

공학석사 학위논문

생산 시뮬레이션을 통한 용접설비 및 공정변경에
따른 생산성 향상에 관한 연구

**A Study of Improvement of Productivity by Change of Welding Facility
and Process Variable Using Production Simulation Technology**

지도교수 박 주 용

2006년 2월

한국해양대학교 대학원

해양시스템공학과

최 동 환

本 論 文 을 최 동 환 의 工學碩士 學位論文으로 認准함.

위원장 : 공학박사 조 효 제

위 원 : 공학박사 박 석 주

위 원 : 공학박사 박 주 용

2006년 2월

한 국 해 양 대 학 교 대 학 원
해 양 시 스템 공 학 과 최 동 환

목 차

List of Tables	II
List of Figures	III
Abstract	IV
제 1 장 서 론	1
1.1 연구 배경 및 필요성	1
1.2 연구 내용	2
제 2 장 조선 Panel Line 공정 분석과 문제 정의	3
2.1 Panel Line의 설비 및 공정 소개	4
2.2 공정 시뮬레이션 제약 조건	7
2.3 공정 IDEF \emptyset 모델링	10
2.4 공정 소요시간 정식화 및 세분화	12
제 3 장 시뮬레이션 모델링	13
3.1 공정 시뮬레이션에 대하여	13
3.2 Panel Line 공정 시뮬레이션 시나리오	14
3.3 QUEST를 활용한 공정 시뮬레이션	16
3.4 범용 개발언어에 의한 공정 시뮬레이션	20
제 4 장 시뮬레이션 조건 변화와 시뮬레이션 결과 고찰	25
4.1 분기점에서의 로드 밸런싱	25
4.2 공법이나 설비 변경시 생산성의 변화 예측	27
4.3 물류 투입순서 변경시 생산성의 변화 예측	29
4.4 시뮬레이터 활용 방안	31
제 5 장 결 론	32
참고 문헌	33

List of Tables

Table 1 Characteristic of Job Stage	9
Table 2 Table of Real Welding Speed used from Empirical Formula	9
Table 3. Comparison of Working Time according to Real Time and Simulation Time	25
Table 4. Comparison of Dispatching Count according to Normal and SRPT Rule	26
Table 5. Comparison of Total Work Time according to Change of Welding Option	28

List of Figures

Fig. 1 Panel Line Layout	3
Fig. 2 Tack Welding Stage Scene & Plates Arrangement with Oil Pusher	4
Fig. 3 SAW Gantry	5
Fig. 4 Turn Over Scene	5
Fig. 5 NC Marking Scene	6
Fig. 6 Margin Space Cutting	6
Fig. 7 Welding Head Arrangement of Complex Machine	7
Fig. 8 Function Modeling of Whole Stage using IDEF \emptyset	10
Fig. 9 Function Modeling of Tack Welding Stage using IDEF \emptyset	11
Fig. 10 Function Modeling of Modification & Finish Stage using IDEF \emptyset	11
Fig. 11 Empirical Formula by Time Analysis	12
Fig. 12 Input Data & DB Architecture	16
Fig. 13 Flowchart and realized SCL Code for confirms the Thickness of the Plates	19
Fig. 14 Information Flow of Simulation System	19
Fig. 15 Layout of Panel Shop Simulator Using QUEST	20
Fig. 16 Web Based Panel Shop Simulator using Development Language	20
Fig. 17 Class Diagram Modeling of Job Stage in Panel Line	21
Fig. 18 Class Diagram Modeling of Whole Panel Line Simulator	22
Fig. 19 Architecture of Deploying a Smart Client Application	23
Fig. 20 Various Simulation Results	24
Fig. 21 Results of Load Balancing using SRPT Rule	26
Fig. 22 Utilization of Job Stage for escorting a Cutting Stage using SRPT Rule	27
Fig. 23 Utilization Change according to Welding Case	29
Fig. 24 Change of Productivity using Johnson's Rule	30

A Study of Improvement of Productivity by Change of Welding Facility and Process Variable using Production Simulation Technology

Dong-Hwan, Choi

Division of Ocean Systems Engineering
Graduate School of Korea Maritime University

Abstract

Digital manufacturing technology innovates the production system in the various industries. Shipbuilding industry also obtains the improvement productivity and the cost reduction. This paper aims to optimize the schedule planning of shipbuilding assembly process. In this paper the facilities and production process of panel line were modeled in the virtual environment. The variations of production simulation were investigated by changing the capacity of the facilities and the order of manufacturing process in the digital shop. The Simulator of Discrete Event base was developed with the tool and general development language. The simulation results gave the detailed analysis of the given schedule such as work time and productivity at every stage, bottle neck stage, etc. The optimized schedule was found by comparing the simulated result for the various cases. Applies this simulation model and it will be able to confirm the exception and the alternative plan of flow. Also when change investigation of equipment plan and understanding the whole situation of line in the management of a corporation, it will be able to apply a simulation. This paper also showed the web-based user interface to use the developed simulator easily.

제 1 장 서 론

1.1 연구 배경 및 필요성

생산비용 절감, 개발/생산기간 단축, 품질 향상, 신속한 소비자 요구대응, 다품종 다량 생산 등은 제조업의 불변의 과제들로 손꼽히고 있다. 이러한 문제들을 해결하기 위해서 제조업체에서는 여러 가지 시도를 하고 있고, 여기에 발맞추어 최근 새로운 패러다임으로 대두되고 있는 개념이 PLM(Product Life-Cycle Management)이다. PLM은 제품의 개발, 기획부터 시제품, 생산, 소멸까지의 제품 생명주기와 그것을 조율하고 조합하는 업무 프로세스를 포함하는 제품 개발 관리를 의미한다(신중계 등, 2001).

PLM의 대표적인 예가 디지털생산 (Digital Manufacturing)이다. 디지털 생산(신중계 등, 2001)은 제조업의 전 과정에 포함된 설계·생산·엔지니어링 정보와 노하우 등 생산시스템의 물리적 논리적 구성요소와 거동을 엄밀하게 모델링 하여 통합된 디지털 모델을 구성하고 가상현실(Virtual Reality) 기술과 네트워크 기술 등을 활용하여 생산의 전 과정에 걸쳐 관련된 여러 부문에서의 의사결정과 제어를 수행하는 기술을 말한다. 디지털생산 기술을 사용하여 실제 작업 전에 생산 공정을 미리 시뮬레이션 함으로써 실제 작업에서 발생할 수 있는 현상을 미리 분석하고 예측할 수 있는 특징을 지니고 있다. 따라서 디지털생산을 적용함으로써 실제 생산에 적용하는 단계에서 발생할 수 있는 설계 오류나 제작상의 문제점을 미리 파악하여 비용과 시간의 낭비를 최소화 할 수 있다. 또한 새로운 제조 방법 및 생산 설비의 변화를 가상생산 환경에서 미리 적용해 볼 수 있으므로 실제 제조 시스템 적용에 필연적으로 발생하는 재계획 및 수정에 따른 비용과 시간을 절감할 수 있는 기술이다.

디지털 생산을 조선 산업에 적용한 것이 디지털 선박생산(Digital Shipbuilding)이다. 디지털 선박생산기술(신중계 등, 2001)은 대표적인 주문형 소량 제조업인 조선 산업기술과 가상생산 기술 결합으로 탄생되는 신개념의 선박 생산 시스템으로 선박의 개념 설계에서부터 운용 유지까지 전 과정을 컴퓨터 속에서 만들고 시뮬레이션 하는 과정을 일컫는다.

여러 설비와 공정이 상호적으로 작용하는 시스템에서 기존의 공정이나 설비를 변경하고자 하는 경우 예상되는 결과를 확인해 보아야 한다. 여기에는 가능성을 가진 여러 방법이 있으며 그 중 3차원 공간에서의 모의 시험하는 방법인 디지털 생산 시뮬레이션이 대표적인 예가 될 수 있다. 하나의 제품이 완성되기까지 많은 공정을 필요로 하는 경우, 시·공간상으로 광범위한 경우에는 공정 사이의 상관관계를 파악하기가 힘들며, 전 단계 공정에서의 변화가 후공정에 미칠 영향을 예측하기 힘들다.

조선의 경우는 특징상 철저한 주문형 일품(一品)수주 산업이고, 공정의 다양성, 생산

의 복잡성 등의 문제로 다른 산업에 비해 자동화가 힘들고 IT 기술을 많이 사용하지만 상황에 따라 알맞게 시스템을 구축·변경 시켜야 한다. 현재 자동차나 전자제품 생산 라인의 경우는 Line의 물류 모델이 비교적 잘 구축되어 있고, 활용되고 있는 실정이다. 그러나 조선소는 배를 건조할 때 Dock에서 배를 Block 단위로 조립하는 생산시스템이므로 Line 생산을 접목하기에 어려움이 있다. 본 연구는 S사에 있는 비교적 자동화가 잘 구축되어 있고, 소조공장에서 중요한 역할을 차지하는 Panel Line의 물량 흐름을 분석하고 시뮬레이션 하여 활용하는 단계까지 범위를 정하였다. 작업계획 엔지니어가 자신이 계획한 스케줄에 대하여 미리 시뮬레이션 해 봄으로써 그 결과를 예측할 수 있고, 문제점 도출 시 대안을 찾는 데 유용하게 쓰일 수 있는 시스템 구축도 연구에 포함되어 있다.

1.2 연구 내용

조선소의 Panel Line의 공정을 분석하고 물량과 라인의 제약 조건을 파악하여 실제 생산 라인을 대신할 수 있는 Virtual Panel Line을 구현하였으며 물량의 전처리, 결과물 정리의 역할을 하는 시뮬레이터를 구축하였다. 제 1 장 서론에 이어 제 2 장에서는 Panel Line의 공정 분석과 시뮬레이션을 하기 위해 필요한 변수와 문제를 정의하였고, 제 3 장에서는 이산사건 시뮬레이션 툴과 프로그래밍으로 각각 구현한 시뮬레이터의 개발과 내용을 다루고 있다. 제 4 장에서는 개발된 시뮬레이터로 Panel Line의 생산성 향상에 초점을 맞춘 여러 조건의 변화와 그 시뮬레이션 결과를 정리하였다.

제 2 장 조선 Panel Line 공정 분석과 문제 정의

Panel Line은 판(Plate)과 판, 그리고 판과 보강재들을 용접하여 Panel을 생성하는 라인으로 이 Panel은 선박을 조립할 때 쓰이는 평블록의 기초가 된다.

조선소 Panel Line의 스케줄은 회사 내부의 여러 정보 시스템을 거쳐 물량에 대한 정보를 수집하고 우선순위(선착순규칙)가 높은 순서를 기준으로 투입순서가 정해진다. 이 작업은 생산 현장의 직장에 의해 결정되고, 오랜 경험을 바탕으로 대략의 작업시간을 산정한 후 처리 물량의 수를 결정한다. 평균적으로 여러 개의 물량을 처리하고, 하루 이상의 물량들에 대해서는 잔업이나 야간작업을 통해 소화한다.

Panel Line은 Fig.1에서 볼 수 있듯이 십여 개의 공정으로 구성되어 있다. Panel Line을 운영하는 관리자는 주어진 자원을 최대한 활용하여 최대의 생산량을 유지하고자 노력하고 있다. 특히, 작업 계획 엔지니어는 라인의 효율을 최대화 하기 위해서 각 스테이션에서 병목 공정이 생기지 않도록 블록의 투입 순서를 결정해야 하고, 물량이 분배되는 절단 공정에서는 후행공정의 잔량에 따라 부하평준화를 미리 고려해야 하기 때문에 작업계획을 하는데 고심을 많이 하고 있다. 또한 결정된 작업계획은 가공의 작업계획과도 밀접한관계가 있으므로 신중을 기해야 한다. 지금까지는 제약조건으로 블록을 구성하는 요소인 용접장, 보강재의 개수로 공수를 산출했지만 이는 정확한 공수를 산출하지 못하므로 보다 정확한 작업계획을 위해 현장에서 고려되어야 할 요소를 추가해야 하고, 보다 정확한 일정계획을 하기 위한 개선된 시스템을 요구하고 있다.

