 되었기 때문에 각 부재의 형상 정보를 담고 있는 'osg::Geode'에 전달할 수도 없고, 전체를 묶는 최상위의 'osg::Group'은 묶어 주는 역할만 수행하기 때문에 위치와 자세 정보를 저장할 수 없다. 따라서 최상위의 'osg::Group' 위에 위치와 자세 정보를 저장하여 하위 노드에 반영할 수 있는 'osg::PositionAttitudeTransform'이라는 자료 구조를 삽입하여 사용한다(Fig. 9 ④).

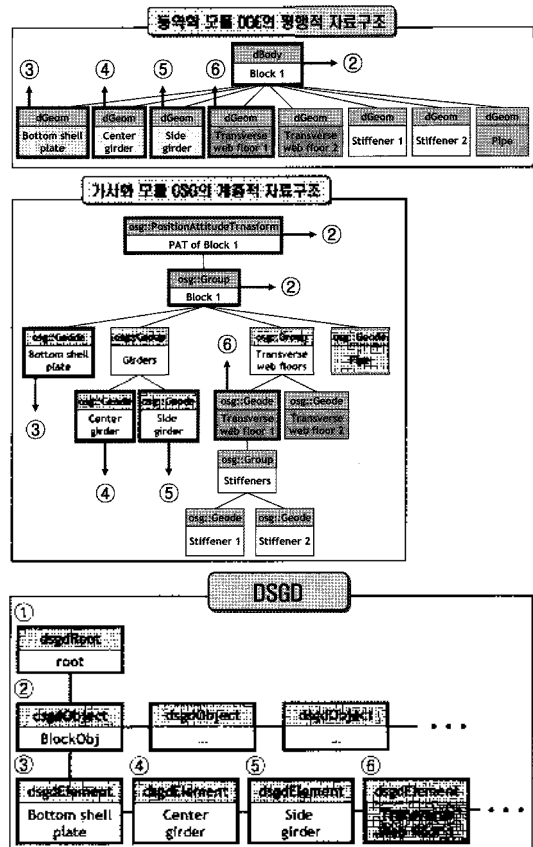


Fig. 10. 본 연구에서 제안한 DSGD의 구성 요소.

3.2 DSGD의 구성 요소

시도 다른 자료 구조를 가진 모듈들을 연동하는 방법으로는 여러 가지가 있으나, 본 연구에서는 이들을 감싸는 상위의 자료 구조를 생성하는 방법을 사용하였다. 본 연구에서는 이러한 상위의 자료 구조를 Dynamics Scene Graph Data structure(DSGD)라고 명명하였다.

DSGD는 크게 'dsgdObject', 'dsgdElement', 'dsgdRoot'의 세 부분으로 구성되어 있다. 우선, 위치와 자세 정보를 주고 받는 동역학 모듈 ODE의 'dBody'와 가시화 모듈 OSG의 'osg::PositionAttitude-Transform'을 묶어주는 'dsgdObject'라는 것이다(Fig. 10②). 'dsgdObject'에는 상징적으로 가시화 모듈의 최상위 그룹 노드를 가지고 있도록 하였다. 그리고 'dsgdObject' 하위에는 실제 각 부재의 형상 정보를 가지고 있는 ODE의 'dGeom'과 OSG의 'osg::Geode'를 포함하고 있는 'dsgdElement'들이 있다(Fig. 10③~⑥). 마지막으로 독립된 볼체가 여러 개일 수 있기 때문에 'dsgdObject'가 여러 개일 수 있고, 이를 전체적으로 총괄하는 'dsgdRoot'라는 것을 두었다(Fig. 10①).

3.3 DSGD를 통한 연동 알고리즘

DSGD를 통해 두 모듈의 자료 구조가 연동되는 알고리즘을 살펴보면 다음과 같다.

