

工學碩士 學位論文

객체지향 정보모델에 기반한 선박조립
공정의 3차원 시뮬레이션에 관한 연구

Research on 3D Simulation for Assembly Process of Ship
Based on Object-Oriented Information Modeling

指導教授 朴 珠 用

2001年 2月

韓國海洋大學校 大學院

造船工學科

車 泰 仁

本 論文을 車泰仁의 工學碩士 學位論文으로 認准함.

主審：工學博士 朴 命 圭 (印)

副審：工學博士 朴 株 用 (印)

副審：工學博士 朴 錫 柱 (印)

2002年 2月

韓國海洋大學校 大學院

造船工學科 車 泰 仁

목 차

Abstract	ii
List of Figures	iii
제 1 장 서론	1
제 2 장 Simulation Based Manufacturing	3
2.1 Simulation Based Design	3
2.2 Simulation Based Manufacturing	4
2.3 시뮬레이션 기반 생산의 요소기술	5
2.4 SBD시스템의 프레임워크	7
2.5 Modeling & Simulation (M&S)	9
2.6 가상 프로토타이핑 기술	9
2.7 모델링 및 시뮬레이션 기술의 적용	10
제 3 장 객체 지향 모델링 기법	12
3.1 데이터베이스의 개요	12
3.2 객체와 클래스	14
3.3 클래스의 상관관계	16
3.4 객체 모델링 기법의 장점 및 단점	21
3.5 객체의 모델링	24
제 4 장 선박조립공정의 분석 및 모델링	28
4.1 개요	28
4.2 공정분석	29
4.3 시스템 설계	30
4.4 객체 모델링	31
제 5 장 선박조립공정의 3차원 시뮬레이션 구현	34
5.1 그래픽 사용자 인터페이스	34
5.2 선박조립공정의 구현 예	39
제 6 장 결론	46
참고문헌	47

Research on 3D Simulation for Assembly Process of Ship Based on Object-Oriented Information Modeling

Tae-In Cha

Dept. of naval Architecture, Graduate School,
Korea Maritime University

Abstract

Shipbuilding is a highly integrated technology and belongs to fabrication industry. A ship is a bulky and complex structure and consists of hundreds of thousands of members. It is built by a lot of material processing and fabricating and takes a long term to be completed. The verification of design, schedule plan and production is necessary before beginning of fabrication in order to confirm the optimal fabrication and high productivity. However, it is very difficult to apply this process in the shipbuilding because it is impossible to make a prototype of a ship.

The 3D simulation is a high technology that can realize the entire fabrication processes in the virtual world and verify the highest productivity and best quality of them. Object modeling technique is also an useful tool to implement the shipbuilding process into the computer world.

This paper deals with the 3D simulation of assembly process based on the object oriented information modeling technique. A lot of information of shipbuilding, especially assembly was analysed and modeled in the view point of object oriented modeling technique. This information model was constructed graphically using 3D CAD system, CATIA. On the basis of this graphic model, the assembly process was simulated using 3D simulation system, DELMIA. This 3D simulation recommended optimal assembly process and could show a possibility of the next generation technology of shipbuilding.

List of Figures

- Fig. 2.1 Virtual Prototyping Scenario
- Fig. 2.2 Dependencies of SBD Elements
- Fig. 2.3 Relationship of SBD/SBM Software Elements
- Fig. 2.4 Framework of Simulation Based Design
- Fig. 3.1 Stage of Object Oriented Technique
- Fig. 3.2 Three Model of Object Oriented Modeling
- Fig. 3.3 Example of Class and Objects
- Fig. 3.4 Example of Objects
- Fig. 3.5 Objects added 'Weight' as Attribute
- Fig. 3.6 Operation of Class
- Fig. 3.7 Objects, Class and Class Instance
- Fig. 3.8 Binary and Ternary Association
- Fig. 3.9 Association and Links
- Fig. 3.10 Role Name for Association
- Fig. 3.11 Link, Association and Aggregation
- Fig. 3.12 A Multi-level Inheritance Hierarchy with Instances
- Fig. 3.13 An Idea of Inheritance
- Fig. 4.1 System Configuration of Assembly Process of Ship
- Fig. 4.2 Relation between Objects
- Fig. 4.3 Flowchart of Process
- Fig. 4.4 Class Diagram of Subassembly
- Fig. 4.5 Definition for Kinematics of Robot
- Fig. 4.6 Welding Machine State Diagram
- Fig. 4.7 Gantry Crane State Diagram
- Fig. 4.8 I/O Programing
- Fig. 4.9 Event Tracing
- Fig. 5.1 Subassembly Process
- Fig. 5.2 Hierarchy of Menu
- Fig. 5.3 Joint Travel Limits

- Fig. 5.4 Volume Properties
- Fig. 5.5 Material Density
- Fig. 5.6 Max Speeds
- Fig. 5.7 Visualization of assembly
- Fig. 5.8 Product synthesis and analysis
- Fig. 5.9 Joint Setting
- Fig. 5.10 Set Kinematics
- Fig. 5.11 Programing
- Fig. 5.12 Device Information
- Fig. 5.13 Visualization of 3D Simulation
- Fig. 5.14 Procedure for Simulation
- Fig. 5.15 Collision Check
- Fig. 5.16 Simulation for Assembly Process of Ship

제 1 장 서론

오늘날의 조선 산업은 거대하고 복잡한 구조물로 이루어져 있고 인간, 기계, 컴퓨터 등의 많은 자원들로 인해 자원 및 정보의 효율적 관리에 어려움이 있다. 특히 세계를 단일시장으로 하고 있는 조선 산업에서 인터넷을 중심으로 한 IT 기술은 조선산업의 글로벌화, 조직의 유연화, 일의 분산화 등을 촉진시키고 있으며, 컴퓨터 및 관련기술을 바탕으로 선박의 설계, 건조 및 테스트에 걸친 전 과정을 혁신적으로 변화시키기 위해 제안하고 추진하고 있는 시뮬레이션기반 설계(Simulation Based Design, SBD) 및 시뮬레이션기반 생산(Simulation Based Manufacturing, SBM)의 디지털 선박 생산(Digital Shipbuilding)과 같은 새로운 개념의 엔지니어링 기술들이 연구 발표되고 있다. 선박과 같은 대형복합 구조물의 경우 개발기간이 2년 이상 소요되고, 이 기간동안 요구사항의 변경이 발생하며, 개념적인 요구사항이 제품정보로 구체화되고 제품화되면서 많은 문제점들이 노출될 수 있다. 그러나 시간이 흐를수록 새로운 요구사항을 수용하거나 문제점들을 수정하는 것은 어렵게 된다. 이러한 문제를 해결하기 위해 일반적으로 사용되는 방법이 시제품(Prototype)의 개발이다.[1][11]

그러나 선박의 경우 하나의 선박을 건조하기 위해서는 막대한 물량 및 시간, 비용이 소요되기 때문에 시제품이라는 것은 생각할 수 없다. 따라서 만들고자 하는 선박에 대한 설계, 생산 및 일정 계획 등에 대한 검증이 불가능하다고 할 수 있다. 그러므로 효율적인 생산을 위해서는 정밀한 설계 정보, 생산 정보의 추출과 추출된 정보들의 상호 교환 및 통합, 그리고 시뮬레이션에 기반한 생산이 필요하다고 할 수 있다. 모델링 및 시뮬레이션(Modeling & Simulation, M&S) 기술의 실현을 위해서는 설계 및 생산공정의 변화와 전산기술, 정보기술, 자동화기술, 시스템 통합기술 등의 새로운 기술들이 필요하다. 이러한 기술들을 조선 및 해양공학분야에 적용하기 위해서는 컴퓨터 상에서 가상의 프로토타이핑과 시뮬레이션 기술을 결합하여 선박의 설계, 건조, 유지 보수의 전 단계에 걸친 자원들을 통합하고 분석 및 평가를 실시간으로 처리하여, 설계 및 건조기간의 단축, 경비의 절감, 품질의 향상을 도모하는 것이 필요하다.[10]

시뮬레이션은 에뮬레이션에 정보가 부여되어 있는 것을 뜻하며, 디지털 선박 생산은 선박 생산과정을 구현하는 컴퓨터 모델을 만들고 이러한 모델을 이용하여 시뮬레이션 및 전체 생산과정을 하나의 통합데이터를 통해 구현하는 시스템의 개념이다. 이러한 시뮬레이션을 통해 새로운 선박의 설계 및 개발과정, 즉 개념적인 요구사항이 제품으로 구체화되는 과정에서 발생하는 기능적인 해석 및 평가, 시스템과 부품 상호간의 상호간섭, 최적운용 및 유지보수의 확보 등 제반 기술적인 문제들을 컴퓨터 기반의 가상세계(Virtual World)상에서 구현하고 검증할 수 있는 시뮬레이션 시스템을 개발할 수 있다.

본 연구에서는 SBD 와 Digital Shipbuilding에 대한 개념과 세부기술들을 정리하고 SBM 기술의 핵심기술인 M&S의 실현을 위한 요소 기술의 분석을 위해 분산 가상환경기술, 공통 시뮬레이션 구조, 가상 프로토타이핑 대한 내용을 정리하였다. 또한, 선박의 조립공정을 시뮬레이트 하였으며, 그 과정에서 리소스 및 아키텍처를 포함한 생산 시스템의 공정(자원과 구성)을 정의하고 최적화하며 예상되는 성과를 예측하였다. 이러한 3차원 시뮬레이션은 공정과정의 평가와 최적화를 위한 가장 유용한 틀이며, 객체지향 모델링 기법은 조립공정을 모델링하고 분석하기 위해 도입되었다. 3차원 시뮬레이션은 시뮬레이션 툴로써 카티아와 델미아를 사용하여 객체지향 모델을 기반으로 기존의 분산되어 있던 시스템 개념 기술 및 정보들을 통합함으로써 실제화하였다. 두가지의 소규모 블록의 조립공정이 시뮬레이트되고 비교되었으며, 시뮬레이션 결과는 조립공정의 최적화와 진보의 가능성을 보여준다.

2. 시뮬레이션 기반 생산(Simulation Based Manufacturing)

2.1 시뮬레이션 기반 설계(Simulation Based Design)

SBD란 미국방성의 ARPA(Advanced Research Projects Agency)에서 선박의 설계, 건조 및 테스트에 걸친 전과정을 혁신적으로 변화시켜 설계 및 건조기간의 단축, 경비의 절감, 품질의 향상을 도모하기 위해 제안하고 추진하고 있는 새로운 개념이자 기술이다. 또한, 제품모델을 토대로 가상 프로토타이핑과 시뮬레이션 기술을 결합하여 선박의 설계, 건조 유지보수의 전 단계에 걸친 제반자원들을 통합하고 실시간 지원하기 위한 차세대 시스템 기술로 정의되고 있다. 컴퓨터 기술을 이용한 가상세계는 일반적인 프로토타이핑 과정에서 발생하는 하드웨어적(Hardware)인 제약 뿐만 아니라 공간적·시간적 제약이 없으며 설계 환경, 제조환경 및 운용환경 사이를 자연스럽게 연결할 수 있는 가상 프로토타이핑의 환경을 제공한다.[5][6][9][11]

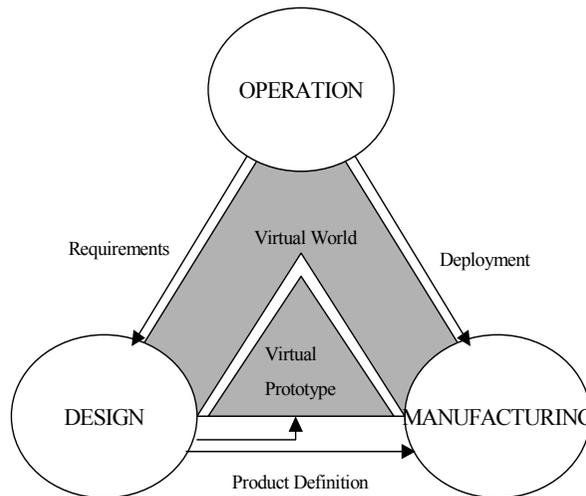


Fig. 2.1 Virtual Prototyping Scenario

공학적인 측면에서 Virtual World는 정확한 모델링(Geometry), 물리법칙의 표현(Behavior, Physics Based Analysis), 현실감(Appearance), 실시간 반응(Real-time Response)을 만족해야 한다. 게임프로그램의 경우 Geometry나 Appearance, 그리고 Real-time Response 측면에서 전자의 조건을 만족하지만 물리법칙과는 무관하며, 애니메이션의 경우 상황표현에 있어서는 현실감이 있지만 실시간 반응의 측면에서 문제가 있으므로 Virtual World와는 거리가 있다.

SBD는 기본적으로 Product and Process Model (PPM), Infrastructure, 그리고 VE (Virtual Environment) 및 VP (Virtual Prototyping)로 구성된다. VE는 설계/개발자들이 제반설계 Resource들에 실시간으로 접근하고 협력할 수 있는 환경(Medium)을 제공한다. VP는 VE내에 생성되는 PPM의 Subset이며 실제품의 형상과 운용환경에서의 거동뿐만 아니라 그 제품의 제조 및 조립과정까지 시뮬레이션 할 수 있다. 따라서 설계/개발자로 하여금 물리적인 시제품이 없더라도 개발하고자 하는 제품을 이해할 수 있는 수단이 된다. VP를 통하여 그 제품을 정확히 표현(Representation)하기 위해서는 잘 정의된 PPM의 사용을 통해서만 가능하다. PPM이란 SBD를 통해 설계/개발자들이 구체화해가는 제품 및 공정정보의 집합체이다. PPM에는 시뮬레이션 및 거동특성, 제품의 정의에 필요한 데이터와 이들을 처리하기 위한 단위 프로그램들이 내재되어 있다. VE와 PPM은 Infrastructure를 통해 통합될 때만 충분히 유용하다. Infrastructure란 설계/개발에 참여하는 구성원들이 PPM에 실시간으로 접근하면서 관련 자원들을 공유/협력할 수 있도록 하는 하드웨어 및 소프트웨어의 구조(Architecture), 설계지원 응용시스템, VE, 고성능/고단위 통합 네트워크이 포함된다.[8][9][10]

2.2 시뮬레이션 기반 생산(Simulation Based Manufacturing)

선박 생산에는 시제품(Prototype)의 제작이 현실적으로 불가능하다. 또한 다 품종 소량생산이기 때문에 시제품의 제작이라는 것이 큰 의미를 가지지 못한다. 선박 생산과 같은 산업에서는 실시간적으로 작업 과정의 가상 구현(Real-time Process Simulation)이 필요하다. 이러한 검증 작업 및 작업과정 최적화를 가능하게 하는 것이 시뮬레이션을 기반으로 한 생산 방식이다. 이러한 시뮬레이션

기반의 생산은 디지털 선박 생산을 위한 빠질 수 없는 기술이다. 과학적인 선박 생산, 즉 연속적이고 일관된 선박 생산을 위한 디지털기반의 생산 통합 시스템의 개념이 바로 디지털 선박 생산이다. 따라서 디지털 선박 생산에는 설계, 생산, 공학에 관련된 요소 기술들과 더불어 각 분야들의 통합/관리 등이 포함된다. 또한 여기에는 기존 작업자의 개인적 기술과 경험에 의존하던 작업들에 대해 컴퓨터를 이용한 새로운 작업 대안(Substitution)들도 포함되게 된다.[2][3]

2.3 시뮬레이션 기반 생산의 요소기술

SBM은 설계 및 건조과정의 모든 자원(Resource)들을 실시간에 완벽하게 통합하기 위한 컴퓨터 기술의 집합으로 정의할 수 있다.