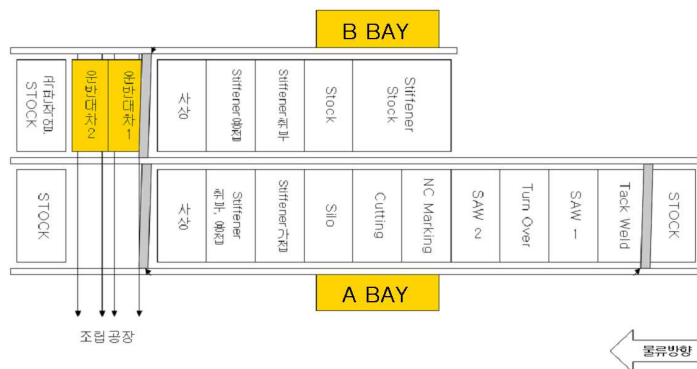


Fig.1 Panel Line Layout

올바른 시뮬레이션 결과를 얻기 위해서는 Panel Line의 공정에 대한 이해와 제약조건을 파악한 후 각 작업장의 세부 공정을 살펴보고 반복적으로 반드시 일어나는 요소를 확인하는 작업이 선행되어야 한다. 이를 위해 장기간 생산 라인에 머무르면서

Stage별 특성을 확인, 각 작업별 시간체크 등의 작업을 수행하였다. 이를 토대로 공정 분석을 'IDEF \emptyset 방법론'으로 완성하였고, 각 Activity들의 작업시간을 분석하여 각 물량의 특정 인자로 이루어진 산출식을 추출하였다.

2.1 Panel Line의 설비 및 공정 소개

(1) Tack Welding Stage

성형 가공된 Plate를 임시로 가접하여 Panel을 만드는 첫 번째 공정이다. Magnetic Crane으로 Plate를 붙여 이동하는 형태이고 작업자는 옮겨온 Plate들을 고정 및 정렬하는 작업을 실시하게 된다. 작업자가 Tap Piece(용접시 끝까지 이어주는 역할, 아크솔립을 방지하는 역할 등)를 FCAW(Flux Cored Arc Welding) 용접으로 부착을 하는 동안 다른 작업자들은 적당한 간격을 두고 Seam에 가접을 실시하게 된다. Area에 여유가 있어도 한 Panel만 처리하고, 완성된 Panel은 후행공정인 SAW1 작업장의 Area 여유분이 있을 때 투입, 그렇지 않을 때는 대기하게 된다.



Fig.2 Tack Welding Stage Scene & Plates Arrangement with Oil Pusher

(2) SAW Stage

가접된 Panel을 완전히 용접하는 공정이다. 전면 용접 후 주판을 Turn Over 하고 후면을 용접하게 된다. 전면 후면 모두 용접머신이 3대가 있고, 겐트리가 있어 각각 독립적으로 움직이게 된다. Submerged Arc Welding 작업자는 용접기 및 토치, Flux를 조절하게 되고 용접시는 슬래그제거, 용접후는 기계 셋팅, 와이어 절단 등의 일을 수행하게 된다. SAW1의 작업영역은 42.5m로 SAW2 작업장(21.7m)에 비해 상당히 넓다. 그러므로 Area 허용범위 안에서 많은 물량을 소화할 수 있다. Panel 용접을 마친 후

후행공정이 비어 있으면, 용접장의 결과물이 후행공정으로 이동하게 된다. 작업시간에 있어서 Plate의 두께에 따라 용접시간이 다르다. 두께가 두꺼워지면 그와 반비례해서 Machine Speed는 늦어지게 된다.



Fig.3 SAW Gantry

(3) Turn Over Stage

SAW1 작업장에서 Panel의 전면 용접을 마친 Block을 뒤집는 공정이다. 반드시 한 Block만 처리하게 된다. SAW1 작업장에서 계속 물량을 넘겨주더라도 SAW2 작업장에서 담아둘 공간이 없으면, 물량을 받고, 대기 상태로 있게 된다.



Fig.4 Turn Over Scene

(4) NC Marking Stage

주판의 용접을 모두 마친 후 마진면 절단이나 Stiffener가 놓일 위치를 Marking 하는 공정이다. Area의 여유가 있어 여러 Block을 받게 되더라도 반드시 작업은 한 Block만 처리하게 된다. 작업자는 Dimension 체크, Controller 조작, NC Data의 입력 등의 일을 수행하게 된다.



Fig.5 NC Marking Scene

(5) Cutting Stage

주판의 Tap piece를 제거하는 공정이다. Tap piece를 제거한 후 작업자는 Grinding, 수정용접을 실시하게 된다. 실제 용접(SAW)이 잘 되었는지 확인하기 위해 작업자는 줄자를 들고 Dimension(가로, 세로, 대각선)을 확인하는 작업을 실시한다. Cutting 공정을 마친 Block은 분기되어 35, 36Bay로 각각 들어가게 된다. 36Bay의 경우는 Silo가 있어 Buffer 역할을 하지만, 35Bay는 Fitup Stage가 비어 있지 않으면 Cutting Stage가 Buffer 역할을 하게 된다.



Fig.6 Margin Space Cutting

(6) FitUp Stage

주판에 Stiffener(Angle, T-bar, Flat-bar)를 세우고 가접하는 공정이다. B Bay 작업장은 Fitup Machine(자력(磁力)이 있어 Stiffener를 붙여 이동함)이 Stiffener를 부착하고, 이동한 후 위치를 세팅하면 작업자가 취부를 하게 된다. B Bay 작업장은 A Bay와 달리 취부 횟수가 많으며 수직으로 세워졌는지, 취부상태는 올바른지 점검하는 시간이 많다. A Bay 작업장은 Stiffener를 배재하는 공정이다. 후공정에 Complex Machine이 취부를 하면서 동시에 용접을 하기 때문에 Stiffener를 세우고 넘어지지 않게 약간의

취부만 하게 된다. Stiffener는 팔렛트에 놓여져 있고, 이를 크레인으로 이송한 후 작업자가 마킹된 선에 맞추어 세우게 된다. 굵은 플랫폼이 놓일 경우 작업시간이 길어진다.

(7) Stiffener Welding Stage

전 공정에서 Fitup 된 Stiffener를 주판에 용접하는 공정이다. 20Pole Machine은 헤드가 5개 달려있고 각 헤드에는 4개의 Tandem torch가 달려 있다. 주판이 이송되면 작업자는 주판의 상태(놓인 위치, Stiffener가 제대로 취부되었는지 여부)를 확인하고 헤드를 정렬, 겐트리 이동, 토치정렬의 과정을 마친 후 Arc Start를 한다. 용접 후 겐트리를 후퇴하고 작업자가 와이어 절단, 점검, brushing 작업 등을 수행한다.



Fig.7 Welding Head Arrangement of Complex Machine

(8) Modification & Finish Stage

Stiffener 용접을 마친 주판을 수정, 마무리 하는 공정이다. 작업자가 많이 배치된 Stage로서 용접 비드형상을 확인한 후 Grinding, 슬래그 제거 작업을 실시하게 된다. 경우에 따라서는 비드에 Primer 칠을 하기도 한다. 중조에서 사용 할 러그를 부착하는 경우 일정하게 15분가량 소요된다.

2.2 공정 시뮬레이션 제약 조건

라인에 있는 물량의 흐름을 시뮬레이션 하기 위해서는 우선 생산라인에 존재하는 각각의 제약요소를 파악하고 이를 시뮬레이션 모델에 반영시켜야 한다. 현실과 가장 유사한 결과를 얻기 위해서는 반드시 거쳐야 할 과정이고 모델의 정도(程度)를 결정하는

중요한 요소이기도 하다. 본 Panel 시뮬레이션 모델에서는 시뮬레이션 입력 데이터의 제약과 현 라인이 안고 있는 제약이라는 두 가지 측면에 대해서 다음과 같이 정리 하였다.

(1) 시뮬레이션 입력 데이터 특성

- 각 Block은 여러 개의 이음매(Seam), 종보강재(Longi)를 가진다.
- 각 스테이지의 작업은 Seam과 Longi의 개수에 따라 다르다.
- Longi의 종류에 따라 용접 회전수가 다르다.
- 선종, Block 종류(LNG선, 컨테이너선)에 따른 용접속도가 다르다.
- 같은 주판이라도 Plate들의 두께는 다를 수 있다.
- Tack Welding 등에서 중요한 Cross Seam에 따른 작업시간이 고려되어야 한다.

(2) 현 라인의 물리적, 운용적 특성

- 각 스테이지마다 공간의 크기가 다르다.
- 한 스테이지에 여러 개의 Block이 존재한다.
- 후행공정이 비워 있으면 선행공정의 결과물이 투입된다.
- 선행공정이 가공 중일 때 후행공정이 비워 있으면 완료된 Block만 후행공정으로 이동된다.
- 투입인원과 생산성 향상은 정비례하지 않는다.
- 큰 작업장에서 작은 작업장으로 이동할 때 병목이 발생한다.
ex) SAW2, NC Marcking, Complex Machine

올바른 시뮬레이션 결과를 유도하고, 실제 Line의 현상을 반영하기 위하여 투입 물량의 기본 Dimension 정보와 그에 따른 회전수, 용접속도(Table 2) 또한 반영되어야 한다. 다음은 물량의 주요 Dimension 정보와 그 의미하는 바를 정리하였다.

- Length : 용접선의 길이
- Breadth : 작업장의 Area를 고려한 투입량 결정
- Thickness : SAW, Longi Welding Stage의 Welding Speed에 영향
- Seam, Longi : Machine의 회전수 결정

Job Stage	Area(Breadth) (m)	여러 Block 존재 가능 여부	여러 Block 동시 작업 여부	기본 배원 (名)
T/W	23.0	×	×	5
SAW1	42.5	○	○	3
SAW2	21.7	○	○	3
NC-Marking	21.7	○	×	2
Cutting	21.6	○	×	2
Fitting	21.6	○	×	3
Welding	21.5	○	×	2
Finish	21.5	○	○	5

Table 1 Characteristic of Job Stage

두께(mm)	전면용접(SAW1)		후면용접(SAW2)	
	Speed(cm/s)	Speed(cpm)	Speed(cm/s)	Speed(cpm)
~ 11	0.444	135	0.462	130
12 ~ 13	0.462	130	0.462	130
14 ~ 16	0.48	125	0.48	125
17 ~ 19	0.545	110	0.545	110
20 ~ 21	0.545	110	0.571	105
22 ~ 23	0.571	105	0.6	100
23 ~ 24	0.612	98	0.632	95
25 ~ 26	0.8	75	0.594	101
27 ~ 28	0.8	75	0.619	97
29 ~ 31	0.857	70	0.6	100
32 ~ 36	0.8	75	1	60
37 ~ 40	1.333	45	1	60

Table 2 Table of Real Welding Speed used from Empirical Data

2.3 공정 IDEF0 모델링

IDEF0 기능 모델링 방법은 조직이나 시스템의 의사결정, 행동, 활동을 모델링 할 수 있도록 디자인 된 방법이다. IDEF0는 SADT로(Structured Analysis and Design Technique) 잘 알려진 그래픽 언어에서 파생되었으며 1976년에 개발이 시작되어 1981년 6월 ICAM(Integrated Computer Aided Manufacturing) Function Modeling Manual로 미 공군에 의해 발표되었고 1986년 미 국방부에 의해 Activity 모델링을 위한 표준 방법(Department of Defense 8020.1-M)으로 채택되었으며 1993년 12월 NIST(National Institute of Standards and Technology)는 미연방 정보처리 표준(Federal Information Processing Standard Publication 183-Integrated Definition for Function Modeling)으로 IDEF0를 채택하였다. IDEF0는 셀(Cell) 모델링 그래픽 표현방법을 바탕으로 시스템을 기능적 관점에서 분석, 커뮤니케이션할 수 있는 기능/활동 모델링 방법이다.