우선 시뮬레이션의 매 시간 간격마다 'dsgdRoot'는 하위의 'dsgdObject'들을 모두 호출하여 ODE로부터 계산된 위치와 자세 정보를 OSG로 넘겨주게 된다. 'dsgdElement'까지 호출할 필요가 없기 때문에 빠른 시간 내에 매 시간 가시화가 가능하다. 간섭이 일어났을 경우에는 ODE에서 간섭이 일어난 'dGeom'들을 알아낼 수 있고, 'dsgdElement'를 통해 간섭이 일어난 'dGeom'에 해당하는 OSG의 'osg::Geode'를 알아낼 수 있다. OSG의 자료 구조는 계층식 자료 구조이기 때문에 부재 간의 상하 관계를 알아내어 그 정보를 사용자에 표시할 수 있다.

3.4 DSGD 자동 생성 알고리즘

ODE도 OSG도 모두 형상 모델링 도구는 아니다. 따라서 본 연구에서 시뮬레이션을 위한 형상 모델링은 상용 그래픽 모델링 시스템인 3D Studio MAX를 사용하였다. 3D Studio MAX로 모델링 된 형상 모델은 OSG의 자체 기능을 이용하여 OSG의 자료 구조로 변환이 가능하다(Fig. 11①). OSG의 자료 구조가 생성된 이후에는 최상위에 'osg::PositionAttitudeTransform'을 삽입하여 ODE로부터 위치와 자세 정보를 전달

받을 수 있도록 한다(Fig. 11②). 이후에는 OSG의 자료 구조를 탐색하면서 DSGD를 생성해 나간다. 'dsgdRoot'는 항상 존재해야 하기 때문에 우선 생성을 한다(Fig. 11③). 그 후 최상위 'osg::Group'을 탐색하여 'dsgdObject'와 연결시키고(Fig. 11④), 'osg::Geode'를 탐색하여 'dsgdElement'와 각각 연결시킨다(Fig. 11⑤). 이렇게 DSGD가 완성되면, 이를 틀 이용하여 ODE의 자료 구조를 생성한다. 우선, 'dsgdObject'에 해당하는 'dBody'를 생성하여 연결시킨다(Fig. 11⑥). 그리고 'dsgdElement'를 통해 'osg::Geode'의 형상 정보를 읽어 들이고, 이 형상 정보를 가지고 'dGeom'을 각각 생성하여 'dsgdElement'에 연결시킨다(Fig. 11⑦). 이러한 과정을 거치면 3D Studio MAX로부터 ODE와 OSG 자료 구조를 자동 생성하고 이를 연결하여 DSGD를 자동 생성할 수 있다.

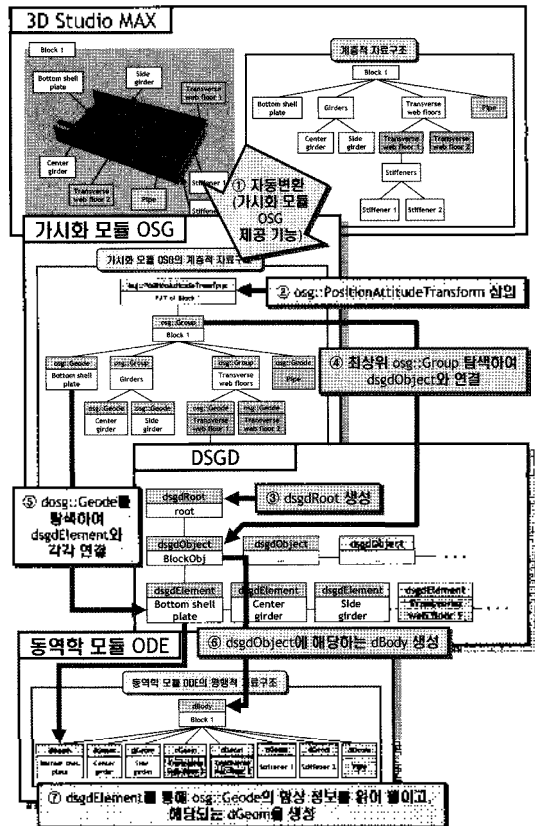


Fig. 11. 본 연구에서 제안한 DSGD 자동 생성 알고리즘.