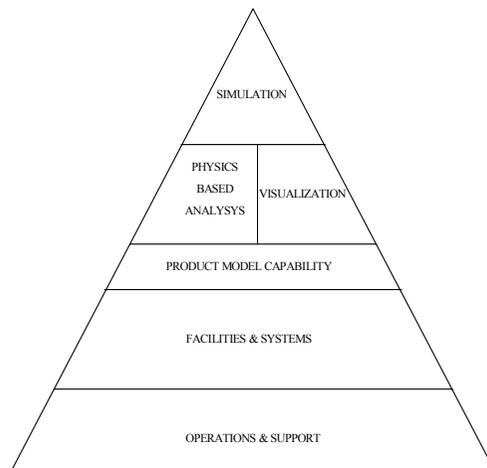


Fig. 2.2 Dependencies of SBD Elements

Fig. 2.2 는 SBD 및 SBM을 구현하기 위한 요소기술과 각 기술의 계층구조를 나타내고 있다. 그림에서 보는 바와 같이 시뮬레이션은 기존의 하드웨어 및 소프트웨어 시스템들의 효율적인 운영(Operation)과 지원(Support)을 전제로 하고 있다. 그 위에 설계 및 엔지니어링 소프트웨어의 기반이 되는 장비와 시스템, 즉 워크스테이션, 네트워크, 서버 및 주변장치들이 필요하다. 이들 장비와 시스템들

은 급격히 증대되는 모델크기, 소프트웨어 개발 및 사용자의 요구에 따라 지속적으로 보완되고 교체되게 된다. 성능해석 및 평가를 위한 각종 CAE 프로그램들도 SBD/SBM의 중요한 요소이다. 기존의 대부분의 CAE 프로그램들은 각각 독립적인 환경에서 개발, 사용되고 있으나 SBD/SBM에서는 제품모델의 형상정보 및 속성정보와 밀접하게 연계된 Physics Based Analysis가 필요하다. 3-D CAD 모델링 시스템들은 기본적으로 가시화(Visualization)기능을 제공하고 있으며 설계/개발자에게는 충분히 유용하다. 그러나 관리자(Manager), 생산관계자(Production Staff), 운용자(Operator)들이 설계의 내용을 검토하고 평가하기 위해서는 보다 현실적인 가시화기능, 즉 Virtual Mockup이 요구된다.

마지막으로 설계 초기 단계에서 제작/건조 및 운용단계의 제반조건을 평가하기 위한 시뮬레이션이 필요하다. 시뮬레이션 기술은 제품모델을 토대로 가시화 기술과 물리적 거동 모델(Behavior Model)을 결합한 Virtual Prototype을 통하여 구현된다.

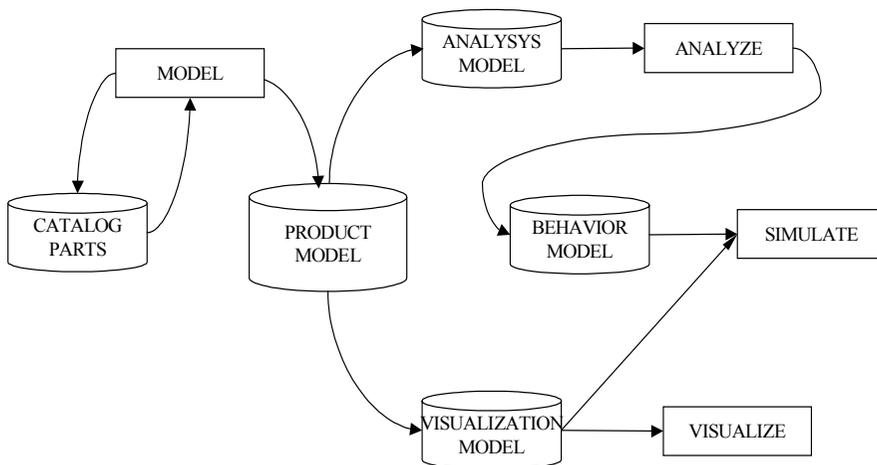


Fig. 2.3 Relationship of SBD/SBM Software Elements

Fig. 2.3은 SBD/SBM의 구성요소인 모델링, 해석, 가시화, 시뮬레이션 및 관련 데이터베이스와의 관계를 그림으로 표현한 것이다.[5][6][8][9][11]

2.4 SBD/SBM 시스템의 프레임워크

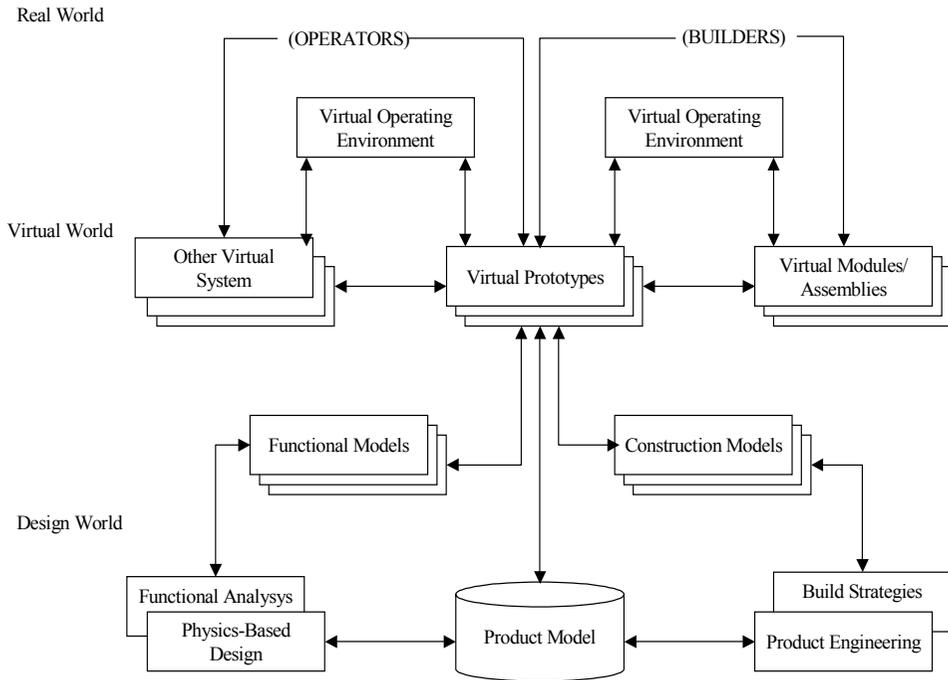


Fig. 2.4 Framework of Simulation Based Design

Fig. 2.4는 이러한 SBD시스템을 구현하기 위하여 ARPA에서 정의한 SBD시스템 프레임워크이다. 그림에서 보는 바와 같이 시스템은 제품모델(Product Model), 기능모델(Function Model), 건조모델(Construction Model)을 토대로 이들을 결합한 Virtual Prototype들과 관련 Virtual System들, 그리고 이들이 구현되는 가상환경(Virtual Environment)으로 구성된다.[9][10][11]

(1) 제품모델(Product Model)

제품모델이란 개발하고자 하는 제품(선박)에 대한 완전한 정의를 포함하는 데이터베이스이다. 이 모델은 형상의 기하학적 표현에 국한된 것이 아니고 중량,

전기 부하, 중량중심 및 거의 모든 성능 파라메타들과 같은 비기하학적 속성들도 포함한다. ARPA에서는 선박의 설계/개발을 위한 제품 모델링 기능을 상용 CAD 시스템에 의존하고 있다.[6][11]

(2)기능모델(Function Model)

공학적으로 의미 있는 Virtual Prototype이 되기 위해서는 거동이 정확히 표현되어야 한다. 즉 물리법칙을 기초로 한 거동이 되어야 한다. Virtual World에서 주어진 기능을 시뮬레이션 할 때 해당 시스템의 거동(Behavior)을 표현하기 위한 모델로 Virtual Prototype의 두뇌에 해당하는 것이 기능모델이다. 즉, 시스템 거동의 양상들을 전달해주며, 제어 입력에 대한 반응들을 출력한다. 대화식 환경의 효율성을 생각하면, 기능 모델은 다른 시스템 모델로부터 입력을 받을 수 있어야 하며, 다른 시스템의 시뮬레이션에서 사용되기 위해 시간 영역대의 출력을 제공해야 한다. 그리고 제품 모델의 기하 및 특성 데이터베이스와 연계될 수 있어야 한다.[6][11]

(3)건조모델(Construction Model)

생산의 측면에서 Virtual Prototype에서의 기능 모델에 대응되는 것이 건조모델이다. 제작을 위한 모델링은 제작물 및 공정(Process)에 대한 표현이 요구되며, 조립순서, 건조기술 혹은 제작계획에 의한 조립과정을 나타낸다. 또한, 건조모델은 용접 뒤틀림과 같이 건조시에 일어날 수 있는 현상을 정량화시키는 어떤 거동 모델도 포함한다. 선박의 여러 가지 제작 양상을 표현하기 위해서는 다수의 모델이 필요하다. 건조모델의 생성은 제작을 위해 아주 적합한 가상의 Prototype을 생성하거나 어떤 Prototype의 제작을 위해 아주 적합한 공정을 생성하는데 있어서 중요한 역할을 한다.[6][11]

(4)가상프로토타입(Virtual prototype)

Virtual Prototype은 제품모델, 그리고 기능모델에서 표현된 다른 거동 및 물리적 속성들, 그리고 공정에 근거를 두고 있는 건조모델을 결합하여 Virtual Environment에서 생성된다. Virtual Prototype이 설계/개발환경에서 요구되는

실시간 반응을 만족하기 위해, 컴퓨터는 제품 모델의 한 부분을 구성하는 형상을 표현하는 거대한 기하 데이터베이스를 재계산할 수 있는 능력이 있어야 한다. 아울러 제어 입력과 prototype, 시스템 및 환경들 간의 상호작용에 대한 실시간 반응을 보장하기 위해 기능모델과 관련된 수많은 계산을 수행할 필요가 있다. 현재로는 설계환경에서 그와 같은 거대한 양의 데이터를 취급할 수 있는 컴퓨터 시스템은 실용화 되어 있지 않다. 그러나 설계/개발자, 운전자 및 제작자들은 관심이 있는 특정기능을 위해 제품모델 데이터베이스의 일부만을 사용하게 되며, 따라서 모든 기능을 갖춘 하나의 Prototype 보다는 여러 가지 기능에 적합한 작은 단위의 Prototype 군이 현실적이다.[6][10][11]

(5)가상환경(Virtual System/Virtual Environment)

Virtual Prototype들은 그들이 운용될 적절한 영역, 즉 Virtual Environment를 필요로 한다. Virtual Environment 내에서 Virtual prototype이 다른 기존의 운용가능한 시스템들과의 통합을 시뮬레이션하기 위해 별도의 Virtual System들을 필요로 할 것이다. 이들 시스템들은 요구되는 VE체험의 전 영역을 만족시키기 위해 시뮬레이션된 환경 내에서 실질적인 물체와 가상 물체의 결합체를 조화시킬 수 있어야 한다. 그리고 기존의 시스템을 Virtual Environment 내에 삽입하기 위해, 우선 그 시스템들은 Virtual Counterpart로 변형되어야 한다. 부득이 하지만, 널리 분산되어 있는 시스템과 시스템 개념(Representation)들을 함께 모으는 것은 분산 대화식 시뮬레이션(Distributed Interactive Simulation)이라고 하는 부수적인 기술의 이용이 필요할 것이다.[6][11][14]

2.5 모델링 & 시뮬레이션(Modeling & Simulation, M&S)

M&S 기술은 기존의 제품 모델링 기술, 엔지니어링 데이터베이스 기술, 가상 현실(Virtual Reality)기술, 컴퓨터 그래픽스 기술, 네트워크 기술, 동시공학(Concurrent Engineering)기술 등이 하나의 통합된 형태로 구성되어 있는 기술이라고 할 수 있다. 이와 같은 M&S 기술은 최근에 관심을 모으고 있는 가상 생산, 가상 조선소, 조종운동 시뮬레이터, 가상 항만시스템 등을 구축하기 위한

기반 기술로서 활용될 수 있다.[10]

2.6 가상 프로토타이핑 기술

가상 프로토타이핑(Virtual Prototyping) 기술은 3차원 CAD를 이용하여 작성된 모델을 실제의 조립과정 없이 컴퓨터 상에서 제품을 조립하거나 작동시키는 등의 작업을 설계단계에서 수행하며 이를 통해 설계의 품질을 검증하여 제품 개발기간 단축, 원가절감 및 품질향상을 통해 전반적인 생산성 향상을 가능하게 하는 기술이다. 즉, 물리적 프로토타입(Physical Prototype)에 비교할 만한 이미지와 기능상의 리얼리즘을 갖는 가상의 프로토타입을 이용하여, 컴퓨터 시뮬레이션을 통해 제품의 설계, 시험, 평가 등의 단계의 결과를 시각적인 방법으로 관찰, 해석, 처리하는 것이다. 이러한 가상 프로토타이핑 기술을 분산된 네트워크 환경에서 구현함으로써, 지리적으로 분산되어 있는 관련 당사자들이 가상의 공간에서 협동 작업을 할 수 있는 환경을 제공하는 것이 협동형 가상 프로토타이핑 기술이다.

가상 프로토타이핑 기술은 M&S 기술에 추가하여 Electronic Collaboration 기능을 가지는데, 이는 CAD, M&S, 가상화, 제품분석, 가상현실 등을 지원하는 도구들의 통합뿐만 아니라, 멀티 미디어 응용, 비디오 화상 회의(Video Conferencing), E-mail 등의 기술을 통한 사람들 사이의 통합을 의미한다. 이러한 협동형 가상 프로토타이핑 기술은 가상환경에서의 설계검증, 가상 조선소와 같은 새로운 개념의 기술을 실현하는 기반이 된다. 가상환경에서의 설계검증 기술은 실시간 3D Ship Model의 Walk-through, Interference Checking, 설계대안 평가, 선박 사고분석 등에 적용되며, 가상 조선소 환경을 통해 Multi-yard Load Tracking, Yard 설계, Finite Capacity Scheduling, 제품데이터와 생산데이터의 링크 등이 가능하다.[11]

2.7 모델링 및 시뮬레이션 기술의 적용

조선 및 해양공학 분야에서는 M&S에 대한 연구 및 적용이 일부 진행되고

있으나, 전체적인 관점에서 가상 조선포를 지향하는 통합적 접근의 시도는 최근에 다시 시도되고 있다. 이렇게 늦어지는 이유는 새로운 개념의 선박이나 대규모 해양 구조물이 소규모 기계 조립품에 비해 수만 종의 부품을 포함하고 있으므로, 이러한 대량의 정보를 취급할 수 있는 하드웨어 및 소프트웨어로 변하는 작업환경을 반영하기가 쉽지 않기 때문이다. 다시말해, M&S 기술을 조선 및 해양공학 분야에 적용하기 위해서는 고성능 H/W, 고해상도 지원 기능, 프라덕트 모델과의 직접 인터페이스 기능, 3D Geometry & Visualization 기능, 재질에 관련된 기술적 특성, 정보계층에 관련된 기능이 요구된다.