Panel Line에서는 Line 전체적인 분석과 아울러 각 작업장의 세부 기능을 묘사하기 위하여 IDEF0 방법론을 사용하였다. 모델링 결과는 현장에서 오랜 경험을 쌓은 기사의 확인과 검증을 거쳐 정도를 향상 시켰다.

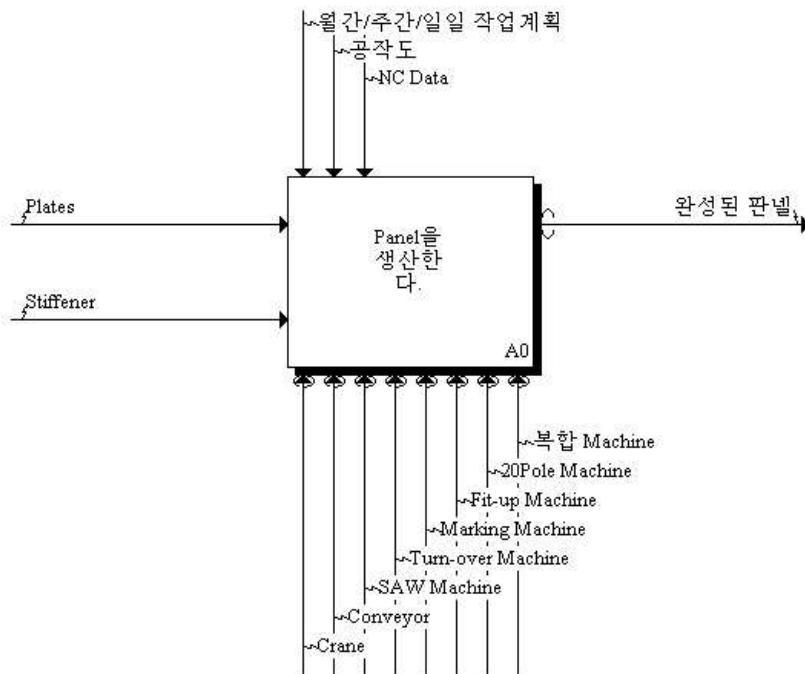


Fig. 8 Function Modeling of Whole Stage using IDEF0

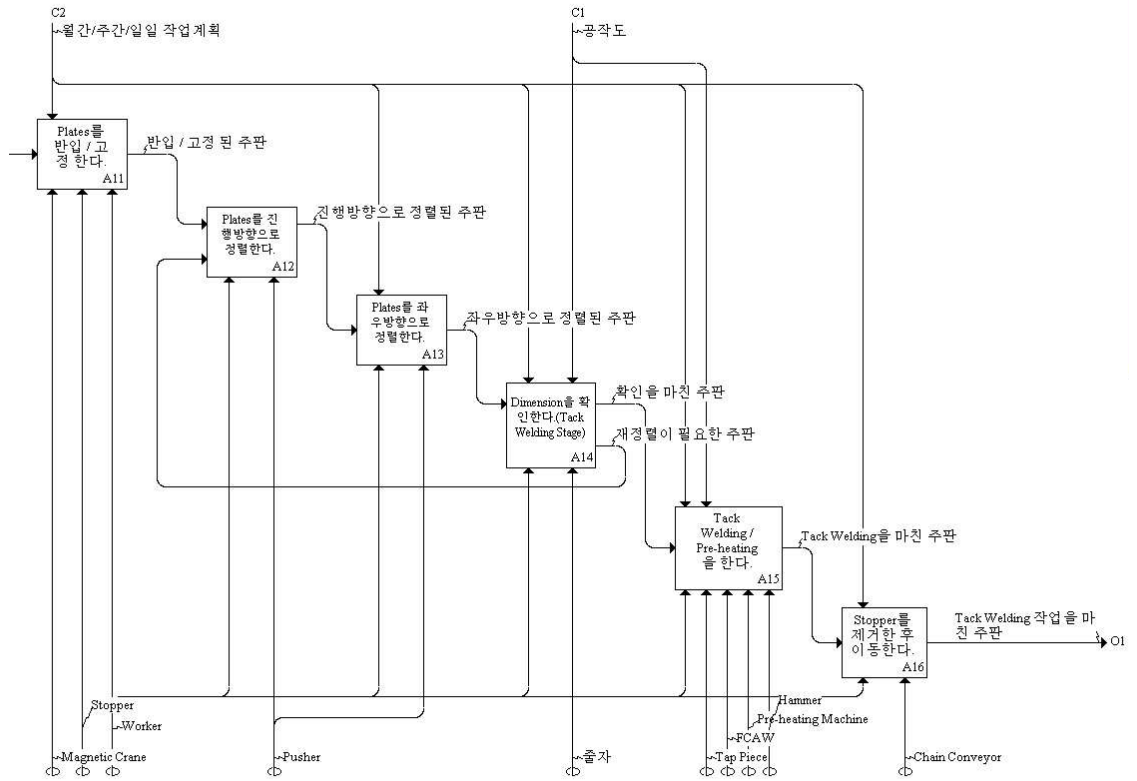


Fig. 9 Function Modeling of Tack Welding Stage using IDEF0

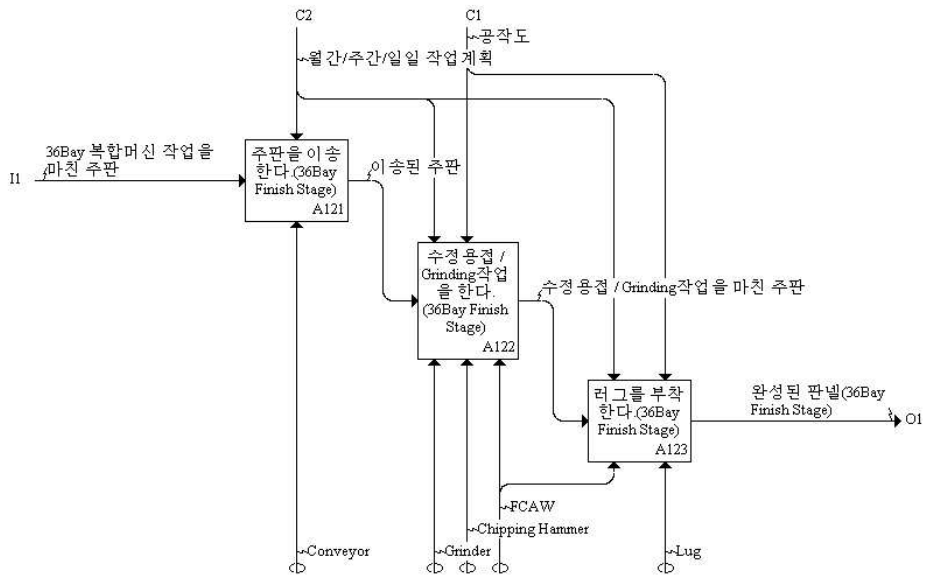


Fig. 10 Function Modeling of Modification & Finish Stage using IDEF0

2.4 공정 소요시간 정식화 및 세분화

분석된 IDEF0 모델에서 각각의 액티비티(Activity)의 내용을 분석한 결과 액티비티와 물량 투입 정보의 요소(Factor)와 연관이 있음을 확인하고, 작업시간을 나타내는 산출식을 완성하였다. 실제로 현장 기사나 직·반장은 나름대로의 경험식을 가지고 대략의 작업시간을 추정하지만 이는 실제 현상을 반영하기엔 요소나 제약조건들이 많이 부족하므로 정도가 떨어지는 편이다. 경험식 이외에도 조업도라고 불리는 작업시간을 계산하는 룰이 있다. 실제 공정에서 사용되는 여러 변수들을 무시하고 Seam과 Longi로 대략의 용접길이만을 판단하고, 이를 Panel의 생산 개수로 환산한 전체 용접길이를 조업도를 평가하고 있다. 컨테이너선과 석유운반선의 블록을 비교하면 같은 용접길이(현재의 기준에서 사용하는 같은 조업도)라도 석유운반선은 용접이 대부분 1회로서 끝나고 공정상 복잡한 부분도 없다. 반면 컨테이너선의 경우는 60mm 이상의 후판이 단차로 들어가게 되고 이는 전면용접 공정에서 부하로 작용하게 된다. 왜냐하면 후판의 단차조정, 용접시 유압 푸셔(Pusher)로 조정하기 때문이다. 또한 후판에는 스티프너 용접시 플랫폼이 붙게 되는데 이 때 각장을 고려하여 여러 번(2~3회) 용접하게 된다. 후판 용접시 대부분 용접속도가 늦어지는 점도 고려해야 할 사항이다. 같은 전체 용접 길이라도 컨테이너선의 블록은 조업도를 높게 고려해야 할 것이다. 현재 이렇게 부정확한 경험식과 조업도를 보완 하고자 새로운 산출식을 유도해 냈고, 이는 현재 라인의 제약조건과 연계되어 정도가 높은 시뮬레이션 결과를 도출하도록 한다. 산출식은 액티비티를 중심으로 계산하였고, 이 역시 현장 기사의 도움으로 수차례 보완하여 산출식의 정도를 향상시켰다.

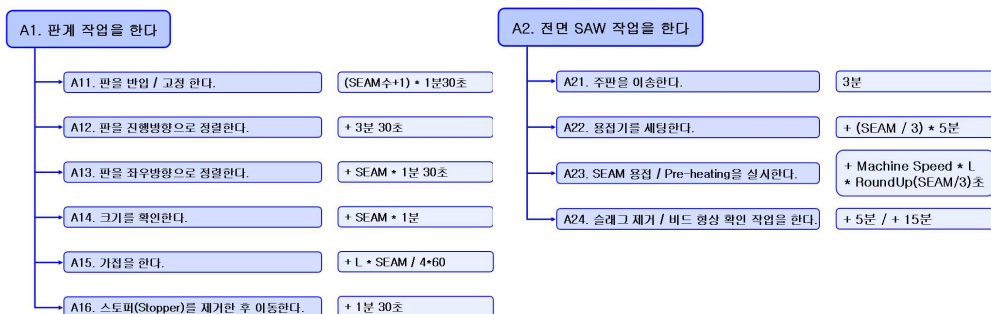


Fig. 11 Empirical Formula by Time Analysis

제 3 장 시물레이션 모델링

일반적으로 많은 물량이 많은 공정을 거쳐 갈 때 모든 물량이 동일한 순서로 각 작업장을 거치는 것이 아니며, 또한 물량마다 거치는 작업장의 형태와 패턴이 다를 때는 최적의 분석적 기법이 존재하지 않으며, 또한 모든 가능한 작업순서를 평가하여 최적 작업순서를 구한다는 것도 현실적으로 거의 불가능하다. 이러한 복잡한 상황에서는 전체 작업일정을 한꺼번에 결정하는 것이 아니라 각 작업장마다 현재 대기 중인 작업 중 어느 작업을 우선적으로 처리해야 할 것인가를 우선순위규칙을 이용하여 동태적으로 결정해 나간다. 따라서 이 경우에는 여러 우선규칙을 상대적으로 평가하기 위해 시물레이션을 사용한다(‘현대 생산·운영관리’, 이상범, 2003).

시물레이션을 사용하는 몇 가지 주요 이유는 다음과 같다.