4. 블록 탑재 시뮬레이션 예제

4.1 블록 탑재 시뮬레이션의 개요

선박을 건조할 때에는 선박을 여러 조각의 블록으

로분할하여 각각 제작하고 최종적으로 도크(dock)라고 하는 곳에서 조립하는 방식이 보편적이다. 도크에서 각 블록을 조립하는 과정을 탑재(erection) 과정이라고 하며, 블록을 들어서 도크에 내려 놓는 기구를 크레인(crane)이라고 한다.

본 연구에서 예제로 삼은 블록 탑재 시뮬레이션의 개요는 다음과 같다. 우선 도크 내에 'Block 1'이 탑재 되어 있다. 그리고 도크 외에 있는 선행 탑재장(pre-erection area)에 'Block 2'가 놓여 있을 때, 크레인이 이동하여 'Block 2'를 들어올려서(Fig. 12①) 도크로 이동하고(Fig. 12②), 'Block 2'를 'Block 1' 옆에 탑재시킨다(Fig. 12③). 이 때, 'Block 1'과 'Block 2'가 간섭을 일으키는 지 검사하고 간섭이 일어났을 경우 어떠한 부재 간에 간섭이 일어났는지 파악하는 것이 본 시뮬레이션의 목적이다¹⁾.

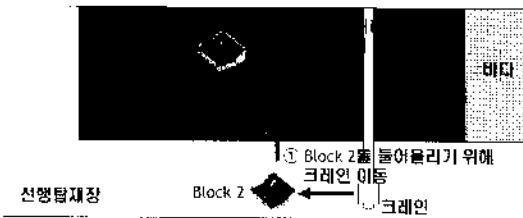


Fig. 12. 블록 탑재 시뮬레이션의 개요.

4.2 블록 탑재 시뮬레이션의 모델링

본 연구에서 블록 탑재 시뮬레이션은 사건(event)이 발생할 때마다 시간을 진행시키는 이산 사건 시뮬레이션(discrete event simulation)^{10,11)}과 매 단위 시간마다 시뮬레이션을 진행시키는 이산 시간 시뮬레이션(discrete time simulation)¹⁰⁾이 혼합된 시뮬레이션(combined discrete event and discrete time simulation)¹²⁾으로 구성하였다.

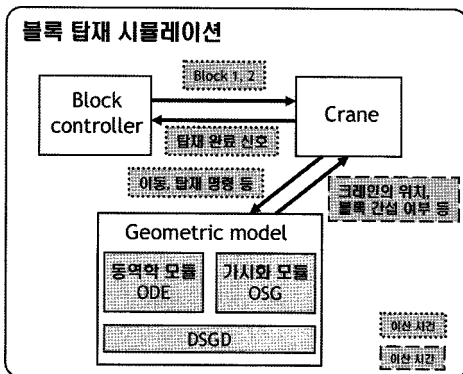


Fig. 13. 블록 탑재 시뮬레이션의 모델링.

Fig. 13에 나타나 있듯이, 블록 탑재 시뮬레이션에는 'Block controller', 'Crane', 'Geometric model'이라는 3개의 시뮬레이션 모델(simulation model)이 존재한다. 'Block controller'는 'Block 1', 'Block 2'를 'Crane' 모델로 전달해 주는 역할을 하고 'Crane'이 'Block 1', 'Block 2'의 탑재를 완료하면 이를 처리하는 역할을 한다. 'Crane'은 'Block controller'로부터 'Block 1', 'Block 2'를 전달 받아 'Block 1', 'Block 2'의 위치로 크레인을 이동시키고 'Block 1', 'Block 2'를 탑재하는 명령을 처리한다. 매 단위 시간마다 'Geometric model'로부터 크레인의 위치나 블록의 간섭 여부를 전달 받는다.