선박 및 선박조립공정 시뮬레이션 시스템에 사용되어지는 계산 CAE 프로그램들은 원가계산, 생산공정 및 스케줄 등을 통합화 한 데이터베이스인 PDM(Product Data Management)과의 연동으로 설계정의, 상태 및 의도들을 가시화하고 관련자들이 동시에 커뮤니케이션을 함으로써 동시공학을 추구할 수 있으며, 선박 및 관련 생산품의 안정성 및 최적화를 평가하는 기준이 되어진다. 선박 및 선박조립공정의 안정성 및 최적화의 평가는 관련 생산품 설계시 이용되어지는 기본계산 프로그램을 응용할 수 있으며, 동적상태 평가는 Visual C/C++ 언어와 ENVISION or IGRIP 시스템에서 제공하는 GSL(Graphic Simulation Language)와 CLI(Command Line Interpreter)를 이용하여 구현 할 수 있다.

제 3 장 객체 지향 모델링 기법

3.1 객체 지향 모델링 기법의 개요

객체 지향 모델링 기법(Object Oriented Modeling Technique)은 실세계에서 일어나는 여러 문제들을 새로운 개념의 모델링과 디자인을 이용한 접근법으로 해결하는 기법으로, 실세계에 존재하는 여러 가지 현상이나 사물에 대하여 객체(Object)라는 개념을 도입하여 모델화하고 공통적인 속성(Attribute)을 가진 각각의 객체들에 대하여 하나의 클래스(Class)로 정의하여, 링크(Link), 연관화(Association), 일반화(Generalization), 상속(Inheritance), 집단화(Aggregation) 등을 이용하여 클래스와 클래스 혹은 객체와 객체간의 관계를 정의하는 것으로 요약할 수 있다. 이러한 객체 지향 기법은 문제를 이해하는데 유용할 뿐만 아니라, 문제를 해결해 가는 과정 중 전문가들간의 정보 교환, 문제 해결을 위한 프로그램의 디자인 그리고 데이터베이스의 설계에도 기존의 모델링 기법에 비해 더욱 유용하게 사용된다. 객체 모델링 기법을 사용하여 실세계의 문제를 해결하는 과정은 Fig. 3.1에 나타낸 것처럼 4가지의 단계로 이루어 진다. 이에 대해서 자세히 살펴보면 다음과 같다.[4][16][17][18]

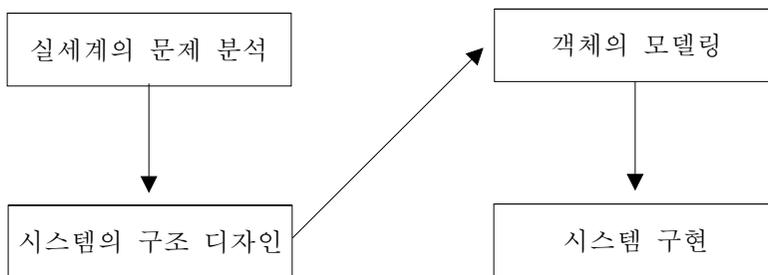


Fig. 3.1 Stage of Object Oriented Technique

첫 번째, 실세계의 문제를 분석하여 이해하고 실제 문제를 해결하는 시스템 구현의 가능성 및 구현하는 방법 등에 대한 판단을 하는 단계로서, 예를 들어 프로그램으로 시스템을 구현할 것인지, 아니면 하드웨어로서 구현할 것인지에

관하여 결정하는 단계이다.

두 번째, 시스템의 전반적인 구조에 대하여 결정하는 단계로서 각 서브 시스템들의 구성과 그 기능에 대하여 결정 등이 이루어진다.

세 번째, 두 번째 단계까지 구성된 모델에 대하여 상세한 내용을 모델링하는 단계이며, 시스템에서 필요한 객체 및 클래스, 이들 객체들의 서로 구분될 수 있는 구체적인 값인 속성, 객체가 발생시킬 수 있는 사건(Event)이나 구체적인 동작인 기능(Operation) 및 객체들의 상관관계에 대해서 모델링한다.

마지막 단계에서는 세 번째 단계까지 정의 및 모델링된 객체와 클래스의 속성, 기능, 상관관계를 이용하여 실세계의 문제를 해결할 수 있는 프로그램 또는 하드웨어를 구현하게 된다.[4][18]

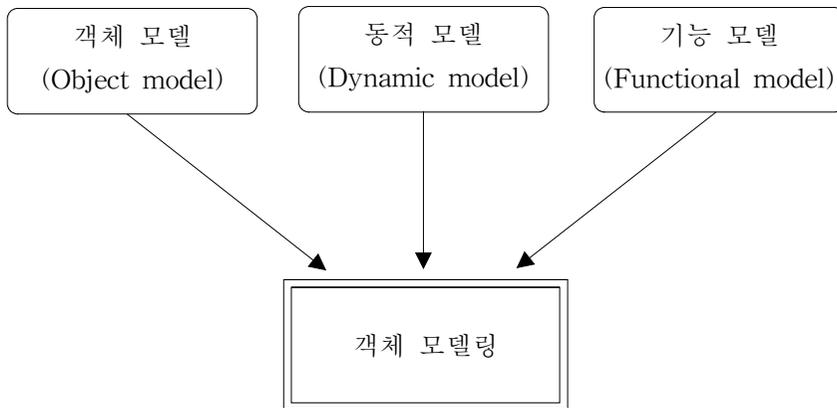


Fig. 3.2 Three Model of Object Oriented Modeling

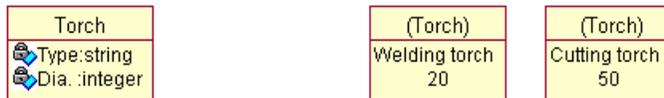
모델링 단계에서 이용되는 모델은 Fig. 3.2에 나타낸 것과 같이 객체 모델 (Object Model), 동적 모델(Dynamic Model), 기능 모델(Functional Model)이 사용된다. 먼저 객체 모델에 대해서 살펴보면 시스템에 있어서의 객체의 구조 즉, 객체의 정의, 정의된 객체 사이의 관계, 속성 및 기능을 기술하는데 사용되는 모델이다. 이러한 객체 모델은 객체 혹은 클래스들을 포함하는 그래픽 다이어그램으로 표기된다. 동적 모델은 시간의 흐름을 고려한 시스템의 흐름과 기능들의 동작 순서를 기술하기 위한 모델이며, 기능 모델은 시스템의 값의 전송이

나, 함수, 매핑 등을 기술하기 위한 모델이다. 이러한 모델들을 기반으로 하여 실제계의 문제를 위한 시스템을 모델링하게 된다.[4][16][17][18]

3.2 객체와 클래스

(1) 객체

객체는 주어진 문제에 대해 뚜렷하게 구분되고 명확한 의미를 갖는 구체적인 개념을 말한다. 또한 객체는 다른 객체에 대하여 독립적이며 유일한 개념을 갖는다. 즉 다른 객체와 엄밀하게 구분이 되는데, 객체는 객체를 구분하기 위한 하나의 객체명과 하나 이상의 속성(Attribute)으로 구성되어 있다. 이러한 객체의 개념은 다음의 Fig. 3.3과 같이 다이어그램으로 나타낼 수 있다.



(a) Class with attributes (b) Objects with values

Fig. 3.3 Example of Class and Objects

객체명은 Torch이며 각 Torch 객체들은 'Type'과 'Dia'라는 2개의 속성을 가지고 있으며 각각 다른 값들을 가지고 있다. 이와 같이 객체는 객체간에 서로 구분될 수 있는 특징을 가지고 있다. 그러나 객체를 모델링하는 과정에서 객체들의 개별성을 확보할 수 없는 경우가 발생한다면 올바르게 객체 모델링이 되었다고는 볼 수 없다.

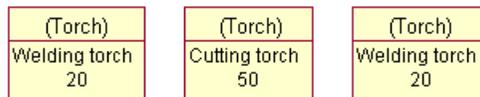


Fig. 3.4 Example of Objects

이러한 예로서 Fig. 3.4와 같이 Torch의 객체 중에 Type과 Dia가 'Welding

torch', '20'인 객체 2개가 중복되어 존재하는 경우가 있을 수 있다. 이러한 경우에는 객체를 표현하는 속성의 범위를 확대하여 객체의 개별성을 확보할 수 있도록 해야 한다. Fig. 3.5는 Fig. 3.4에서 중복된 객체의 오류를 수정하기 위하여 'Weight'라는 속성을 추가하여 모델링한 것이다.

(Torch)	(Torch)	(Torch)
Welding torch	Cutting torch	Welding torch
20	50	20
10	15	5

Fig. 3.5 Objects added 'Weight' as Attribute

(2)클래스(Class)

클래스란 동일한 속성(Attributes)과 기능(Operations)을 갖는 객체들의 집합을 의미한다. 객체와 클래스의 차이점은 클래스에는 기능(Operations)이라는 요소가 추가되는 것이다. 기능은 클래스로부터 생성될 객체의 구체적인 거동을 나타내는 역할을 한다. 기능이 포함된 클래스의 모델은 Fig. 3.6과 같다. Fig. 3.6에서는 File이라는 이름을 가진 클래스의 속성이 'Name', 'Size', 'Date'이며 기능이 Copy라는 것을 나타내고 있다.[18]

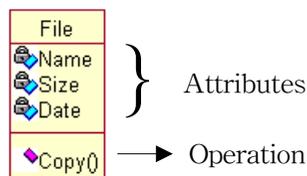


Fig. 3.6 Operation of Class

이러한 클래스가 객체로부터 생성되는 것을 Fig. 3.7에 나타내었다. 'Torch'라는 여러 객체로부터 공통적인 속성들을 이용하여 Torch 클래스를 정의하였으며, 새로운 Torch 객체의 생성이 필요하다면 Torch 클래스로부터 사례(Instance)를 생성하게 된다. 따라서 이러한 그룹화를 통해 하나의 클래스로 만들어진 각 객체들은 공통적인 속성과 거동을 갖는 추상적 개념으로 일반화되며 문제의

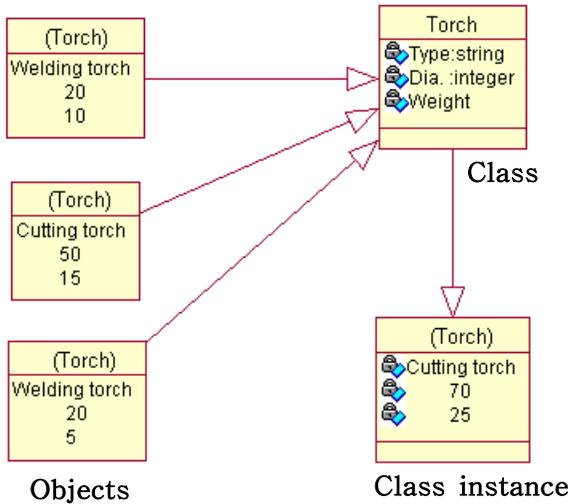


Fig. 3.7 Objects, Class and Class Instance

대상인 객체의 모델링이 용이하게 되고 동일한 속성을 지닌 객체가 사용될 때마다 새로운 코드를 기입할 필요가 없고, 클래스로부터 새로운 사례(Instance)를 생성함으로써 프로그램 코드의 재사용성을 높일 수 있다. [12]

3.3 클래스의 상관관계

클래스와 클래스의 상관관계는 링크(Link)를 비롯하여 연관화(Association), 일반화(Generalization), 상속(Inheritance), 집단화(Aggregation) 등의 개념을 도입하여 연관 지을 수 있다. 각각의 의미와 이후 객체 모델링을 위한 다이어그램은 다음과 같다.

(1) 링크와 연관화

링크는 객체들간의 물리적인 또는 개념적인 연결을 의미한다. 이러한 링크들이 동일한 구조와 의미를 가진 집합의 형태를 취하고 있을 경우 이를 연관화(Association)라고 하며 연관화의 형태에 따라서 두 객체가 연관화를 이루는 것과 셋 이상의 객체가 연관화를 이루는 것으로 나눌 수 있으며, 대응관계에 따라

서는 일대일(1:1)과 일대다(1:n), 다대다(n:n) 연관으로 나눌 수 있다.[4]

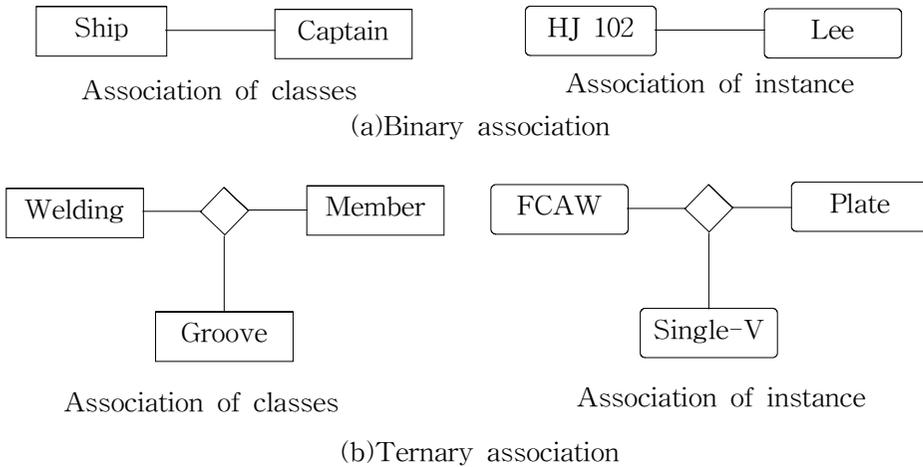


Fig. 3.8 Binary and Ternary Association

이 두 가지 개념에 대해서는 Fig. 3.8 및 Fig. 3.9에서 설명하고 있다. Fig. 3.8(a)에서는 Ship, Captain 클래스 사이의 연관과 그 인스턴스인 HJ 102 와 Lee 객체의 연관을 나타내고 있으며, Fig. 3.8(b)에서는 Welding, Member, Groove클래스 사이의 연관 및 그의 인스턴스에 대한 연관을 나타내고 있다.