1. 해가 수리적 분석에 의하여 쉽게 혹은 전혀 구해지지 않을 때
2. 설비계획안을 경영진에게 설명할 때
3. 제안된 시스템이 어떻게 작동할 것인지를 운영담당자에게 설명할 때
4. 제안된 시스템의 작행 가능성을 시험할 때
5. 작업능력과 저장 공간 소요량을 결정할 때
6. 수리모형의 유효성을 확인할 때
7. 실제 시스템, 환경, 혹은 운영절차에서의 변화에 대한 영향을 예측할 때

Panel Line을 흐르는 물량과 Line에 존재하는 여러 제약요소들을 고려해서 시간의 흐름에 따른 물량의 처리 및 작업시간 등의 정확한 결과를 얻기 위해 Line 시물레이션을 수행 하였다. 이번 장에서는 시물레이션 툴을 이용한 연구와 범용 개발 언어로 구현한 시물레이터를 소개한다.

3.1 공정 시물레이션에 대하여

시물레이션은 실제 상황에 대한 모의실험으로서, 컴퓨터 프로그래밍 언어를 사용해 표현한 모델을 시간의 경과에 따라 변화시켜서 시스템의 동적 거동과 구조, 운용절차, 의사 결정 규칙 등을 효과적으로 결정함으로써 적절한 의사 결정과 정책 수립에 도움을 준다. 이러한 목적의 물리적 모델은 시물레이터(Simulator)인데, 예를 들면, 조종사를 교육시키기 위한 비행 시물레이터가 있다. 최근 '가상(또는 인위적)현실(Virtual Reality)'이 신속하게 현실을 모의 실험하는데 중요한 역할을 하고 있다. 예로써 실제

생산 설비를 활용하지 않고 컴퓨터 디스플레이 상에서 가상 생산 작업을 나타내는 가상공장(Virtual Factory)이 있다.

가상생산에서 얻을 수 있는 가장 강력한 장점은 What-if 시뮬레이션이다. 실제로 테스트 하는 경우 비용이 많이 드는 작업들을 각 시나리오에 따라 컴퓨터상에서 디지털화 하여 시뮬레이션 해 봄으로써, 결과를 미리 알 수 있으며 최적의 프로세스를 선택할 수 있는 것이다. 전 세계적으로 자동차 및 항공 업계에서는 디지털 생산을 선택이 아닌 필수적인 작업으로 여겨서 채택하고 있으며, 향후 디지털 생산은 제조업 전반에 걸쳐서 필수적이며 강력한 솔루션으로 자리매김 할 전망이다.

일반적인 시뮬레이션에서 시스템은 시간의 경과에 따라 상태가 변화해가게 되며, 따라서 시간의 관리는 중요하게 된다. 시뮬레이션 기법을 구분할 때에도 이 시간의 관리를 어떻게 하느냐에 따라서 구분하는 것은 매우 의미 있는 일이다. 시간에 의해 연속 시스템과 이산사건 시스템으로 구분 되는데 이산사건 시스템은 동적(dynamic) 시스템이다. 동적 시스템의 어떤 시간에서의 상태(state)는 그 시간에서 시스템이 갖는 상태변수의 값으로 결정되며, 시스템의 상태 변수는 시간에 따라 변화되어 간다. 이산사건 시스템은 임의의 시간에 순간적으로 상태변이를 일으키며 한 상태에서 머무는 시간이 불규칙적인 시스템으로 특징 지을 수 있다. 따라서 운영체제 이상 레벨의 컴퓨터 시스템, 컴퓨터 네트워크, 생산시스템, 교통 시스템 등은 모두 이산 사건 시뮬레이션(Discrete Event Simulation)으로 모델링 될 수 있다. 여기에서 Event란 시뮬레이션 모델에 상태변화를 가져오는 사건을 말하며, 이산사건 시뮬레이션은 모델의 상태변화가 일어나는 경우에만 시간을 진행시키는 방법이다. 본 논문에서는 Panel Line의 여러 공정과 각 작업장별로 발생하는 여러 Event와 시간의 흐름에 따른 결과를 살펴보기 위하여 이산 사건 시뮬레이션을 공정 시뮬레이션의 주요 기법으로 활용하였다.

3.2 Panel Line 공정 시뮬레이션 시나리오

본 모델에서 수행한 Panel Line 공정 시뮬레이션은 다음과 같은 절차로 수행하였다. Line에 들어오는 블록은 여러 가지 속성을 가지고 있다. 예를 들면 Bay 정보, Block Name, Seam, Longi, Length, Breadth, Thickness(1~7EA)가 있다. 제일 먼저 T/W 작업장에 오기 전에 작업장의 용적이 얼마나 되는지 체크한다. 이 용적은 판의 Breadth를 합한 것으로 용적을 넘지 않게 한다. 용적의 여부에 따라 추가 투입이 결정된다. SAW1 경우 작업장에 들어올 수 있는 최대의 판의 수는 5개이다. T/W 작업장에서 산술식에서 계산된 시간만큼 작업을 하고 SAW1 공정으로 들어가게 된다. SAW1 작업장은 42.5m의 용적을 가지고 있으므로 여러 개의 판을 적치할 수 있다. 이 작업장에서는 3개의

Submerged Arc Welding Machine이 있고 이 Machine의 작업 속도는 Plate의 두께에 따라 용접속도가 다르게 주어진다. 두 Plate를 비교하여 두께가 얇은 판의 용접속도를 작업장에서 반장들이 사용하는 테이블 값(Table 2)을 이용하여 모델에 적용한다. 그러므로 판 두께에 따라 각 Machine들의 용접속도는 전부 다를 수 있고, 주판이 이동될 때에도 이를 고려하여야 한다. 작업이 끝난 판은 후행공정의 작업여부에 따라 투입이 결정된다. 후행공정은 T/O로 작업소요시간은 한 판당 10분가량 소요되며, 한 번에 한 블록밖에 작업하지 못한다. T/O된 주판은 SAW2 작업장에 들어가게 된다. SAW2 공정은 SAW1과 같은 Machine의 수를 가지고 작업시간도 같이 반영하나 용적이 작아 SAW2 작업장이 전체 Line에서 BottleNeck¹⁾이 되는 주요 원인이 된다. 다음 공정은 NC-Marking 작업장으로 이면 용접이 마친 주판을 작업자가 Dimension을 확인하고, Controller를 조작 후 NC Data를 받아 자동으로 마킹을 실시한다. 마킹을 마친 주판은 Cutting 작업장으로 이동하는데 작업자가 T/W Stage에서 부착한 Tap Piece를 제거하고 끝단을 손질한다. Tap Piece의 수는 Seam수와 직접적인 관련이 있으므로 작업시간은 Seam수만을 고려하여 계산한다. 절단공정이 끝난 Block은 각각의 물량정보에 의해 A Bay와 B Bay로 이동하게 된다. B Bay를 먼저 살펴보면 절단공정을 마친 후 Transfer에 의해 B Bay로 이동하면 미리 대기하고 있는 Stiffener Pallet을 탑재한 대차와 Fitup Machine에 의해 작업을 하게 된다. B Bay 공정은 Fitup 이후 주판을 정렬치 않고 바로 Stiffener 용접을 실시하므로 Fitup 공정에서 주판을 정렬하는데 시간을 많이 소비한다. Longi 수에 따라 작업시간이 달라지므로 Longi 수만이 작업시간에 영향을 주는 유일한 변수이다. 취부를 마친 주판은 20Pole 작업장에서 용접을 하게 되는데, 이 용접장은 Machine의 Speed 뿐만이 아니라 주판의 타입, 즉 두꺼운 플랫바가 붙는 경우는 회전수를 늘리게 되므로 이 또한 고려해야 할 사항이다. 이는 Product의 정보(선종, 탑재위치, 최대 Plate 두께)를 받아 결정하게 된다. Stiffener 용접을 마친 Product는 Finish Stage에서 수정용접 및 러그를 부착하게 된다. 수정용접을 할 때에는 Longi수와 그 Longi의 길이 즉 Length값이 고려해야 할 사항이다. A Bay에서는 취부가 아닌 배재공정을 하게 되는데 작업자가 크레인으로 Stiffener를 가져와서 3~4군데 취부를 한다. Complex Machine 작업장에서는 주판을 측단검출기로 먼저 정렬하는 과정이 포함된다. 그리고 취부와 동시에 용접을 실시하게 되는데 20pole 공정과 마찬가지로 Machine Speed와 회전수, Length 값이 작업시간의 주요 변수들이다. 물량에 따라서는 취부를 마치고 바로 Line을 벗어나는 Product가 존재할 수 있고 외판 Marking에 따른 T/W 작업장에서의 비드작업 과다지연 등이 일어날 수 있으며, 배원 계획에 따라 전체 Line이나 작업장별로 작업시간이 달라질 수 있다.

1) **BottleNeck** : 정체가 일어나는 곳. 병목 작업장을 뜻함.

3.3 QUEST를 활용한 공정 시물레이션

시물레이션 연구수행을 위한 주 도구인 소프트웨어는 크게 시물레이터(Simulator), 시물레이션 언어(Simulation Language), 범용 언어(General Language)와 같이 분류할 수가 있다. 시물레이터는 특정분야에 특화된 문체의 설계 및 운용을 지원하기 위한 전용의 시물레이션 패키지를 나타내는 전문용어이다. 여기에는 항만 시물레이터, 항공기 조정 시물레이터, 열차시물레이터, 공항 시물레이터 등 분야별/목적별로 수많은 시물레이터들이 개발되어 사용되고 있다. 시물레이션 언어는 모델링 및 실험수행, 분석, 보고서 작성 등과 같이 시물레이션 연구를 위한 서브루틴(기능)들을 미리 만들어둔 시물레이션 전용 언어이다. GPSS, SIMAN(ARENA), Simscript, SLAM 등이 지금까지 많이 사용되어 왔으며, 제조시스템의 AutoMod, MAP/1, SimFactory, PROMODEL, Taylor II 와 같이 응용분야에 특화된 시물레이션 언어도 있다. 범용 언어는 C, Visual Basic, Pascal, Fortran 등과 같이 우리가 흔히 알고 있는 부류의 개발언어들을 지칭한다.

본 연구에서는 Delmia사의 QUEST를 활용하여 Panel Line의 공정 시물레이션을 수행하였다. QUEST는 공장 설비의 레이아웃과 공정 흐름을 효율적으로 모델링하고 시물레이션 하여 시스템을 분석, 검증하는 유연한 객체지향 이산 사건 시물레이션 솔루션(Flexible Object-Based Discrete Event Simulation Tool)이다. QUEST는 대화식의 3차원 그래픽 유저 인터페이스를 제공하고, 별도의 컴파일 없이 실시간으로 시물레이션을 수행함으로써 좀 더 정확하고 신속한 결과값을 얻을 수 있는 장점이 있다. 이외에도 객체기반 모델 구축과 효과적인 분석을 제공하는 등의 이점이 있어 시물레이션에 활용하였다.