'Geometric model'은 크레인과 'Block 1', 'Block 2'의 형상 정보를 저장하고 있고, DSGD를 바탕으로 동역학 모듈 ODE, 가시화 모듈 OSG와 연결되어 있어 매 단위 시간마다 크레인의 위치, 블록 간섭 여부를 계산하고 이를 'Crane'에 전달하는 역할을 한다.

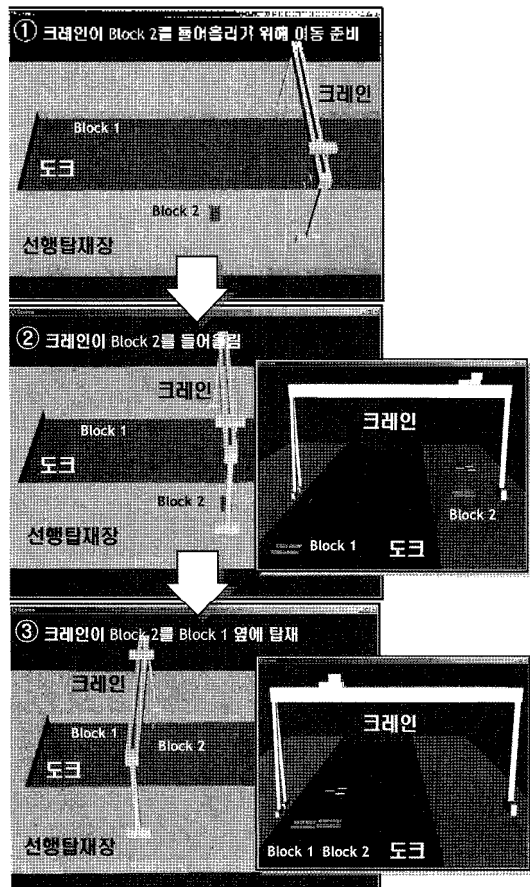


Fig. 14. 블록 탑재 시뮬레이션 결과.

4.3 블록 탑재 시뮬레이션 결과

Fig. 14는 앞서 서술한 블록 탑재 시뮬레이션의 개요에 따라 시뮬레이션을 수행한 결과 화면이다. 각 시뮬레이션 모델 간에 정보를 주고 받으면서 시뮬레이션을 개요 대로 수행하고 있는 것을 볼 수 있다.

Fig. 15는 'Block 2'를 'Block 1' 옆에 탑재시킬 때 간섭이 일어나는 것을 확대한 화면이다. 'Block 1'의 'Stiffener 1'과 'Block 2'의 'Pipe'간에 간섭이 발생한 것을 확인할 수 있으며, 간섭이 발생한 부재와 위치를 표시해준다. 이는 DSGD를 통해서 ODE와 OSG 간에 효과적으로 정보를 주고 받는 것을 나타낸다.

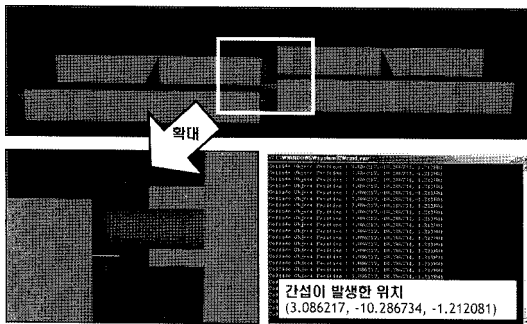


Fig. 15. 블록 탑재 시뮬레이션 결과로 'Block 1'과 'Block 2'가 간섭을 일으킨 경우.

5. 결론 및 향후 연구 계획

본 연구에서는 동역학 모듈 ODE와 가시화 모듈 OSG를 통합하여 연동할 수 있는 상위의 자료 구조인 DSGD를 제안하였고, 이를 블록 탑재 시뮬레이션에 적용하여 그 효용성을 검토하였다. DSGD의 핵심은 두 자료 구조가 평행적이건 계층적이건 개별적으로 통합될 수 있다는 것이고, 두 자료 구조 중 어떤 방향으로든지 정보를 주고 받을 수 있기 때문에 두 자료 구조의 장점을 모두 활용할 수 있다는 것이다. 그리고 매 시뮬레이션마다 최소한의 계산을 통해 두 자료 구조를 갱신(update)할 수 있도록 하였다. 또한 매다수 모듈의 자료 구조에서 제공하는 기능을 활용하였기 때문에 본 연구에서 대상으로 삼은 ODE나 OSG 의

의 다른 모듈에도 확장 적용이 가능하다.