Fig. 3.9(a)에서는 두 클래스 사이의 1:1 링크에 의한 연관화를, Fig. 3.9(b)에서는 두 클래스 사이의 n:n 링크에 의한 연관화를 보여주고 있다. 즉, Fig. 3.9(a)는 하나의 Ship에 하나의 Captain이 1:1로 연관이 되고 Fig. 3.9(b)에서는 하나의 'Joint'에 여러 개의 'Groove'를 적용할 수 있고, 하나의 Groove가 여러 개의 Joint에 사용될 수 있음을 의미하여 다대다의 연관을 나타내고 있다.

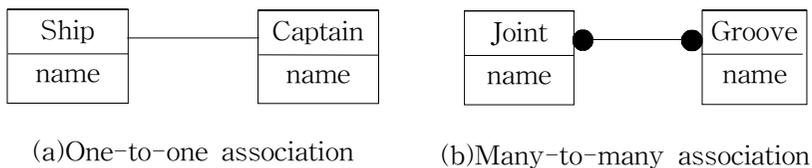


Fig. 3.9 Association and Links

연관에 대한 구체적인 설명 또한 다이어그램에 추가해서 나타낼 수가 있는데,

이는 Fig. 3.10과 같다. Person클래스와 Company클래스의 연관 관계가 Works-for라는 것을 구체적으로 나타내고 있으며 좌측과 우측에 각각 나타낸 employee와 employer는 각 클래스가 연관에서 역할을 하고 있는 지를 나타내고 있다. 이것을 Role name이라고 한다.[17][18]

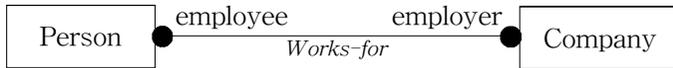


Fig. 3.10 Role Name for Association

(2) 집단화(Aggregation)

Fig. 3.11과 같이 클래스들이 모여서 집단의 성질을 가지는 경우를 집단화(Aggregation)라 한다. 즉 Welding Equipment는 welding robot과 gantry, welding machine, controller로 구성이 되고, welding machine은 welding torch, wire feeder, power/gas supply로 구성되는 것을 나타내며 “◇”으로 나타낸다.

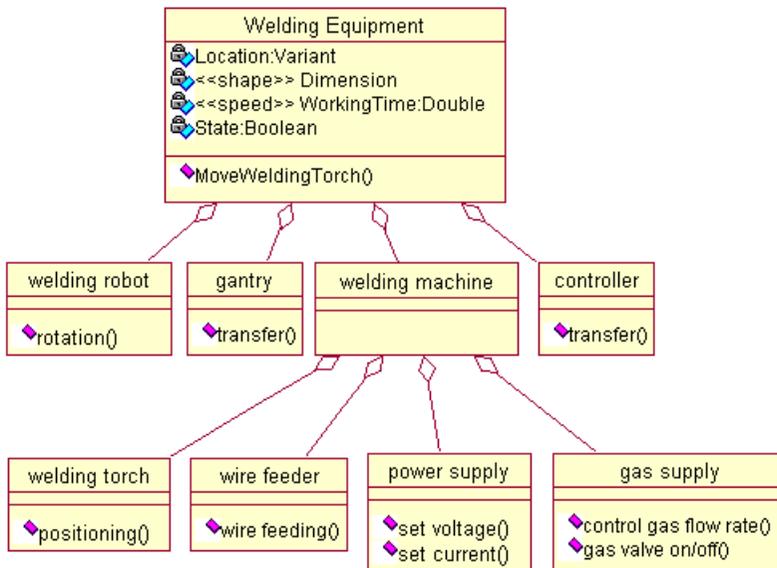


Fig. 3.11 Link, Association and Aggregation

(3) 일반화와 상속

일반화(Generalization)와 상속(Inheritance)은 서로 차별성을 지닌 클래스들 사이의 유사성을 추출하여 공유하는 수단이다. 실세계의 문제 대상에 대해서 유사성과 차별성을 기반으로 유사한 속성들을 포함하고 있는 상위클래스(Superclass)와 독특하게 구별되는 속성을 포함한 하위클래스(Subclass)로 구분하는 것이 일반화이며 상속(Inheritance)은 하위클래스가 상위클래스로부터 추출된 공통적인 속성과 기능을 상속받는 것을 의미한다. 이러한 상속의 개념을 이용하여 상위클래스와 하위클래스는 속성과 기능을 공유할 수 있다. Fig. 3.12는 상속에 대한 예를 보여주고 있는데 “△”으로 표시된다. Fig. 3.12의 가장 상위클래스는 ‘Equipment’이며 하위클래스로는 ‘Welding robot’, ‘Gantry’, ‘Torch’ 등이 있다. 이들 중에서 Equipment 클래스와 Torch 클래스를 예로 보면, Torch 클래스는 ‘diameter’, ‘height’ 와 같은 고유 속성을 가지면서 상위클래스인 ‘Equipment’로부터 공통 속성인 ‘name’, ‘manufacturer’, ‘weight’ 를 상속받게 된다. 또한 Torch 클래스의 하위클래스인 Welding torch, Cutting torch 클래스들은 Equipment 클래스와 Torch 클래스의 모든 속성들을 상속받을 수 있다. 이는 Fig. 3.13과 같다. 이로서 Cutting torch의 클래스로부터 새롭게 생성된 객체 즉, 새로운 인스턴스(Instance)는 Fig. 3.12의 하단에 나타낸 것처럼 Cutting torch 클래스의 속성인 nozzle, cutting positioning, marking positioning을 비롯하여 상위클래스로부터 상속받은 속성인 name, weight, manufacturer, diameter, height 모두를 상속받게 되며 Cutting torch 클래스에서 생성된 객체는 Fig. 3.13에서 보는 것과 같이 Cutting torch 자신의 속성 이외에 상위클래스로부터의 속성을 모두 상속받은 것을 알 수 있다.[4][18]

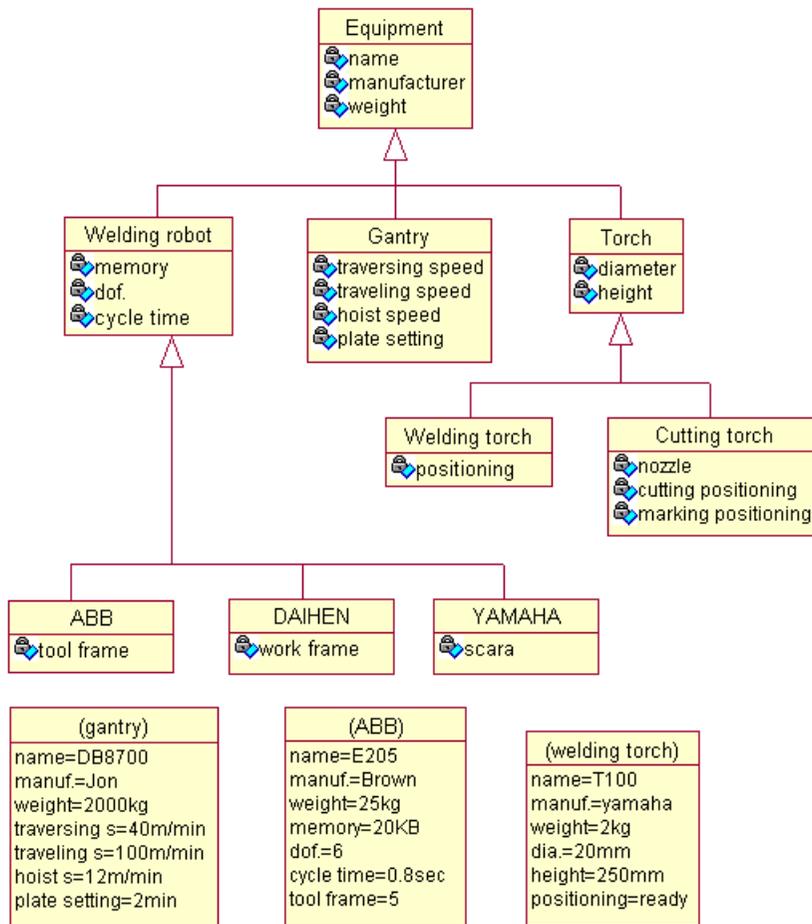


Fig. 3.12 A Multi-level Inheritance Hierarchy with Instances

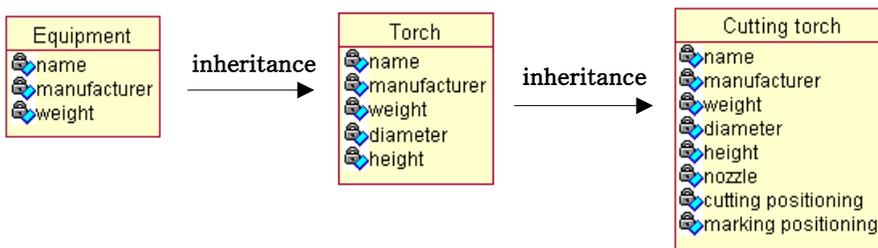


Fig. 3.13 An Idea of Inheritance

4. 선박조립공정의 분석 및 모델링

4.1 개요

본 연구에서는 객체지향 정보모델, SBD/SBM의 개념과 관련기술을 이용한 Digital Shipbuilding 과 관련한 선박조립공정의 소규모 블록의 조립공정의 시물레이션 및 최적화의 효과를 검증하였다. 이를 위해서 조립공정의 블록에 대한 형상 모델링을 수행하였으며, Fig. 4.1은 선박조립공정의 3차원 시물레이션 시스템의 구성도를 나타낸 것이다.[8]

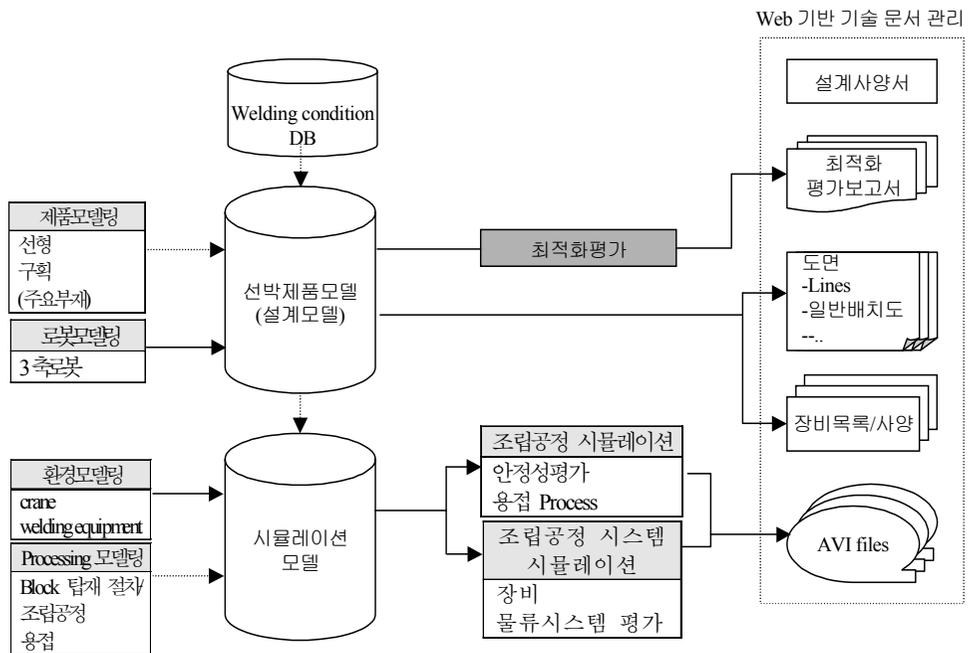


Fig. 4.1 System Configuration of Assembly Process of Ship

4.2 공정분석

복잡한 공정 및 객체를 효율적으로 모델링하기 위해서는 대상으로 하는 객체 내에서 각 일들이 가지는 특징과 다른 일들과의 연관성을 정확히 파악하고 정의해야만 한다.

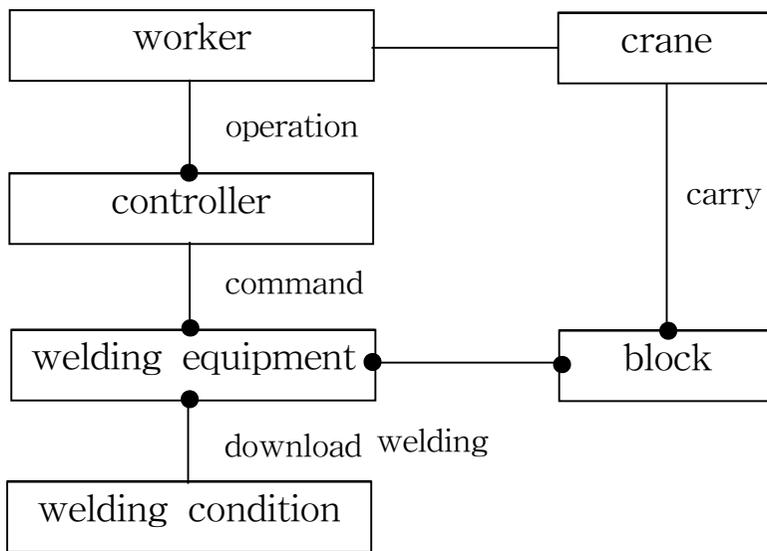


Fig. 4.2 Relation between Objects

공정분석은 실세계의 문제를 분석하여 이해하고 실제 문제를 해결하는 시스템 구현의 방법 및 가능성 등에 대한 판단을 하는 단계로서 모델링 하고자 하는 시스템의 행위를 추상적으로 기술하고 작업경로 전체를 처리되는 순서에 따라 각 공정을 기술한다. 부재의 종류 및 가공 특성, 물류 흐름 및 가공 능력 등을 분석하고 시나리오를 작성한다.[1]

Fig. 4.2는 조립공정에서 볼 수 있는 대표적인 객체들간의 관계를 보이고 있으며 조립공정의 시나리오에 대한 개략적인 flowchart는 Fig. 4.3에 나타낸다.

Component Process Model에서 주요 파라메터는 용접속도, 토치위치 및 지향각, 용접방향과 위빙, 등이며, Input은 Plate와 Stiffener, Output은 용접된 Block이다. Process Parameters는 아래와 같다.