ID	date	hull_no	block	bay	seam	long	L	B	t1	t2	t3	t4	t5	t6	t7	vnc
1	20040213D	1489	SS16LS	5	2	7	19.4	10.5	38	67	0	0	0	0	0	0
2	20040213D	1489	SS16LP	6	2	10	19.4	10.5	15	12.5	12	0	0	0	0	0
3	20040213D	1489	SS16LIS	6	2	10	19.4	10.5	15	12.5	12	0	0	0	0	0
4	20040213D	1517	SS18BLP	5	1	5	18.5	7.8	18	17	0	0	0	0	0	0
5	20040213D	1517	SS18BLS	6	1	5	18.5	7.8	18	17	0	0	0	0	0	0
6	20040213D	1489	SE44P	5	1	4	21.3	11.4	16	20	0	0	0	0	0	0
7	20040213D	1489	SE44S	6	1	4										
8	20040213D	1517	SB17DP	5	4	1					15	15.5	17	0	0	0
9	20040213D	1517	SB17DS	6	4	1					15	15.5	17	0	0	0
10	20040213D	1517	SD11TP	5	1	2					0	0	0	0	0	0
11	20040213D	1517	SD11TS	6	1	1					0	0	0	0	0	0
12	20040213N	1466	SS19P	5	5	2					17	15	17.5	0	0	0
13	20040213N	1466	SS19S	6	5	2					17	15	17.5	0	0	0
14	20040213N	1517	SD11LS	5	1	7					0	0	0	0	0	0
15	20040213N	1517	SL13C	6	3	1					13	0	0	0	0	0
16	20040213N	1517	SD10LS	5	1	6					0	0	0	0	0	0
17	20040214D	1517	SL15C	6	3	1					15	0	0	0	0	0
18	20040214D	1461	SE43P	5	2	9					0	0	0	0	0	0
19	20040214D	1461	SE43S	6	2	9					0	0	0	0	0	0
20	20040214D	1463	SS11P	5	2	1					0	0	0	0	0	0
21	20040214D	1463	SS11S	6	2	1					0	0	0	0	0	0
22	20040214D	1461	SS14LP	5	2	1					0	0	0	0	0	0
23	20040214D	1461	SS14LS	6	2	1					0	0	0	0	0	0
24	20040214D	1518	SE41LLS	5	1	0					0	0	0	0	0	0
25	20040214D	1517	SB17TP	6	2	1					0	0	0	0	0	0
26	20040214D	1517	SB17TS	5	2	1					0	0	0	0	0	0
27	20040214D	1489	SB17DP	6	5	1					15	16	14.5	0	0	0
28	20040215D	1489	SB17DS	5	5	12	19	18.3	15.5	15.5	15	15	16	14.5	0	0

Fig. 12 Input Data & DB Architecture

QUEST 모델상에서 현 Line의 Layout을 옮겨놓고 각각의 작업장에 대한 로직을 완성하였다. 작업장뿐만 아니라 Line의 Constraints 역시 로직에 반영시켜야 하고, DB 연계, 결과 추출, 흐름 제어 등도 로직에 포함되어 있어야 하며, 이는 PASCAL 기반의 SCL(Simulation Control Language)이라 불리는 언어로 구현되었다. 물류 정보값(Fig. 12)을 받아오는 DataBase 구축의 DBMS는 SQL Server를 사용하였고, 이를 QUEST와 연계시키고 이벤트를 주고받기 위한 Connection을 제공하는 DELFOI 라는 Middleware를 사용하였다.

흐름 제어를 위해 Conveyor 위에 deck라는 가상의 포인트를 생성하고, 물량이 deck 위에 놓였을 때 후공정의 상태를 체크하거나 여러 속성 값들을 변경, 후공정으로 넘겨주는 역할을 수행하게 하였다. 작업장(Machine)에 들어온 물량들은 산출식에 근거하여 작업시간을 계산하고, 그 시간만큼 작업장에 머무르게 된다. 다음은 구현된 QUEST 모델의 몇 가지 로직의 예이다.

(1) SAW1 wait deck 로직

- ① SAW1 Machine의 Area와 점유공간을 파악한다.
- ② wait_dec에 들어온 Block의 폭을 계산하여 SAW1 Machine 작업장에 추가로 들어갈 수 있는지의 여부를 판단한다.
- ③ 넘겨주게 될 때는 Machine의 점유공간과 넘겨주는 Block수를 조절하고, 위의 공정을 반복하게 된다.
- ④ Area 크기에 제약을 받아 추가로 넘겨주지 못하게 될 때는 Machine과 work_dec에 넘겨준 Block수를 전하고 작업을 실시해도 된다는 신호를 준다.
- ⑤ Machine의 작업이 끝날 때까지 대기하게 되고, 작업이 끝났다는 신호와 함께 ①번 순서로 작업을 재개하게 된다.

(2) SAW1 work dec 로직

- ① wait_dec에서 작업을 해도 된다는 신호가 올 때까지 대기를 하다가 Block을 Machine으로 보내는 역할을 한다.
- ② Machine에 보낸 Block수와 wait_dec에서 보낸 Block수를 비교하여 작업의 지속여부를 결정한다.

(3) SAW1 Machine 로직

- ① wait_dec으로 부터 작업 시작신호를 기다린다.
- ② work_dec으로 부터 wait_dec에서 넘겨준 Block수만큼 Block을 받는다.
- ③ 현재 Machine에 있는 Block들을 돌면서 두께값에 해당하는 Cycle_Time을 계산한다.
- ④ 각 thickness로부터 추출된 3개의 Cycle_Time중 가장 큰 값을 추출한다.
- ⑤ 작업시간 만큼 SAW 용접을 한다.
- ⑥ 현재 Machine에 있는 Block들을 체크하면서 Seam수와 작업한 Seam수를 비교하여 Block을 내보내 준다.
- ⑦ 작업을 마치게 되면 Machine과 각 dec의 속성 값을 0으로 초기화 한다.

(4) Stiffener Welding Stage 로직

- ① wait_dec으로부터 작업 개시 신호를 기다린다.
- ② wait_dec으로부터 넘겨받은 Block수만큼 루프를 돌면서 Block을 요청한다.
- ③ Machine에 들어온 Block의 정보(Seam, Length, Longi수)를 파악한다.
- ④ Machine의 종류에 따라 분기되어 각 해당 Cycle_Time을 추출하고 작업을 하게 된다.
- ⑤ Machine의 점유공간을 현재 Block의 Breadth를 뺀 값으로 갱신한다.
- ⑥ Block을 내 보낸다.
- ⑦ 해당 Machine과 연결된 각 dec의 속성 값을 초기화한다.

사용자는 개발된 User Interface를 통해 일정별로 해당하는 물량을 선택하고 QUEST 모델을 로딩하고 시뮬레이션을 수행한다. 내부적으로는 Middleware인 DELFOI를 통해 시뮬레이션에 필요한 물량정보를 요청하거나 결과를 DB에 저장하며, Fig. 14은 QUEST 모델을 이용하여 시뮬레이션 하는 경우 정보의 흐름을 나타내고 있다.

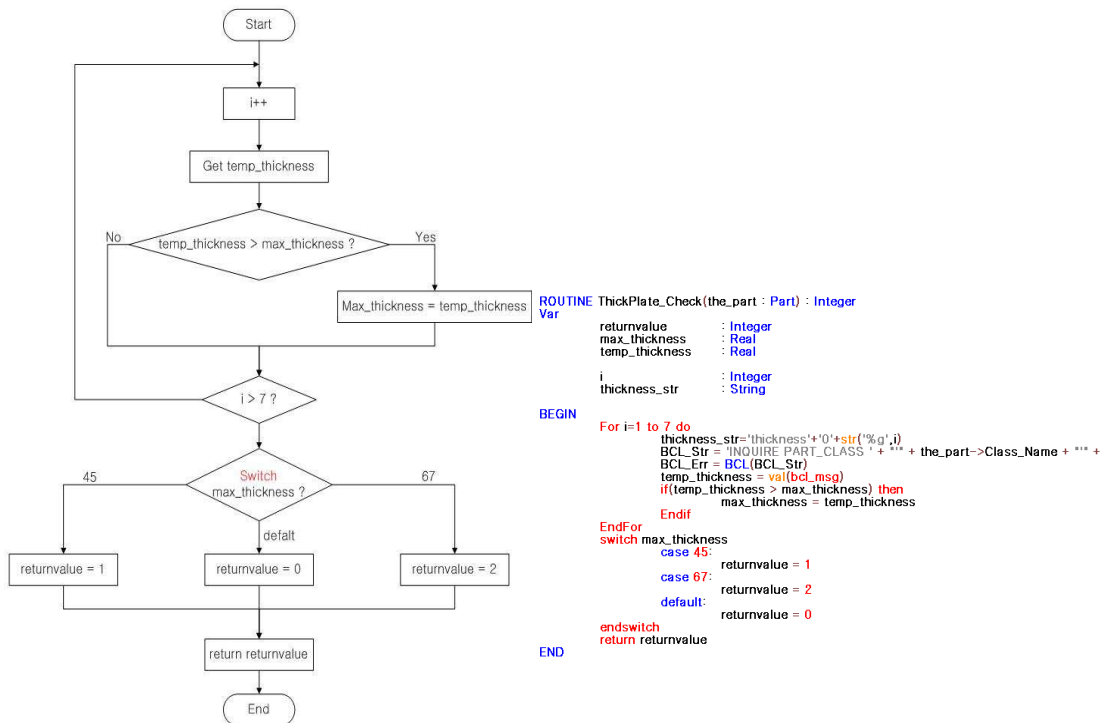


Fig. 13 Flowchart and realized SCL Code for confirms the Thickness of the Plates

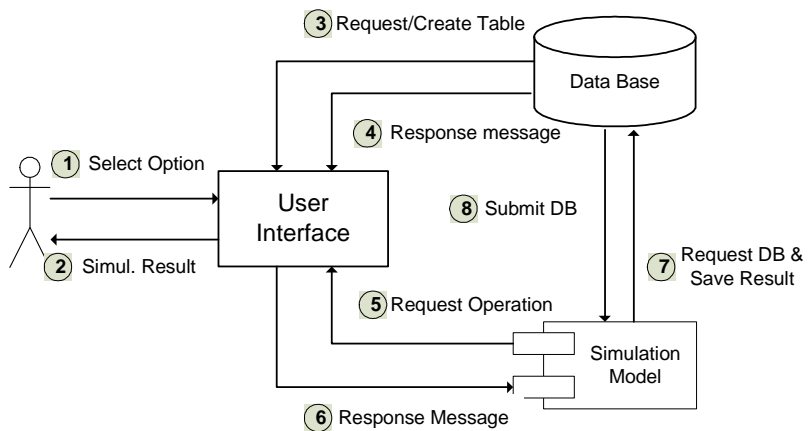


Fig. 14 Information Flow of Simulation System

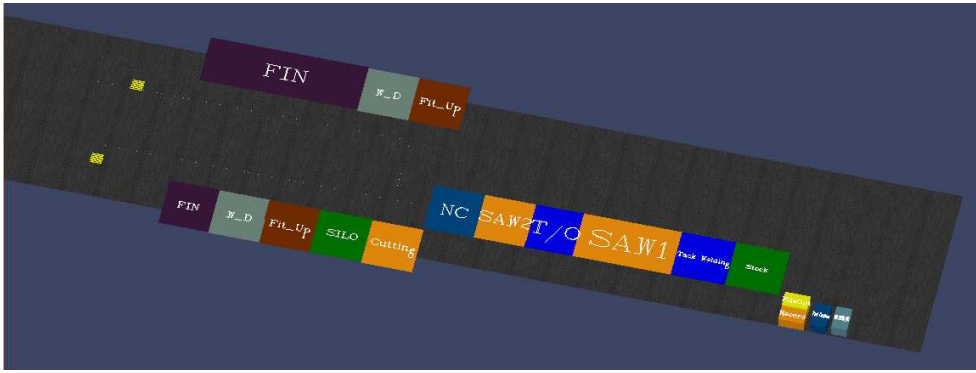


Fig. 15 Layout of Panel Shop Simulator using QUEST

3.4 범용 개발언어에 의한 공정 시물레이션

일반 작업자가 QUEST 모델을 사용하고, 컨트롤 하기란 교육을 받고 숙련되기 전까지 결코 쉬운 일이 아니다. 새로운 인터페이스에 대한 요구가 대두되었고, 그 결과 범용 개발언어로 시뮬레이터 전체를 새로이 모델링 하게 되었다. 이 시뮬레이터는 QUEST의 주요 장점인 이산 사건 시물레이션 모듈을 그대로 범용 개발 언어 (Microsoft Visual C#.Net)로 이식하였으며 QUEST가 처리하지 못하는 물량 데이터의 입력이나 결과물 정리를 유연하게 처리한다.