향후에는 동역학 모듈과 가시화 모듈 외에 시뮬레이션에 필요한 각종 모듈과의 통합 방안에도 연구할 계획이다. 이러한 연구를 바탕으로 최종적으로는 완성된 시뮬레이션 프레임워크(simulation framework)를 제안할 수 있을 것이다.

감사의 글

본 연구는 a) 대우조선해양(주), b) 국방과학연구소, 수중운동체기술특화센터 SM-11과제 "수중 운동체의 체계/부체계 기능 및 성능 시뮬레이션을 위한 네트워크 기반의 가상(Virtual) 복합 시스템 모델 구조(Architecture) 연구" c) 서울대학교 해양시스템공학 연구소, (d) 울산대학교의 지원으로 이루어진 연구 결과의 일부임을 밝히며, 이에 감사 드립니다.

참고문헌

1. Open Dynamics Engine, <http://www.ode.org>
2. Wild Magic, <http://www.geometrictools.com>
3. DynaMechs, <http://dynamechs.sourceforge.net>
4. DsTool, <http://www.cam.cornell.edu/guckenheimer/dstool.html>
5. Open Scene Graph, <http://www.openscenegraph.org>
6. Visualization Library, <http://www.visualizationlibrary.com>
7. OGRE 3D, <http://www.ogre3d.org>
8. Russell Smith, *Open Dynamics Engine v0.5 User Guide*, 2006.
9. 방경운, "조선공정 계획용 이산 사건과 이산 시간 혼합형 시뮬레이션 프레임워크", 석사 학위 논문, 서울대학교, 2006.
10. Zeigler, B. P., Praehofer, H. and Kim, T. G., *Theory of Modeling and Simulation*, 2nd Edition, Academic Press, 2000.
11. Zeigler, B. P., *Object Oriented Simulation with Modular Hierarchical Models*, Academic Press, 1990.
12. Praehofer, H., Sametinger, J. and Stritzinger, A., "Concepts and Architecture of a Simulation Framework based on the JavaBeans Component Model", *Journal of Future Generation on Computer Systems*, Vol. 17, No. 5, pp. 539-559, 2001.



차 주 환

2002년 서울대학교 조선해양공학과 학사
 2004년 서울대학교 조선해양공학과 석사
 2008년 서울대학교 조선해양공학과 박사
 2008년~현재 서울대학교 공학연구소
 연구원

관심분야: 모델링 및 시뮬레이션, 다물
 체계 동역학, CAD, 음향수 모델링



노 명 일

1998년 서울대학교 조선해양공학과 학사
 2000년 서울대학교 조선해양공학과 석사
 2005년 서울대학교 조선해양공학과 박사
 2005년~2007년 서울대학교 공학연구소
 해양시스템공학연구소 선임연구원
 2007년~현재 울산대학교 조선해양공학
 부 전임강사

관심분야: 조선해양생산시스템, 좌석 설
 계, CAD, CAPP, CAGD



이 규 열

1971년 서울대학교 공과대학 조선공학
 과 학사

1975년 독일 하노버 공과대학 조선공
 학 석사(Dipl.-Ing.)

1982년 독일 하노버 공과대학 조선공
 학 박사(Dr.-Ing.)

1975년~1983년 독일 하노버 공과대학
 선박설계 및 이론연구소, 주정부
 연구원

1983년~1994년 한국기계연구원 선박
 해양공학연구센터, 선박설계, 생
 산자동화 연구사업(CSDP)단장

1994년~현재 서울대학교 공과대학 조
 선해양공학과 교수

관심분야: CAD, CAGD, 선박 설계, 시
 뮬레이션, CAIS