Equipment/Process	Required Parameters
Gantry Crane	Gantry Speed [m/sec] Cross Travel Speed [m/s]
Buffer(Input, Output)	Capacity [pieces] Location [m*m*m]
Welding Machine	Welding Speed
Worker	Welding Speed Checking Time
Plate, Stiffener	Name, Location

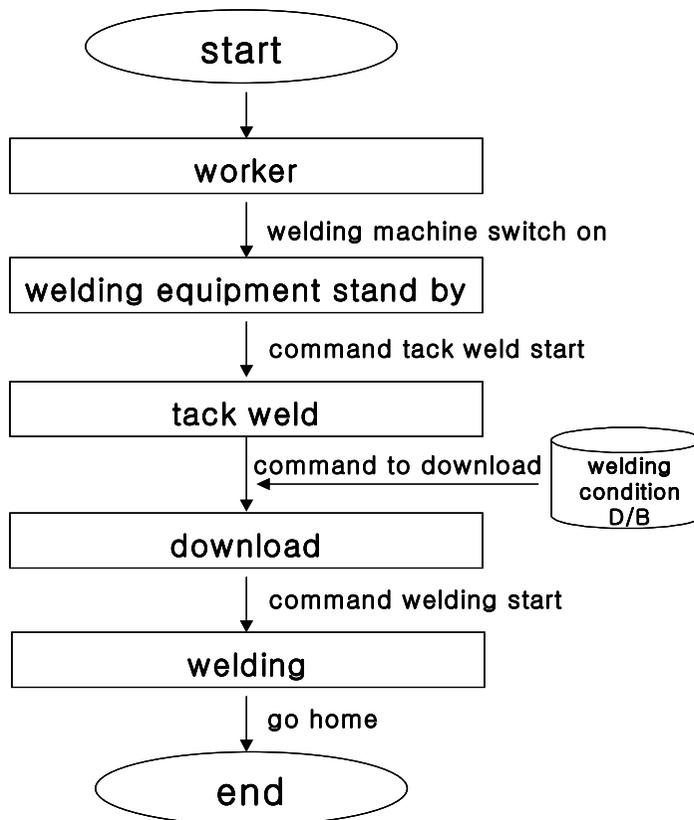


Fig. 4.3 Flowchart of Process

4.3 시스템 설계

시스템의 전반적인 구조에 대하여 결정하는 단계로서 각 서브 시스템들의 조직화와 그 기능에 대하여 결정 및 문제 해결을 위한 전략, 자원의 배치 등이 이루어진다. 시스템 설계 매니저는 자료 저장 관리에 대한 접근 방법을 선택하고 운송장비, 공장 건물 등의 글로벌 자원 접근 처리 방법을 규명해야 한다.[18]

4.4 객체 모델링

시스템 설계에서 결정된 모델에 대하여 상세한 내용을 모델링하는 단계이며 시스템에서 필요한 객체 및 클래스, 이들 객체들의 서로 구분될 수 있는 구체적인 값인 속성, 객체가 발생시킬 수 있는 사건이나 구체적인 동작인 기능 및 객체들의 상관관계에 대해서 모델링한다. 연산들은 구체적인 구현 방법을 모색해야 하고, 속성들도 구현하고자 하는 특성을 모두 반영할 수 있어야 한다.

모델링 단계에서 이용되는 모델은 객체 모델(Object model), 동적 모델(Dynamic model), 기능 모델(Functional model)이 사용된다. 이러한 모델링은 클래스 계층구조를 이용한 매우 강력한 클래스 재사용을 지원한다. 하위 클래스 생성은 소프트웨어 재사용 측면에서 볼 때, 두가지 중요한 의미를 갖는다. 첫째로 하위 클래스 생성은 기존의 클래스에 대한 수정을 매우 유연하게 지원한다. 두 번째는 기존의 클래스에 새로운 기능 추가가 용이하다는 것이다. 시뮬레이션 베이스가 방대해지면, CAD툴에서의 Modeling이 용이하지 못하다. 고품질의 시스템을 개발하기 위해서는 요구사항을 정확히 분석하고 목표로 하는 시스템을 설계하는 분석, 설계작업이 매우 중요하며 분석모델을 바탕으로 설계가 진행되고 설계모델을 기반으로 구현이 된다. 또한, 분석과 설계 단계에서 생성되는 다이어그램 및 모델링이 동일하며 객체 모델링 단계에서 생성하는 정보 모델은 선정된 객체지향 언어로 체계적으로 번역되어 코드가 생성될 수 있기 때문에 충분한 분석과 정확한 설계가 요구된다.[1][18]

(1) 객체 모델

객체의 정의, 정의된 객체 사이의 관계, 속성 및 기능을 기술하는데 사용되는 모델이다. 이러한 객체 모델은 객체 혹은 클래스들을 포함하는 그래픽 다이어그램으로 표기된다. 시스템에 요구되는 객체들의 관계(relationship)들을 보여줌으로써 시스템의 정적인 구조(Static Structure)를 포착하는 데 사용된다. 즉, 실제계에 가장 근접된 표현을 가능하게 하는 Modeling이다.

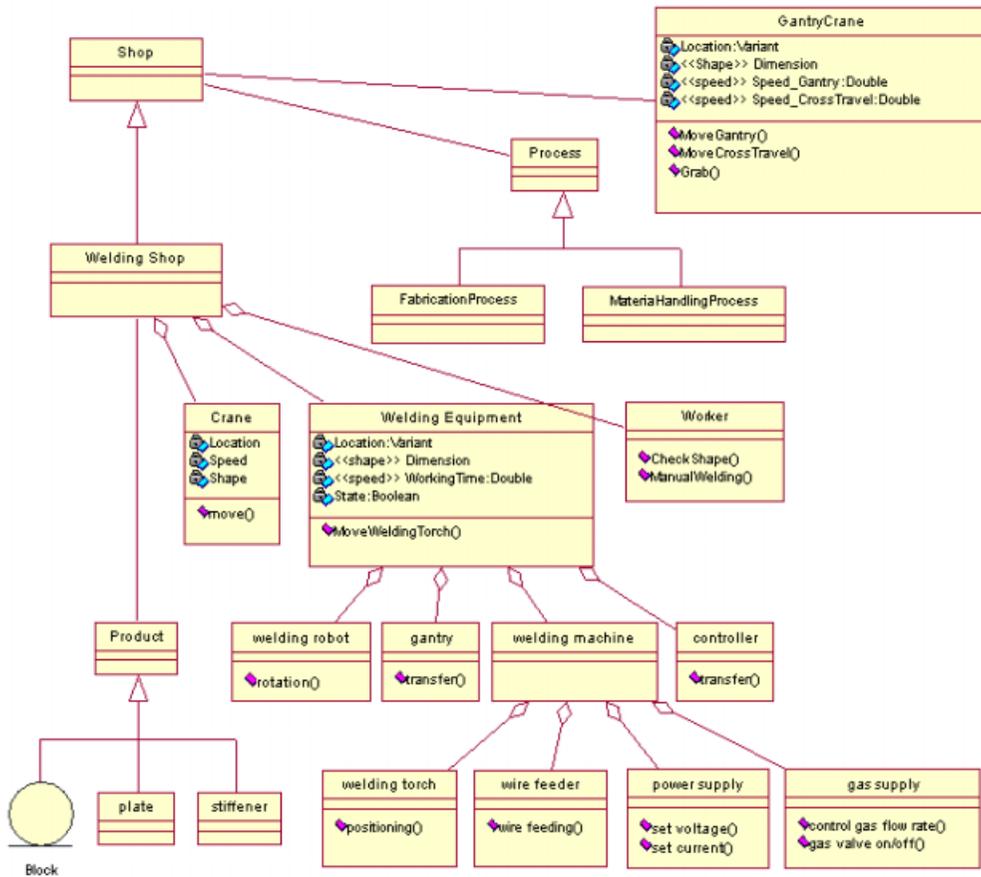


Fig. 4.4 Class Diagram of Subassembly

Fig. 4.4는 조립공정의 클래스 다이어그램이다. 다이어그램 내의 객체들은 3차원 모델링 또는 시뮬레이션 구현시에 CAD툴에서 Part 단위로 모델링 될 것이며,

그때의 각 객체들에 대한 속성과 관계를 정의하고 있다. Welding Robot의 경우를 보면 로봇의 관절은 각각의 Part 단위로 모델링된 후 Fig. 4.5의 로봇과 같이 조립된 형태로 구현되며, 객체모델에서 생성된 로봇에 대한 Property 정보 (Manufacturer, Material Density, Speed 등)는 Fig. 4.5의 좌·우측에 보여지는 정보들로 구체화된다. 이들중 독립된 움직임을 갖는 모델은 시뮬레이션 구현시

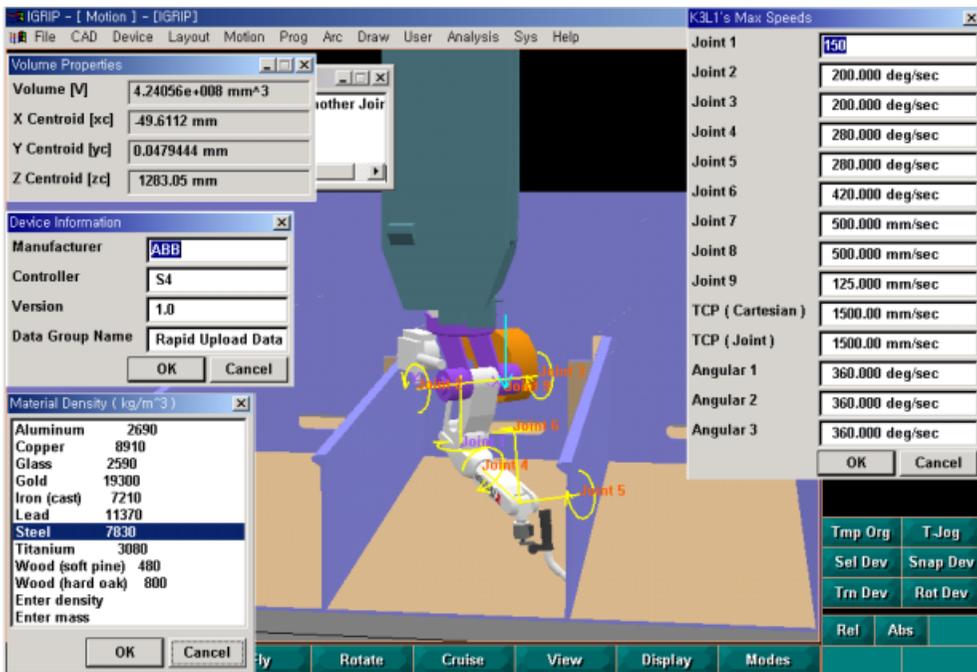


Fig. 4.5 Definition for Kinematics of Robot

Device의 형태로 구현되고, 객체모델에 기술되어 있는 기능과 객체들간의 관계가 바로 그러한 움직임의 형태로 구현된다. Fig. 4.5의 'gantry'의 'transfer' 기능은 동적모델의 단계를 거쳐 좀 더 구체화되며, 시뮬레이션을 위해 kinematics 조건들을 설정하는 단계에서 이전 동적 모델에서 규명된 기능들을 'traversing speed', 'traveling speed', 'hoist speed'의 형태로 구체화한다. 또한, 시뮬레이션에서 Crane의 'move'기능은 구체적인 행위를 기술하는 code의 형태로 구현되고, code는 선택된 언어((비주얼)C++, GSL, CLI) 등으로 구현 될 것이다.

(2) 동적 모델

시간의 흐름을 고려한 시스템의 흐름과 기능들의 동작 순서를 기술하기 위한 모델이다. 전형적인 일련의 상호행동에 대한 시나리오를 준비하고, 객체들 사이의 이벤트를 규명하며, 시나리오에 대한 Event Trace를 실시한다. 시나리오를 상태도로서 접목한 State Diagram을 구성하고 객체들 사이의 사건을 대응시켜 시스템 계층간의 완결성과 일치성을 확인한다.

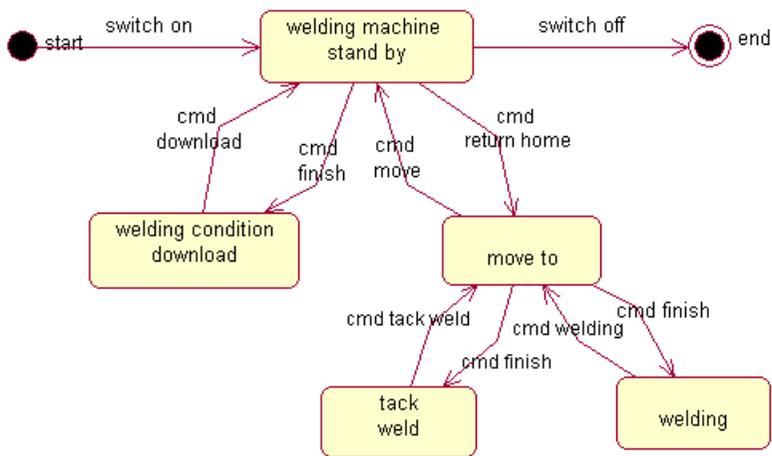


Fig. 4.6 Welding Machine State Diagram

Fig. 4.6은 Welding Machine의 상태도로 Welding Machine의 입장에서 바라본 각각의 상태와 상태사이의 관계를 보이고 있으며 Fig. 4.7은 Gantry Crane의 상태도이다. 또한, 시나리오는 상호작용(Interaction) 다이어그램으로 표현된다. 시퀀스 다이어그램은 바로 시나리오의 그래픽적 뷰이며 시간적인 순서로 정렬된 객체들의 상호작용을 보여준다. 이것은 시나리오의 기능을 수행하기 위해 필요한 객체, 클래스와 그들 사이에 존재하는 메시지를 갖는다는 특징을 가진다. 각 사건들은 객체 모델링에서 규명된 객체들 사이의 정보 흐름을 나타내게 되고 각 사건의 정보를 보내는 객체와 정보를 받는 객체가 밝혀진다. 사건의 순서와 사건을 주고받는 객체들은 Fig. 4.9와 같이 사건 추적도에 나타나게 되며

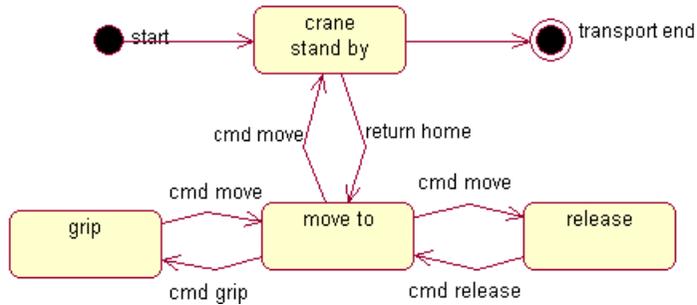


Fig. 4.7 Gantry Crane State Diagram

수직선은 객체를 표현하고 수평 화살표는 사건의 흐름을 나타낸다. 수직선의 위에서 아래의 순서로 시간을 나타낸다. 이러한 동적 다이어그램은 시뮬레이션 구현시에 프로그램 코드의 프로시저 형태로 반영된다.