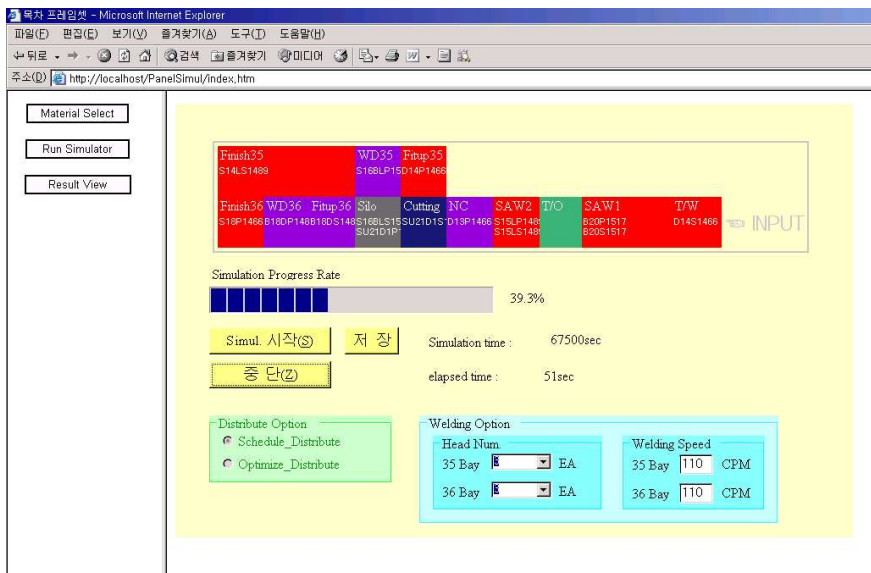


Fig. 16 Web Based Panel Shop Simulator using Development Language

작업자가 시물레이션 모델을 사용하기 위해서는 웹 페이지에 접속한 후 시물레이션 할 물량을 결정하고 서버의 시물레이터를 원격 실행시키게 된다. 시물레이션이 끝나면 결과를 몇 가지 형태(텍스트, 그래프, 테이블)로 살펴볼 수 있게 제작하였다. 이외에도 용접할 때 헤드수를 늘리거나, 용접속도에 변화를 주었을 때의 해당 작업장뿐만 아니라 전체 라인에 미치는 영향도 고려할 수 있도록 하였다. 이는 라인에 장비의 증설, 설비 개선, 공법의 변화가 생길 때 미리 그 영향을 예측할 수 있도록 하는 기능이다.

웹 기반의 Application의 가장 큰 장점으로서는 별다른 Setup 이나 Install 과정 없이 일반 웹 브라우저에서 바로 실행 가능 하다는 점이다. 개발된 웹 프로그램은 기존의 ASP가 아닌 ASP.Net 기반으로 제작되었다. ASP 프로그램은 인터프리터 방식인데 반해, ASP.Net 프로그램은 컴파일 되어 실행되기 때문에 실행속도가 빠르다는 장점이 있다. 이외에도 비연결 지향의 ADO.Net을 사용하여 리소스를 적게 차지하고, 객체지향 프로그래밍이 활용된 점이 주요 특징이다. 알고리즘을 포함하는 Behind Code는 C#을 사용하였고, Server의 IIS(Internet Information Service) 기반에서 실행되도록 되어 있다. User가 웹 브라우저 주소창에 서버가 있는 경로를 입력하면 메인 창이 활성화 되고, 시물레이션이 가능하게 설정된다.

여러 가지 세부공정을 분석하여 이해하기 쉽고, 재사용성이 있는 모델을 구축하기 위하여 객체지향 분석 기법을 사용하였다. 객체지향 분석 방법(Object Modeling Technology)은 시스템을 체계적으로 분석하고 개발함으로써 시스템의 복잡함에 대한 이해를 도와주며, 거대 시스템 개발에 있어서 시스템의 적응성을 확보할 수 있는 방법이다(Shin & Shin, 2002). 본 연구에서는 객체모델 개발을 위해 UML을 사용하여 각 작업장 분석 및 시물레이터를 모델링(Fig. 17, 18)하였다.

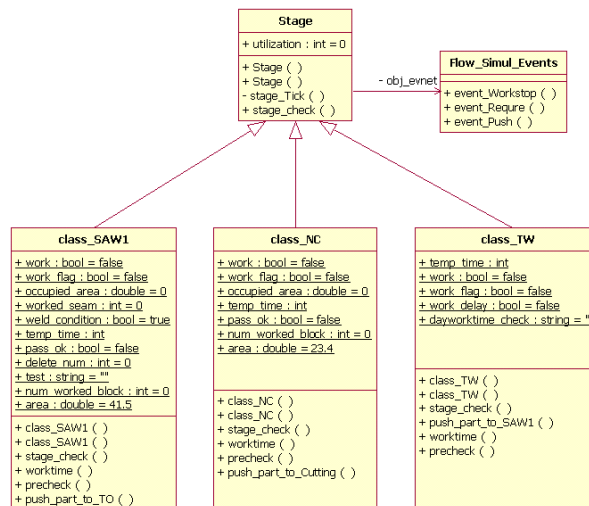


Fig. 17 Class Diagram Modeling of Job Stage in Panel Line

Fig. 17은 각 작업장 마다 하는 일은 다르지만 공통된 속성이나 기능을 포함하는 Stage Class를 설명하고 있다. 시뮬레이터 전체적으로 전역 변수를 뒤서 시간을 증분 하는 것과 마찬가지로 각각의 작업장에서도 Timer 컨트롤을 이용하여 시간의 흐름에 따라 여러 이벤트를 발생하고 결과를 처리하게 하였다. 또한 Stage Class는 Line의 주요 이산 처리인 작업의 종료, 후공정에서의 물량 요구, 물량을 투입하는 것과 같은 이벤트들을 포함하는 Flow_Simul_Events Class를 이용하여 이산 사건을 처리한다. 이 Stage Class를 추상클래스로 활용하여 각 작업장마다 상속받아서 기본적인 기능에 충실하면서도 작업장별로 특색 있는 기능들을 수행하게 개발하였다. Fig. 18은 시뮬레이터의 핵심인 QUEST의 Panel Line 모듈을 C# 프로그래밍으로 옮겨 놓은 Architecture를 보여 준다. 시뮬레이터를 전반적으로 관리하고 전역 변수 및 컨트롤들을 포함하는 Simulator Class와 Panel Line 각 작업장별로 산출식 및 계산 루틴을 담고 있는 Cal_Routine Class, 관련 클래스를 모두 포함하는 Panel_Simul Namespace가 주요 내용이다. 이들 모든 클래스들은 독립적으로 운영되는 파일들로 이루어져 있으며 이것은 클래스의 재사용이나 향후 유지 보수 측면에서 유용하다.

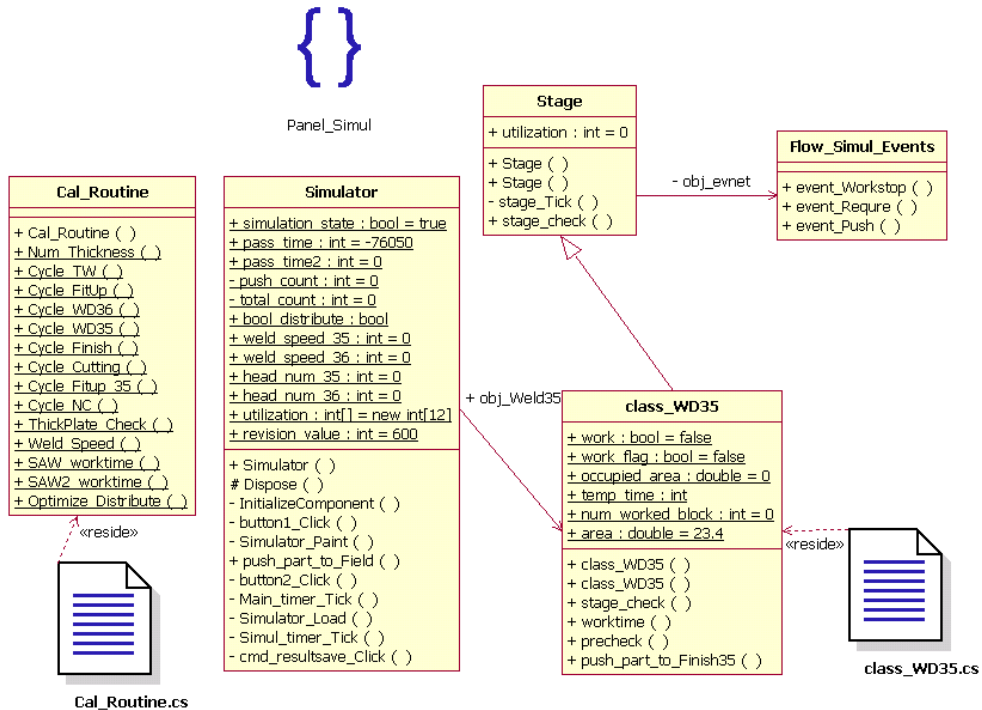


Fig. 18 Class Diagram Modeling of Whole Panel Line Simulator

이러한 객체지향 분석과 같은 체계적 분석을 통해 디지털 생산 시뮬레이션의 프레임워크를 구성하고, 이는 분석 이후 설계, 구현 단계의 수행을 위한 기초 데이터로 활용된다.

User가 시뮬레이션 하고자 하는 물량을 선택하고, 결과를 보여주는 모듈은 서버 측에서 처리가 가능하지만 실제 시뮬레이터로 계산을 담당하는 부분까지 서버가 담당하면 네트워크 부하의 원인이 된다. 그러므로 네트워크 로드 밸런싱을 위해 시뮬레이터 연산에 대해서는 클라이언트 측에서 실행하는 것이 여러 측면에서 이롭다. 이를 위해 기존에는 네트워크에서 프로그램을 내려 받아 로컬에 설치를 하거나 보안의 문제점을 안고 ActiveX를 구동하기도 했다. 본 논문에서는 서버와 클라이언트간의 Application 문제를 해결 하고자 Smart Client 기술을 이용하였다. Fig. 17과 같이 사용자가 Panel Line 시뮬레이터 서버로 접속을 하면 해당 Application을 내려 받고 로컬의 .Net Framework 기반에서 시뮬레이션 연산을 수행한다. 시뮬레이션 후 주요 결과 값만 ‘저장’ 버튼을 클릭함으로써 서버측 DB에 업데이트 된다. 각 클라이언트 컴퓨터에 소프트웨어를 물리적으로 배포하지 않고 실행(레지스트리 등록 불필요)되거나 자동적인 업데이트 기능은 Smart Client만의 이점이며 이로써 Application이 서버와 클라이언트 사이에서 완전히 분리된다.