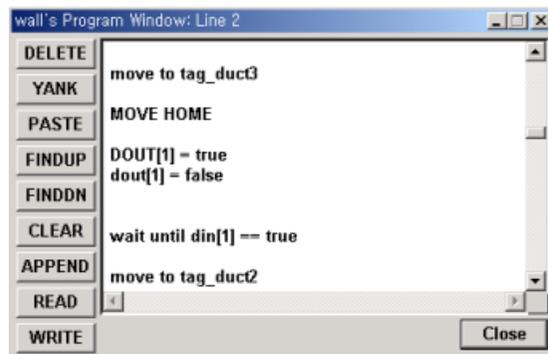


Fig. 4.8 I/O Programming

두 객체사이의 신호를 주고 받는 상황의 프로그래밍의 예가 Fig. 4.8이며, 여기에서 신호를 주는 표현은 'DOUT[1]=true'으로, 신호를 받는 표현은 'wait until din[1]=true'로 구현되고 있으며 []안의 번호는 통신포트를 의미한다.

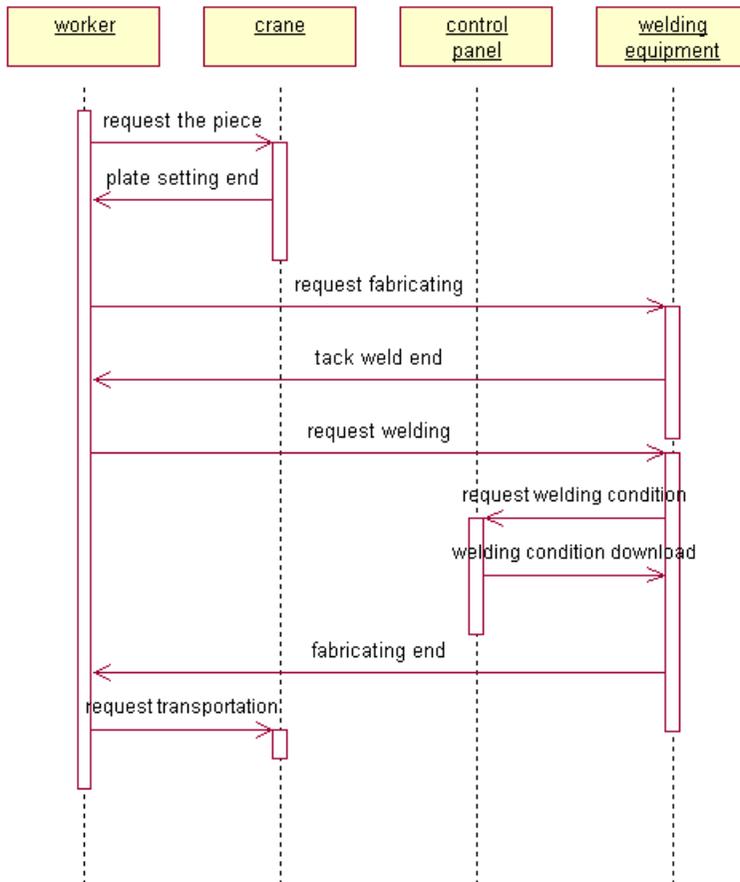


Fig. 4.9 Event Tracing

(3)기능 모델

시스템의 값의 전송이나, 함수, 매핑 등을 기술하기 위한 모델이다. 값들과 다른 값들과의 관계 그리고 이들과 관련된 함수들에 대해 나타낸다.

5. 선박조립공정의 3차원 시뮬레이션 구현

분석과 설계를 거쳐 정의 및 모델링된 객체에 대해서 클래스의 속성, 기능, 상관관계를 실세계의 문제를 해결할 수 있는 적절한 툴을 사용하여 구현하게 된다. 본 연구에서는 조선소의 성형공장 내 소조립 공정을 대상으로 구현하였다.



Fig. 5.1 Subassembly Process

5.1 그래픽 사용자 인터페이스

많은 CAD 시스템 뿐만 아니라 시뮬레이션 시스템들은 시스템의 유연성과 작업 효율성을 높이기 위해서 시스템에 적합한 그래픽 사용자 인터페이스(GUI : Graphical User Interface)를 사용하고 있다. 일반적으로 그래픽 사용자 인터페이스는 입력의 가시화, 각 모듈간의 유연한 접속, 중간과정 및 최종결과의 가시화 측면을 고려하여 설계, 개발되어진다. 또한 그래픽 사용자 인터페이스 설계를 통해서 구현하고자 하는 시스템의 기능과 시스템의 효율성을 개발단계에서 어느 정도 검증할 수 있다.

모델 기반 선박조립공정 시뮬레이션 시스템의 프로토타입에 대한 그래픽 사용자 인터페이스는 상용 시뮬레이션 도구인 ENVISION, IGRIP, Virtual NC 시스템에서 제공하는 것을 활용하며, GSL(Graphic Simulation Language), CLI(Command Line Interpreter) 함수들을 이용하고자 한다. 각각의 명령어나 시뮬레이션에 필요한 기능들이 메뉴로 정의되며, 최상위 명령어가 최상위 메뉴가 된다. 하나의 명령어에 속하는 세부명령어들은 풀다운 메뉴로 구성되어진다.

Fig. 5.2 는 모델 기반 조립공정 시뮬레이션 시스템을 개발하기 위한 메뉴들의 계층이다. [DEVICE] 메뉴는 선박조립공정 시뮬레이션을 위해서 이미 CAD 시스템에서 모델링 되어 데이터베이스에 저장되어 있는 부재의 형상, 구획, 크레인 및 용접로봇 형상들에 대한 정보를 읽어 오는 곳이다. [LAYOUT] 메뉴는 WORKCELL을 꾸미기 위해 필요한 환경변수들을 지정하고 각각의 DEVICE들을 정의한다. [ANALYSIS] 메뉴는 조립공정 시뮬레이션 프로토타입 시스템을 수행하는데 필요한 해석 및 계산을 수행한다. 시뮬레이션 과정에서 CAE 프로그램 수행에 필요한 데이터를 생성하는 방법은 ENVISION/IGRIP 시스템에서 제공하는 GSL 함수를 이용한다. [MOTION] 메뉴에서는 DEVICE, 프로그램들과 MOTION으로 이루어진 WORKCELL을 동시에 동작시키고 시간에 따라 그 동작을 분석할 수 있게 하였다.

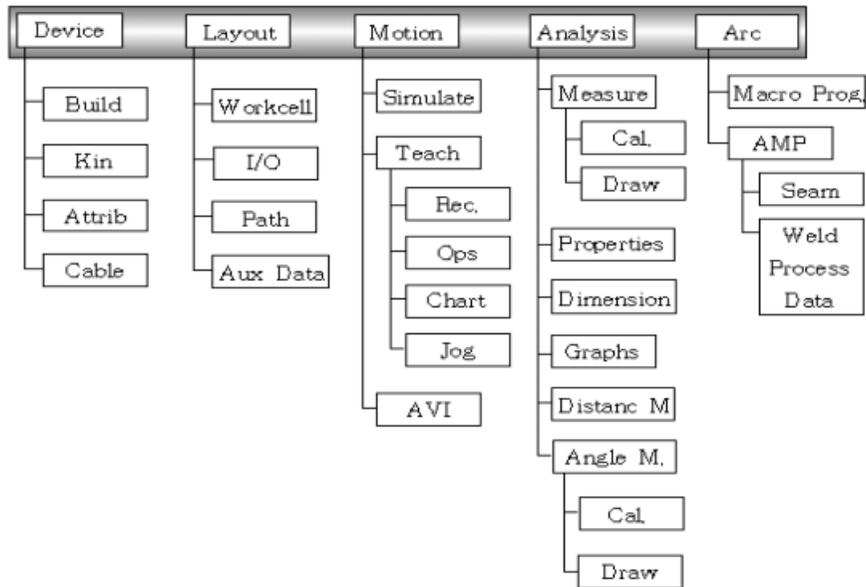


Fig. 5.2 Hierarchy of Menu



Fig. 5.3 Joint Travel Limits

Fig. 5.3은 [DEVICE] 메뉴의 서브 메뉴인 [ATTRIB] 메뉴를 선택하였을 경우에 나타나는 화면이다. 그림에서 보듯이 GEOMETRY 상에서 선택한 부재의

PARAMETER 들의 DEFAULT 값으로 가시화되지만 사용자가 원하는 경우에는 그 변수 값들을 수정할 수 있으며, Fig. 5.4 은 Volume Properties를 보여주고 있다.

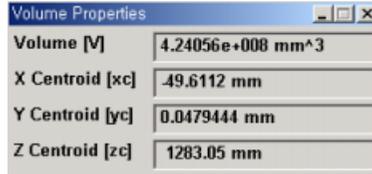


Fig. 5.4 Volume Properties

MATERIAL OF ROBOT의 값은 DEFAULT로 STEEL이 SETTING 되어 있지만, Fig. 5.5 과 같이 선택 수정할 수 있다.

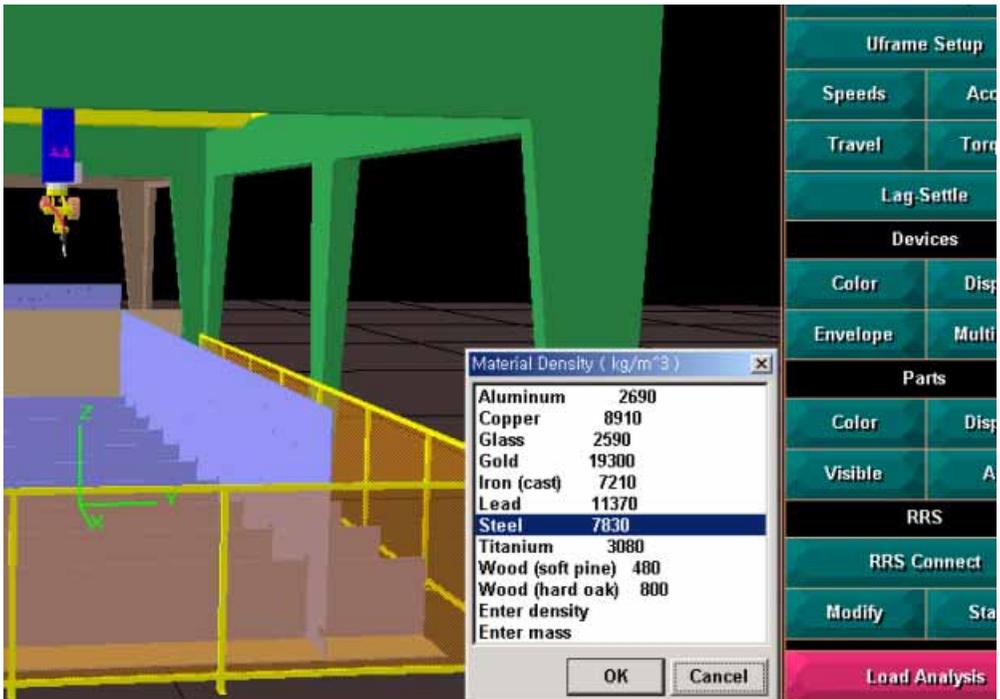


Fig. 5.5 Material Density



Fig. 5.6 Max Speeds

Fig. 5.6은 [DEVICE] 메뉴의 서브 메뉴인 [SPEEDS OF ATTRIB] 메뉴를 선택하였을 경우에 나타나는 화면이다. 이 화면에서는 CRANE/GANTRY DEVICE 에 관련된 ATTRIBUTE 들을 설정하여 CRANE/GANTRY의 형식, 이동속도, 이동 가속도, 구속조건들을 정의할 수 있다. 선박조립공정 시물레이션 프로토타입 시스템에서는 시물레이션내의 각종요소들(블록/구획/로봇모델, CAE 입/출력 데이터 파일, 구동환경 파일 등)을 분류하여 그 정보들을 DIRECTORY 별로 저장하였다. 여기서 말하는 DIRECTORY는 UNIX나 DOS 상에서의 DIRECTORY 개념뿐만 아니라 시물레이션 과정에서 발생하는 내부 데이터 및 외부 데이터들을 저장하기 위한 가상적인 계층구조를 포함한 개념이다.

5.2 선박조립공정의 구현 예

선박 조립공정의 블록 및 로봇형상은 조립공정 시뮬레이션을 보다 효율적으로 수행하기 위해서 상용 CAD 시스템을 이용하여 3차원 곡면으로 모델링 하였으며, 구획배치 모델링은 정의하고자 하는 구획을 둘러싸고 있는 경계곡면들을 정의하고 그 다음에 경계곡면들로 이루어진 구획을 정의하는 순서로 진행하였다. 경계 곡면은 평면 혹은 선체외판과 같은 자유곡면이 사용되어진다. 로봇 모델링은 Gantry에 붙어 블록에 용접을 하는 용접로봇에 대한 정보를 모델링하는 것으로 부착되는 로봇의 개수는 2기이며 모두 Gantry하단에 부착되도록 객체를 3D CAD 툴로 part 단위 모델링 한 후 시뮬레이션 툴에서 불러 들여 각 Part Model들을 Fig. 5.7 과 같이 결합한다.