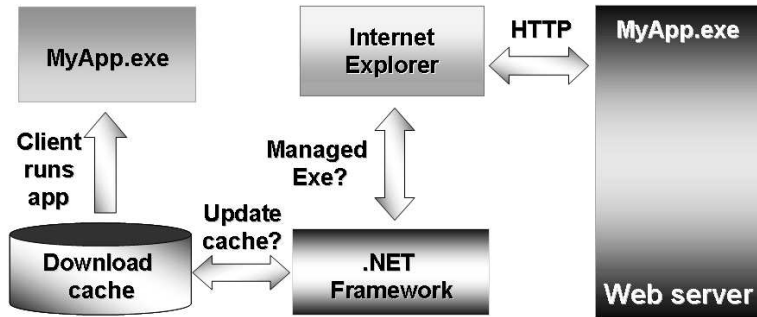


Fig. 19 Architecture of Deploying a Smart Client Application

각 클라이언트측 컴퓨터의 CPU 성능에 따라 시뮬레이션에 소요되는 시간이 달라지는데 Pentium4 2.4GHz를 사용하는 컴퓨터는 10일간의 물량을 처리하는데 42초, Pentium3 800MHz를 사용하는 컴퓨터는 1분 12초가 걸렸으며 이 수치는 QUEST를 사용하여 동일한 물량을 시뮬레이션 하는 시간보다 약 4~5배 빠른 수치이다. QUEST 모델은 시뮬레이션이 수행되면서 Middleware인 DELFOI와 연계하여 DB에 계속적인 쿼리문을 요청하고, 전송하기 때문에 시뮬레이션에 소요되는 시간이 길다는 것이 단점이다.

시뮬레이션 결과는 Fig. 20과 같이 텍스트, 그래프 등으로 나타낼 수 있다. 시뮬레이

션을 실시하고 실제 작업시간, 시뮬레이션 경과 시간은 텍스트 형식으로 표시되고, 작업장별로 전체 평균 공정률, 하루 처리 물량에 대한 작업시간은 그래프로 쉽게 확인할 수 있다. 절단 공정 이후의 물량의 분기에 대해서도 테이블 형식으로 살펴볼 수 있도록 개발 되었다.

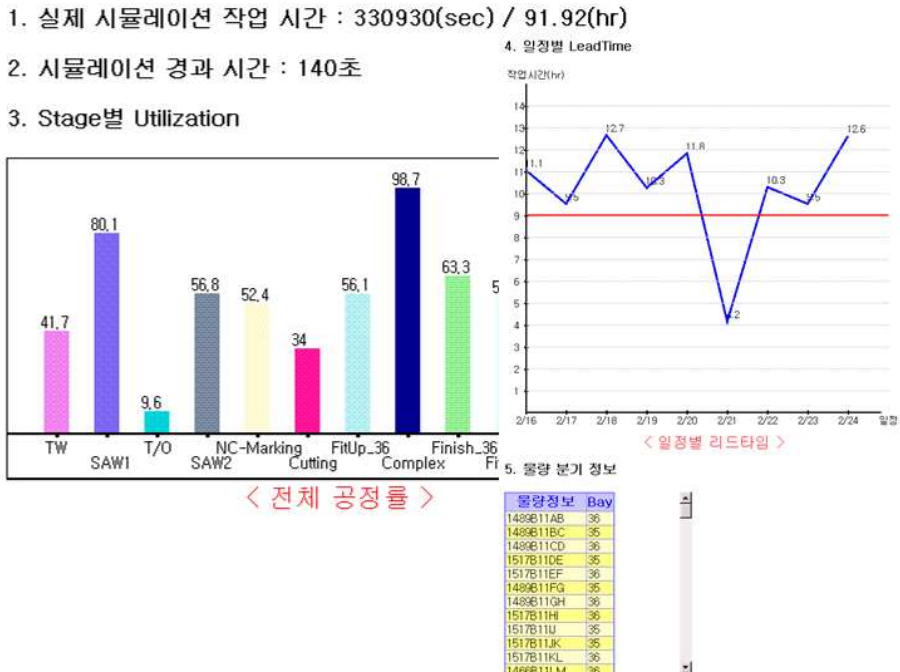


Fig. 20 Various Simulation Results

제 4 장 시뮬레이션 조건 변화와 시뮬레이션 결과 고찰

본 연구에서는 QUEST와 C# 언어로 각각 시뮬레이터를 개발하였다. 2월 16일부터 21일 사이의 실적 데이터가 있는 물량을 시뮬레이션 하여 그 결과를 Table 3 에 나타내었다.

대조군 기 간	실제 작업 시간	QUEST	C# Simulator
2/16~21(일주일)	73.5hr	70.0hr	72.8hr

Table 3. Comparison of Working Time according to Real Time and Simulation Time

Table 3의 결과와 같이 실제 작업시간과 시뮬레이션 모델의 결과 값은 약간의 차는 존재하지만 그 차이는 무시해도 될 정도로 미미하다. 시뮬레이션 결과는 현장의 기사들의 도움으로 확인 되었으며 시뮬레이션 모델이 현장의 Panel Line의 역할을 대신할 동일한 모델로 입증 되었다. 이번 장에서는 QUEST 모델과 C# 언어로 개발된 모델 중에 결과치가 더 비슷하고 다양한 결과 및 빠른 수행능력을 보이는 C# 개발 시뮬레이터로 생산성 향상을 위한 시뮬레이션 조건을 변화 시키고 그 결과를 살펴보기로 한다.

4.1 분기점에서의 로드 밸런싱

현재의 패널라인은 절단 공정이 끝난 후 A Bay와 B Bay로 분기가 이루어지는데, 분기에 대한 정확한 근거 자료가 없는 실정이다. 직장이 적당히 분류를 해서 로드를 줄이는 방향으로 작업을 수행해 왔다. Panel Line과 같은 단속공정의 일정계획(Job Shop Scheduling)은 시뮬레이션을 통해 작업배정규칙의 효과성을 측정하는 많은 연구가 진행되어 있다. 이 시뮬레이션 연구 결과에 의하면 최단처리시간(Shortest Processing Time : SPT)이 최선의 규칙들 중의 하나라는 것이 밝혀졌다. 일반적으로 SPT를 사용하여 일정계획을 수립하면 대기작업의 수가 적게 된다. 개발된 모델에서는 절단 공정 이후의 공정(FitUp, Stiffener Welding, Finish, Silo)들에서 남아있는 물량들을 파악하고, 현재 작업하고 물량의 남아있는 작업시간과 대기물량의 전체 작업시간을 계산하여 작업시간이 적은 Bay로 분기토록 하는 로직(우선순위 규칙(Dispatching Rule) 중 SRPT (Shortest Remaining Process Time) 룰)을 사용하였다. Fig. 21은 시뮬레이터를 활용하여 추출된

결과물 중 하나인 분기로직을 사용하여 Panel이 어느 작업장으로 가야 할지를 보여주는 테이블이다. Table 4는 분기로직을 사용하기 전과 사용 후의 투입된 물량의 수를 보여준다. 분기로직을 사용하기 전에는 Bay별로 약간의 물량 차이(4개)는 있었지만 로직을 사용 후 투입되는 물량의 수가 정확히 일치 하였다. 이는 각 Bay의 작업장별로 부하를 평준화 시키는데 효과가 있다고 판단되며 자세한 내용은 Fig. 20과 같이 SRPT Logic을 사용하면 투입 물량의 평준화 및 작업장 가동률 감소에도 도움을 주는 것으로 나타났다. 결국 이러한 내용들은 전체 작업시간에도 영향을 끼쳐 일반 시뮬레이션 결과는 263,250sec 였지만 SRPT Rule 적용 후 전체 작업시간은 256,410sec로 하루 약 15분간의 작업시간 감소 효과가 있었다. 이러한 수치는 상대적으로 미소하지만 월단위의 시뮬레이션이나 분기별 물량을 사전에 시뮬레이션 하면 그 효과는 정량적으로도 막대하리라 예측된다.

5. 물량 분기 정보

WorkDay	Ship_Num	Block	InfoBay	GotoBay
20040216	1462	B11DS	5	5
20040216	1466	S17P	6	6
20040216	1466	S17S	5	5
20040216	1489	E43P	6	6
20040216	1489	E43S	5	5
20040216	1466	S18P	6	6
20040216	1489	S14LP	5	5
20040216	1489	S14LS	5	5
20040216	1489	B18DP	5	6
20040216	1489	B18DS	6	6
20040216	1517	S16BLP	5	5
20040216	1517	S16BLS	5	6
20040217	1466	D14P	6	5
20040217	1489	SU21D1P	5	6
20040217	1489	SU21D1S	5	6
20040217	1466	D13P	5	6
20040217	1489	S15LP	6	5

Fig. 21 Results of Load Balancing using SRPT Rule

Simulation 종료시 Line을 빠져나간 물량		
사용한 로직	A Bay로 들어간 물량	B Bay로 들어간 물량
Normal Rule	30	34
Dispatching Rule	32	32

Table 4. Comparison of Dispatching Count according to Normal and SRPT Rule

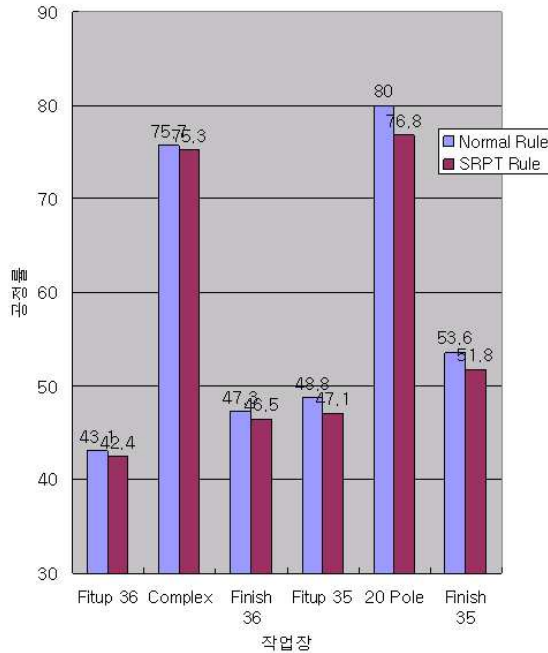


Fig. 22 Utilization of Job Stage for escorting a Cutting Stage using SRPT Rule

4.2 공법이나 설비 변경시 생산성의 변화 예측

시뮬레이션을 통하여 현실에서는 적용하기 힘든 설비나 공법의 변경시 효과를 사전에 예측할 수 있다. 생산라인의 한 작업장에서 설비를 증설해야 하는 경우 해당 작업장 뿐만 아니라 전체 라인에 미치는 영향, 선·후 공정에서의 공정률 변화, 작업 인원의 배치 등 여러 가지 요소들을 고려해야 한다. Panel Line 시뮬레이터는 시뮬레이터 설계 단계에서 부하 공정으로 평가 받는 35,6 Bay의 Longi 용접장의 설비를 증설하거나 용접속도를 제어함으로써 생산성의 변화를 예측할 수 있도록 개발 되었다. 구체적으로는 현재 용접 헤드수는 5개이지만 6개, 7개로 늘렸을 때의 결과나 현재 110CPM의 용접속도에 변화를 주면 어떤 결과가 나타나는지를 전체 작업시간과 공정률의 변화를 통해 예측하게끔 하였다.

시뮬레이션은 4.1절과 같은 동일한 물량으로 SRPT Rule을 적용하고 시뮬레이션 한 Normal Case와 헤드수를 3개에서 5개로 증설한 경우인 Case 1, 용접속도를 110CPM에서 200CPM으로 증가해서 시뮬레이션 한 Case 2, 두가지 옵션을 모두 적용한 Case 3로 나누어 시뮬레이션하고 그 결과를 Table 5에 나타내었다.