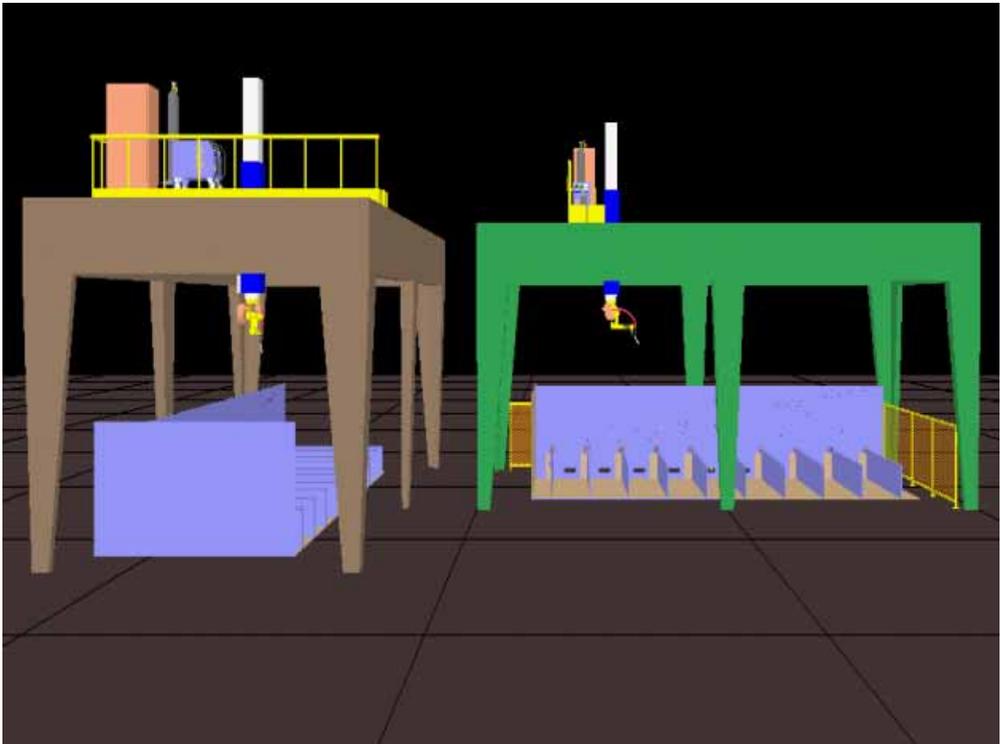


Fig. 5.7 Visualization of assembly

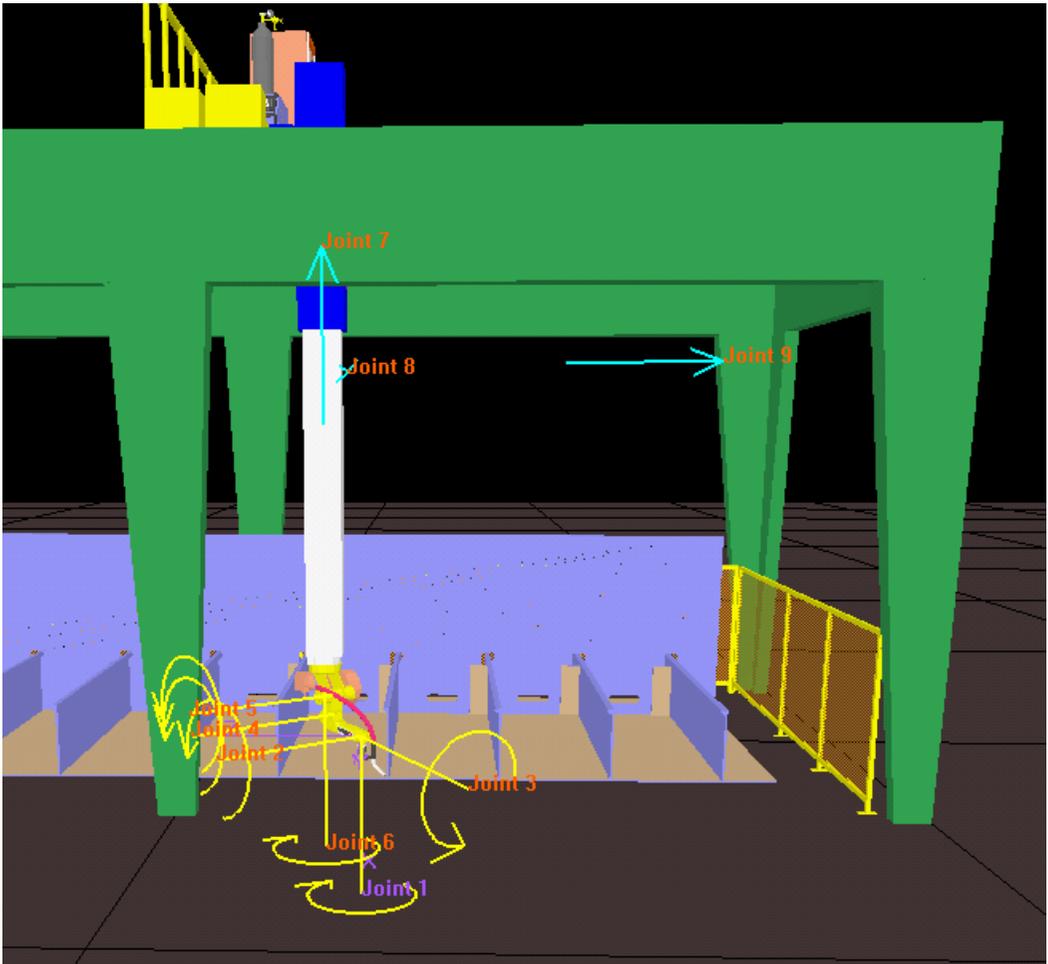


Fig. 5.8 Product synthesis and analysis

그리고 Fig. 5.8 에서와 같이 Gantry 와 Robot body 의 자유도 및 각 관절의 특성치를 정의하고, 결합된 모델들 간의 간섭을 확인한다. 로봇의 경우, 이 과정에서 로봇관절의 회전 범위 또는 한계 이동범위를 정의하고, Jog 모드에서 로봇 관절간의 간섭을 확인할 수 있다. 관절이 정의된 범위를 벗어나면, 이는 로봇관절의 파손이라 판단할 수 있으며 Fig. 5.9와 같이 Part의 color가 변함으로써 사용자에게 가시화 시켜준다.



Fig. 5.9 Joint Setting

또한, 로봇과 부재간의 Collision Check가 가능하고, 그 결과에 따라 Part의 Rebuilding이 가능하다.

다음으로 시뮬레이션에 필요한 Kinematics 조건들을 부여한다.

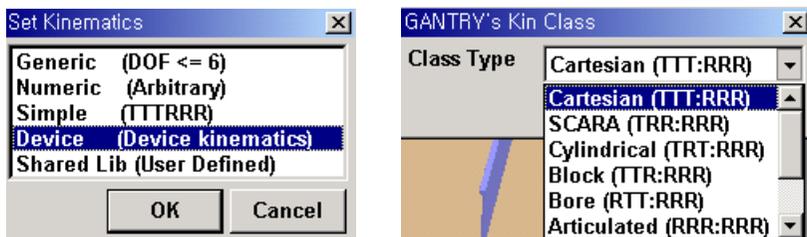


Fig. 5.10 Set Kinematics

Gantry에 붙어 있는 Robot의 Kinematics 의 Class는 Cartesian(TTT:RRR)으로

한다. T는 Translation을, R은 Rotation을 의미한다. 따라서 Fig. 5.10 에서 선택한 Kin Class는 3개의 Translation과 3개의 Rotation을 갖는 것을 의미한다. 이러한 조건설정이 끝나면 Fig. 5.11과 같이 구체적인 행위를 구현하기 위해 프로그래밍을 한다. (비주얼)C++, GSL, CLI를 이용해서 이미 생성한 정보 모델에 기반한 시뮬레이션을 구현한다. GSL의 Logic은 Pascal 기반이며, C++이나 Open GL등으로 사용자 인터페이스를 만들어 사용할 수도 있다. Fig. 5.11과 같은 프로그램은 직접코딩해도 무방하나 사용자의 편의를 위해 Command 버튼의 클릭으로도 프로그래밍이 가능하다.

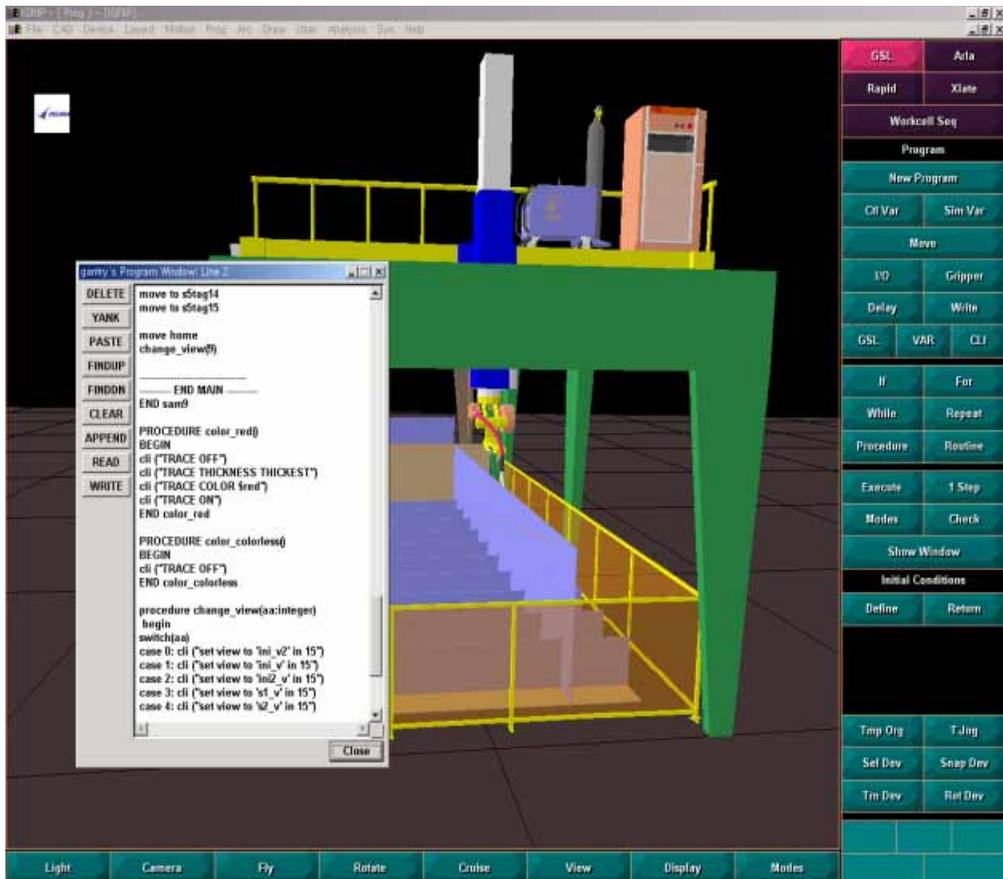


Fig. 5.11 Programing

tooldata	20 Parameters
wobjdata	18 Parameters
WELDDATA STRUCTURE DEFINITION	11 Parameters
SEAMDATA STRUCTURE DEFINITION	29 Parameters
WEAVEDATA STRUCTURE DEFINITION	15 Parameters
welddata	12 Parameters
seamdata	30 Parameters
weavedata	16 Parameters

Manufacturer	ABB
Controller	S4
Version	1.0
Data Group Name	Rapid Upload Data
<input type="button" value="OK"/> <input type="button" value="Cancel"/>	

	Speed (%)	Motype	Termttype	Round	Cfg	T1
stoppath	100.000%	Default	Default	-1.000	0	0
stoptag1	100.000%	Default	Default	-1.000	0	0
stoptag2	100.000%	Default	Default	-1.000	0	0
stoptag3	100.000%	Default	Default	-1.000	0	0
stoptag4	100.000%	Default	Default	-1.000	0	0
stoptag5	100.000%	Default	Default	-1.000	0	0
stoptag6	100.000%	Default	Default	-1.000	0	0
stoptag7	100.000%	Default	Default	-1.000	0	0
stoptag8	100.000%	Default	Default	-1.000	0	0
stoptag9	100.000%	Default	Default	-1.000	0	0

Fig. 5.12 Device Information

이렇게 프로그래밍한 후 작업 프로세스에 있어서 사이클 타임을 단축시키기 위해 충돌이 전혀 없는 로봇 궤적을 생성하고 최적화를 수행한다. 그리고 시뮬레이션 기록은 전사적인 작업과 프로세스 가시화용으로 Fig. 5.13과 같이 AVI 파일을 생성하여 활용할 수 있다. 또한, 향후 재활용을 위해 시뮬레이션 결과 및 Data, 그리고 모델링된 Part나 Devices Library를 Fig. 5.12와 같이 데이터 베이스에 저장한다.

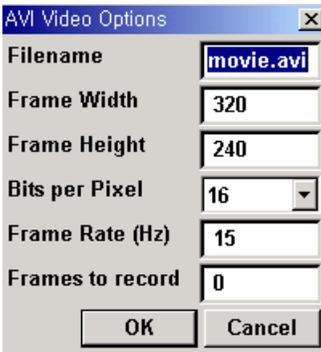
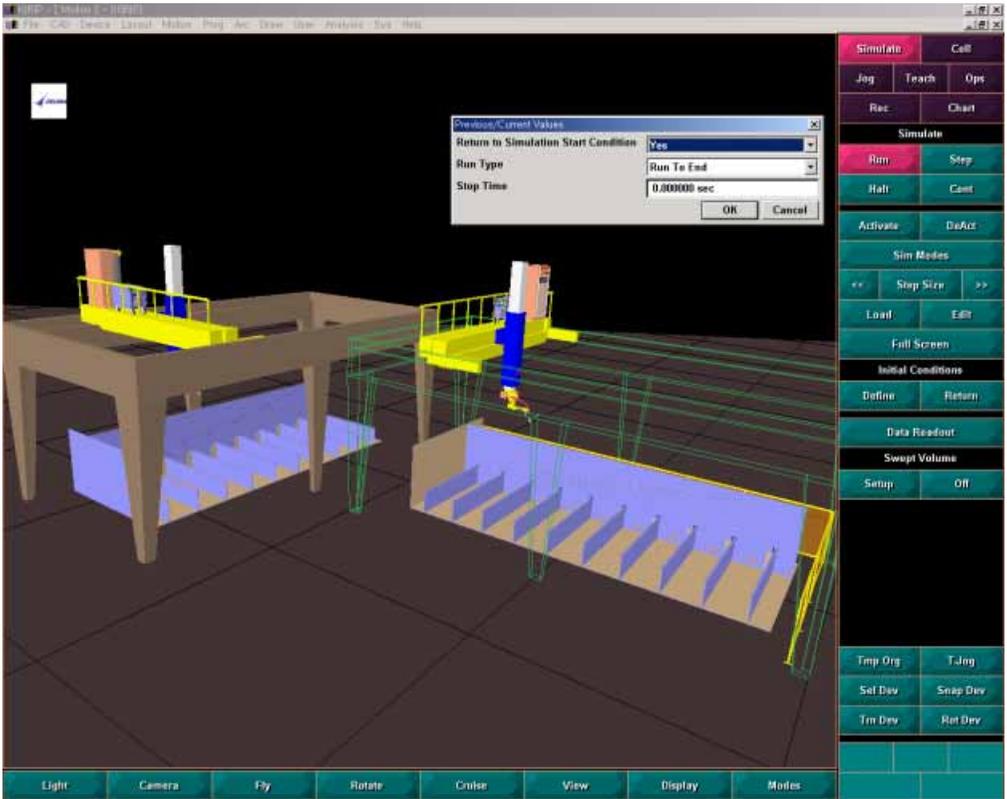


Fig. 5.13 Visualization of 3D Simulation

Fig. 5.13은 프로세서 가시화용으로 사용할 AVI동영상 파일을 생성하는 과정이

다. 또한 웹기반의 VRML파일로도 활용가능 하다. Fig. 5.14는 IGRIP를 사용한 시스템 구현절차를 나타내고 있으며, 선박조립공정 시물레이션 시스템 수행시 고려되어지는 사항들은 다음과 같다.

1. 블록, 로봇, 크레인 등을 모델링하고, CAE 프로그램 등을 이용하여 선박조립 공정 시물레이션을 수행한다.
2. 선박조립공정 순서 및 Robot 의 이동거리 등과 같은 제약조건들은 정의된 값을 벗어나지 못한다.
3. 선박조립공정 시물레이션 도중에 교차기법을 이용하여 로봇과 블록의 충돌검사(Collision Check)를 수행한다.

또한, 선박조립공정 시물레이션의 결과로 선박조립공정 순서도, 블록조립에 소요되는 시간, 블록과 로봇과의 Collision, Crane 및 Gantry의 최적배치 등이 생성된다.

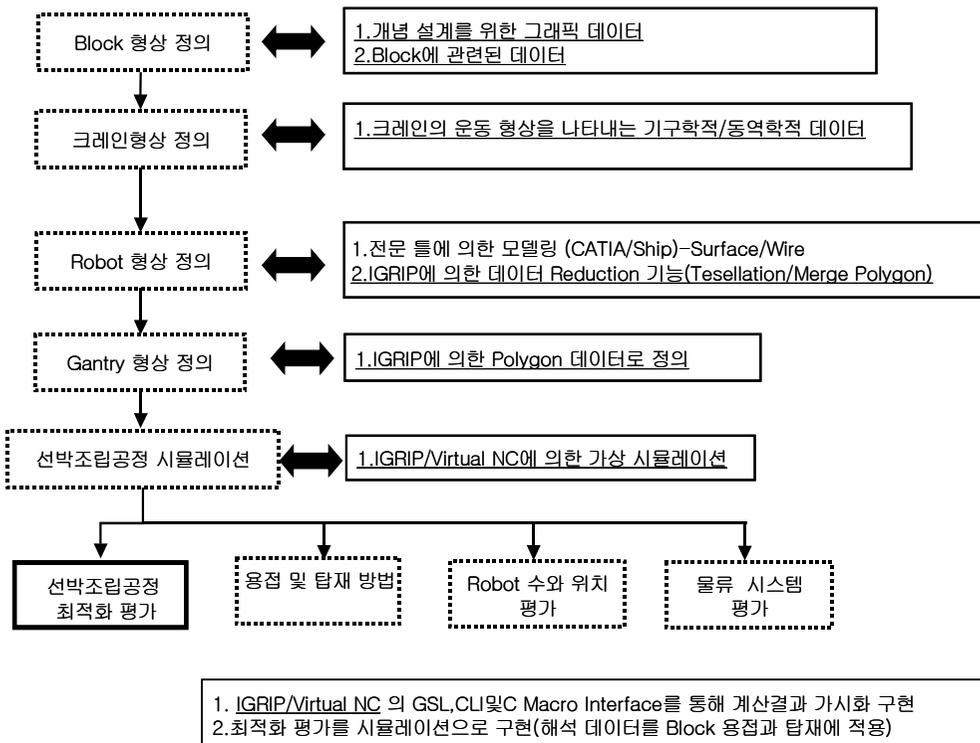


Fig. 5.14 Procedure for Simulation

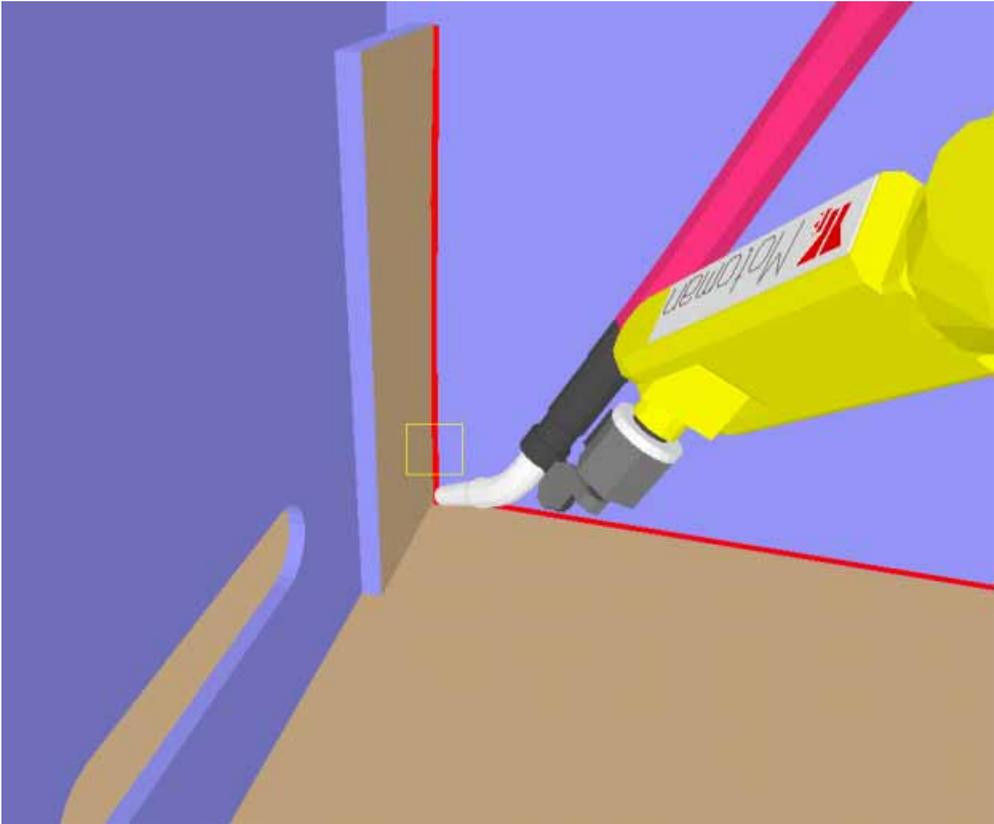


Fig. 5.15 Collision Check

Fig. 5.15 는 수평용접에서 수직용접으로의 전환, 다시말해 용접자세의 변화에 대한 용접로봇과 부재의 Collision 및 용접로봇의 거동을 확인하는 과정이며, Fig. 5.16은 모델링 결과 및 선박조립공정 과정을 시뮬레이션 한 결과를 나타내고 있다. Fig. 5.16에서 알 수 있듯이 Robot의 Cycle Time을 측정하고 비교할 수 있으며, 용접선 하나에 대한 경과 시간도 체크가 가능하다. Fig. 5.16은 두가지의 경우에 대해서 시뮬레이션 한 예이다. 하나는 수직용접의 방향을 상향으로 한 경우이고, 또 다른 하나는 수직용접의 방향을 하향으로 한 경우이다. 본 시뮬레이션에서는 수직용접의 경우 상향대비 하향의 속도를 1:3의 비율로, 또 같은 상황에서 하향의 경우를 멀티패스한 조건 등의 경우로 시뮬레이션 하였다.

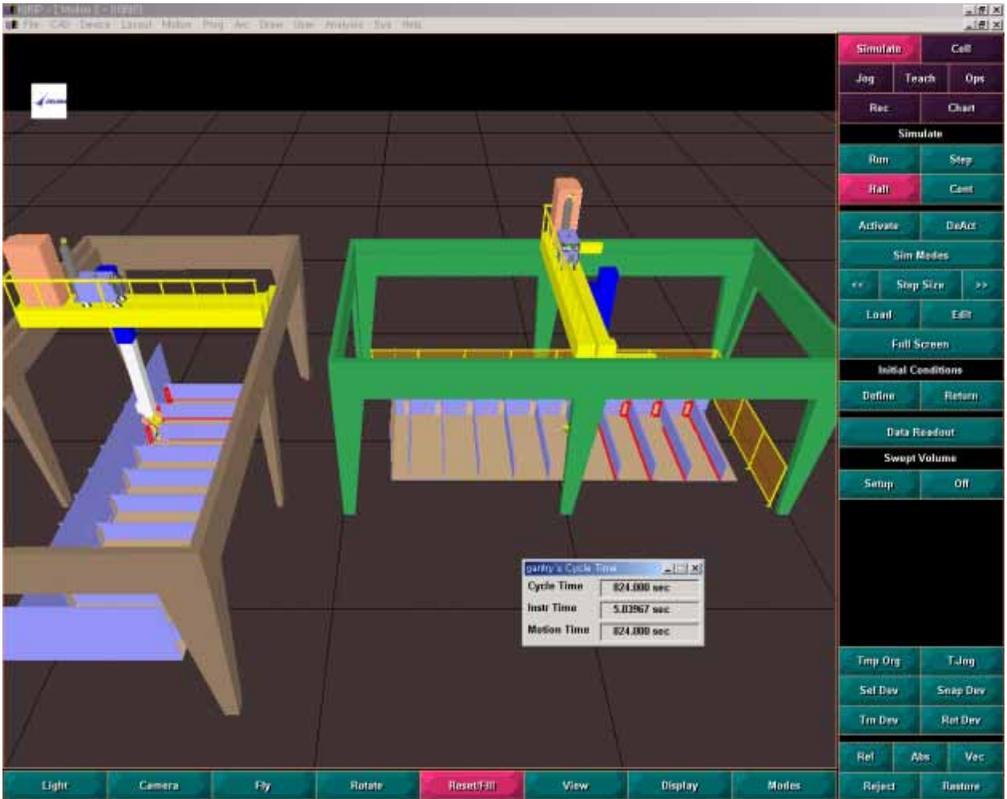


Fig. 5.16 Simulation for Assembly Process of Ship

이러한 시뮬레이션을 통해 Fig. 5.15에서와 같이 로봇의 거동과 부재와의 충돌 그리고 최적화된 용접선 등은 알 수 있지만 용접을 함에 따른 부재의 변화에 대한 용접자세의 변화와 그때의 최적화 등에 대해서는 그러한 경우에 대한 데이터베이스 구축이 부족한 때문에 구현함에 어려움이 있었다. 하지만 본 연구를 통해 조립공정의 최적화 및 시뮬레이션의 효과는 충분히 확인하였다.

6. 결론

본 연구에서는 객체지향 정보모델 및 시뮬레이션 기반 생산 개념과 관련 기술을 정리하였으며, 이를 이용한 Digital Shipbuilding과 관련한 선박조립공정의 소규모 블록의 조립공정의 시뮬레이션 및 최적화의 효과를 검증하였다. 이러한 연구를 토대로 하여 다음과 같은 결론을 얻었다.

1. 객체지향 모델을 기반으로 기존의 분산되어 있던 시스템 개념 기술 및 정보를 통합함으로써 실제화하였다.
2. 객체 모델링을 통하여 구축된 조선소의 주요 제조 설비의 객체 정보를 가상 제조(Digital Manufacturing) 기술을 이용하여 구현하였다.
3. 공정분석의 순차도(Sequence Diagram)를 이용하여 자재 물류 흐름의 가상 생산 및 디지털 시뮬레이션을 수행함으로써 선박 건조 공정(Process)의 디지털 시뮬레이션이 가능하였다.
4. 시뮬레이션을 통해 생산성 측정, 간섭 확인, 실시간 성능분석, 물류 흐름 분석과 제어를 통하여 획기적인 생산성 향상을 기대할 수 있으며, 인간공학 분석 적용, 유지보수 및 훈련 등이 가능하다.
5. 다양한 대용량 데이터의 가시화 및 작업공정의 유효성 판단과 최적화가 가능함을 확인하였으며, 가상현실 환경에서 동작 포착이 가능하기 때문에 장비와 작업자 동작간의 상관 관계를 빠르게 전개하고 분석할 수 있게 되었고, 공정계획 모델, 공정흐름 분석 등과 연결됨으로써 보다 진보한 디지털 매뉴팩처링이 가능하게 되었다.

선박 생산은 시제품의 제작이 현실적으로 불가능하기 때문에 공정의 시뮬레

이선이 실제 공정에 선행될 수 있다면 생산 효율성의 증대에 큰 기여를 할 수 있을 것이다. 하지만 이러한 공정과 공장의 시뮬레이션을 위해서는 막대한 작업량이 필요할 뿐 아니라 정확한 시뮬레이션을 위해서는 전산화된 고 정도의 생산 정보가 필요하다.

본 연구를 통해서 발생된 문제점 및 아직 개발이 미진한 선박조립공정 방법 평가 및 블록 탑재에 따른 크레인/젠트리의 최적위치 평가 등과 같은 부분들은 향후 계속적으로 보완할 것이다. 또한, 선박조립 공정 시뮬레이션 시스템에는 외부환경에 대한영향을 고려하지 않고 있지만 추후 확장되는 시스템에는 이러한 기능을 반영할 계획이며, 좀더 실제적인 선박 생산 공정을 구현할 예정이다.

참 고 문 헌

- [1] 박주용, 박현철, 차태인, “객체지향 정보모델에 기반한 선박조립공정의 3차원 시뮬레이션에 관한 연구”, 대한조선학회 추계 학술대회 논문집. 2001
- [2] 신중계, 이장현, 우종훈 2001 “디지털 선박 생산,” 대한조선학회 춘계 학술대회 논문집, pp. 8-13
- [3] 신중계 2001 “디지털생산의 개념과 조선산업 응용,” 제2회 용접공정연구위원회 기술세미나, pp. 1-13.
- [4] James Rumbaugh, Michael Blaha, William Premerlani, Frederick Eddy, William Lorensen, Object-Oriented Modeling and Design, Prentice-Hall, Inc., 1991, pp 1-83, pp 278-295
- [5] Jones, G. and Hankinson, T., "Simulation Based Design for ShipDesign and Acquisition", ICCAS'94, Bremen, Germany
- [6] Jones,P., 'virtual Environments in the Development of Ships', ICCAS'94, Bremen, Germany
- [7] Andrea Favretto, Fincantieri S.p.A., "Automatic Robot Programming for Welding of Ship Structures:Weld Data Generation and Usage for Collision Free Robot Paths", ICCAS'99, Trieste Italy
- [8] 이순섭, 이종갑, 김홍태, “SBD 기법을 이용한 컨테이너 적·하역 시뮬레이션, Journal of Ships & Ocean Engineering Vol31 February 2001
- [9] 이순섭 외, “컨테이너 적·하역에 따른 선박의 안정성 시뮬레이션”, 대한조

선학회 추계연구발표회 논문집, 1999

- [10] 김홍태, 이종갑, 유병세, 이순섭, 임재민, “Modeling & Simulation 기술을 이용한 컨테이너 적하역 안정성 평가”, Journal of Ships & Ocean Engineering Vol31 February 2000
- [11] 이종갑 강원수, “시뮬레이션 기반 선박 설계기술”, Journal of Ships & Ocean Engineering Vol21 December 1995
- [12] “공학시뮬레이션을 위한 협동형 가상 프로토타이핑 기술“ Journal of Ships & Ocean Engineering Vol27 December 1998
- [13] Review of Object-Oriented Simulation The Korea Society for Simulation '98 춘계학술대회 논문집 1998, 5 pp.1-16
- [14] Design and Evaluation of Object-Oriented Simulator for Manufacturing 한국시뮬레이션학회 1997년 춘계 학술대회 발표집 1994. 11 pp. 13-17
- [15] “객체지향 기법을 이용한 공장운용 환경 하에서의 혼합시뮬레이션 구현“ Journal fo the Korea Society for Simulation 1998, 6 v.7, n.1, pp.15-26
- [16] 임장곤, 박주용, 2000 “객체지향 모델을 활용한 웹기반 조선용접정보시스템,” 대한조선학회 추계 학술대회 논문집, pp. 21-24
- [17] 이경철, “객체 모델링 기법을 적용한 선박검사 정보 시스템에 관한 연구”, 한국해양대학교 대학원 석사논문, 2000
- [18] 임장곤, “객체 지향 모델을 활용한 웹기반 조선용접정보 시스템에 관한 연구”, 한국해양대학교 대학원 석사논문, 2001