Case	전체 작업시간
Normal 5Head, 110CPM	256,410sec (71.2hr)
Case 1 7Head, 110CPM	253,620sec (70.4hr)
Case 2 5Head, 200CPM	253,600sec (70.4hr)
Case 3 7Head, 200CPM	254,610sec (70.7hr)

Table 5. Comparison of Total Work Time according to Change of Welding Option

Table 5의 결과를 살펴보면 헤드수를 증가 시키는 것과 용접속도를 향상하는 방법 모두 일반 작업과 비교하여 생산성의 증가를 정량적으로 예측할 수 있다. 용접 헤드수를 증가시키고 용접속도를 높이면 더 높은 생산성을 기대할 수 있지만 이들 작업장은 작업 처리 속도가 빨라지는 대신 후공정에서 이를 적시에 처리하지 못해 물량을 많이 적치하는 것을 발견할 수 있었다. 즉 이 결과는 Line의 물류 흐름 특성상 특정 Job Shop의 생산성이 증가해도 선·후 공정에서 적절히 처리할 능력이 부족하면 전체적으로 생산성을 떨어뜨리는 것을 보여준다. 헤드수를 늘리거나 용접속도를 증가시킬 경우 일주일 물량에 대해 비슷한 공기단축 효과를 볼 수 있다. 따라서 가장 효율적인 공기단축 방법으로는 비용을 고려하여 용접속도를 현재 110CPM에서 200CPM로 증가시키는 것이다. 현재 사용되고 있는 용접와이어는 1.2 ϕ 로서 와이어 직경을 1.4 ϕ 나 1.6 ϕ 로 변화시키면 용접속도도 200CPM까지 변경 가능하고 시뮬레이션 결과처럼 공기단축 효과를 예측할 수 있다. Fig. 23은 각각의 Case에 따른 작업장의 공정률 변화를 나타낸다. 공기단축과 공정률 변화나 선후행 공정의 공정률 변화와 경제성을 비교하여 설비의 변경이나 공법에 변화를 줄 수 있다. 시뮬레이터는 이와 같은 문제가 주어 졌을 때 Back Data를 제시할 수 있는 유용한 도구이다.

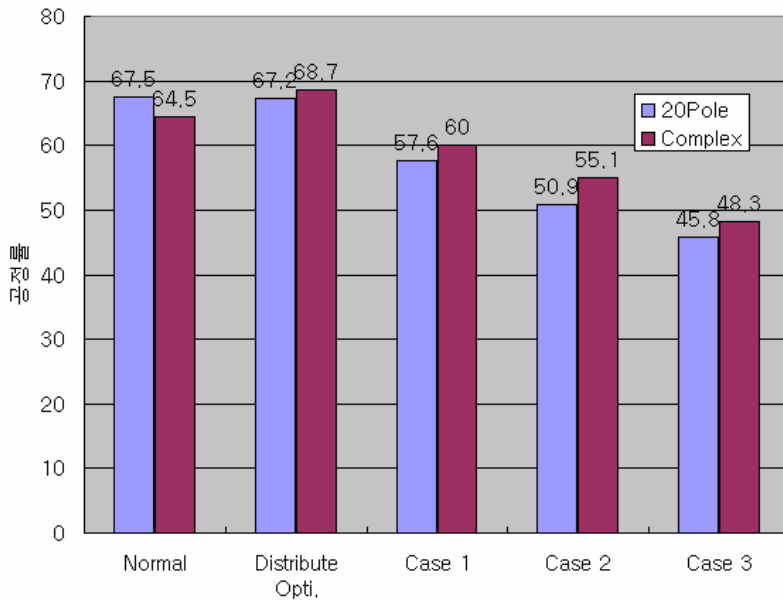


Fig. 23 Utilization Change according to Welding Case

4.3 물류 투입순서 변경시 생산성의 변화 예측

생산설비의 증설이나 공법의 변경과 같이 추가 비용을 투입하는 방법 외에 생산성 향상을 도모 하는 방법이 Line에 투입되는 물량의 순서를 조정하는 것이다. Panel Line과 같은 여러 작업장을 거치는 물량에 대해서는 최적화 된 작업순서를 결정하기가 쉽지 않다. 물류 분야에서 사용되는 존슨(Johnson 1954)의 규칙을 이용하여 최종작업이 마지막 작업장에서 완료되는 시간, 즉 모든 작업이 끝나는 시간이 최소가 되도록 작업순서를 결정한다. 이 규칙은 최종작업의 완료시간 뿐만 아니라 작업장들의 총유휴시간도 최소화 시킨다.

(1) Johnson's Rule 이란?

두 개의 작업장을 거치는 경우

단계 1 : 각 작업마다 작업장 1과 작업장 2에서의 처리시간을 산정한다.

단계 2 : 작업장 1, 2에 관계없이 처리시간이 가장 짧은 작업을 선택한다. 이 가장 짧은 처리시간이 작업장 1에서 발생하면 그 작업을 가능한 앞순위에 놓고, 작업장 2에서 발생하면 그 작업을 가능한 나중순위에 놓는다.

단계 3 : 단계 2에서 순위가 결정된 작업은 고려대상에서 제외한다.

단계 4 : 모든 작업의 순서가 결정될 때까지 단계 2와 단계 3을 반복한다.

(2) Panel Line Simulator에 Johnson's Rule 적용

두 개의 작업장의 물류 투입 순서를 고려하는 Johnson's Rule을 복잡한 여러 작업을 거치는 Panel Line에 적용하면 두 개의 작업장을 확장시킨 개념으로 접근 가능하다. 특정 작업장에 들어가는 물량의 작업시간을 산출식을 통해 미리 계산하여 그 결과치가 작은 순서대로 물량을 투입하여 시뮬레이션을 수행 하였다. 아래의 Fig. 24는 T/W, SAW, Longi Welding 작업장별로 Johnson's Rule을 적용하여 물량을 정렬하여 시뮬레이션한 결과이다.

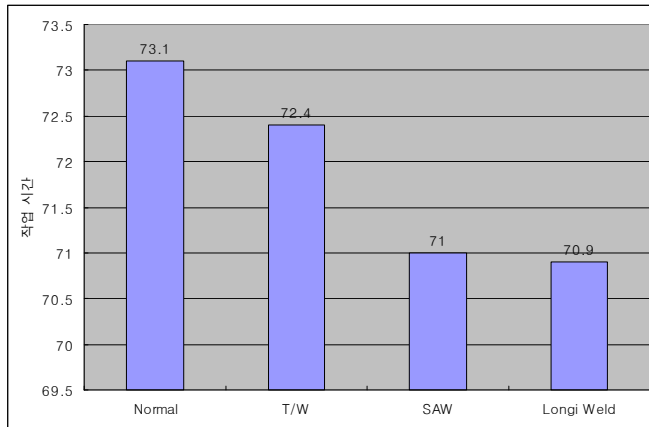


Fig. 24 Change of Productivity using Johnson's Rule

Fig. 24의 결과는 동일한 물량들에 대해서 물량의 투입순서를 변경하는 것이 생산성의 증가와 귀결되는 것을 보여준다. T/W 작업장과 같이 부하가 낮은 작업장은 그 효과가 적지만 Plate를 맞대기 용접하는 SAW 작업장이나 Longi를 용접하는 작업장처럼 가동률이 높은 작업장은 Johnson's Rule 적용효과가 크다고 할 수 있다. 물량의 투입순서를 미리 바꾸어 봄으로써 그 결과를 예측하는 것도 시뮬레이션의 또 다른 장점중의 하나이다.

4.4 시뮬레이터 활용 방안

(1) 조업도 평가 기준 마련

현재 실제 공정에서 사용되는 여러 변수들을 무시하고 Seam과 Longi로 대략의 용접 길이만을 판단하고, 이를 Panel의 생산 갯수로 환산한 전체 용접길이를 조업도를 평가하고 있다. Container와 Oil Tanker Block을 비교하면 같은 용접길이(현재의 기준에서 사용하는 같은 조업도)라도 Oil Tanker는 용접이 대부분 1 회로서 끝나고 공정상 복잡한 부분도 없다. Container의 경우는 60t 이상의 후판이 단차로 들어가게 되고 이는 SAW 공정에서 부하(후판의 단차조정, 용접시 유압 Pusher로 조정)로 작용하게 된다. 후판에는 Stiffener 용접시 Flat-Bar가 붙게 되는데 이때 각장을 고려하여 여러 번(2~3회) 용접하게 된다. 후판 용접시 대부분 용접속도가 늦어지는 점도 고려해야 할 사항이다. 같은 전체 용접 길이라도 Container Block은 조업도를 높게 고려해야 할 것이다. 현재 이와 같은 특성은 무시된 채 제한된 몇 개의 Factor만을 가지고 조업도를 평가하므로, Simulation을 통해 모든 Factor를 고려하여 정도가 높은 조업도를 제시할 수 있다.

(2) 배원 계획 수립

직장이나 기사 수준이 아닌 현장의 반장들은 장기보다는 주로 단기 계획에 중점을 두고 업무를 진행하고 있다. 예를 들어 라인에 있는 각 반별로 인원을 조정한다거나, 정상일과 이후의 작업 지시를 하는 경우이다. 본 물류 시뮬레이터를 사용하면 특정일, 특정 스테이지에 대한 부하를 미리 파악할 수 있으므로 부하가 많은 작업장으로 인원을 늘리는 조치 등으로 전체 작업장의 부하를 평준화할 수 있다. 그리고 정해진 물량의 Lead Time을 미리 알 수 있으므로 Lead Time이 많은 날은 정상 업무 이외에 잔업, 야간작업의 분량도 산정할 수 있게 지원해준다.

(3) 생산라인의 설비계획 이해 증진

시뮬레이션 모형을 창조하는 과정은 시뮬레이션 대상이 되는 활동들에 대한 상세한 이해와 기록을 필요로 한다. 시뮬레이션 모형을 활용하여 많은 예외와 흐름대안들을 확인하게 된다. 설비계획의 변경안 검토나 Panel Line의 전체적인 상황을 경영진에게 이해시킬 때에도 시뮬레이션을 활용할 수 있다.

제 5 장 결 론

본 연구에서는 Panel Line의 물류 시뮬레이션을 위하여 공정분석을 선행하고, 각 작업장의 내부 로직을 구성, 이를 이용하여 이산 사건 기반의 시뮬레이터(QUEST 모델, C# 언어로 개발된 모델)를 구축하고 검증하였다.

본 연구의 결과물이 보여주는 단적인 예(물량을 처리하는데 소요되는 시간, 제한시간에 얼마나 많은 물량을 처리하는가) 이외에도 시뮬레이터를 여러 가지 측면에서 활용 가능하다. 예를 들어 작업장 공정률을 살펴봄으로서 특정 작업장에 부하가 많이 걸리면 인원을 더 배치해서 Line 전체적으로 부하율을 일정하게 조정한다거나, 장비개선을 도모함으로써 문제를 해결 할 수도 있다. 물론 본 연구에서는 Stiffener Welding 작업장만 시뮬레이션 옵션을 주도록 되어 있지만, 다른 작업장도 확장시켜 시뮬레이션 해 볼 수 있음을 의미한다. 전체 Lead Time을 분석해서 특정일에 작업시간이 많이 걸리면 인원을 더 투입하거나, 잔업·야간작업을 계획할 때 Back Data로 활용 할 수 있다.

현재 사용되고 있는 조업도 평가의 직접적인 기준으로도 사용 가능하다. 시뮬레이션 결과를 더 많은 Case에 대입하여 검증 작업을 거쳐야 하겠지만 현재 사용하고 있는 제한된 Factor(Panel, Seam, Longi)로 공정의 특성을 고려하지 않은 조업도 사용 보다는 실제 공정에서 중요시 되는 여러 Factor(회전수, Machine Speed, 기본 Dimension, 작업 Area 등)를 모두 고려한 정도 높은 조업도 제시가 본 연구의 성과 중 가장 큰 의미가 있다고 본다.

C# 개발언어로 이산 사건 처리를 가능하게 하는 시뮬레이터를 독자적으로 개발하였으며 이는 고가의 소프트웨어를 대신하여 시뮬레이터를 구축할 수 있음을 의미한다. 또한 시뮬레이션 수행시간의 단축, 다양한 결과 처리, 데이터 가공의 측면에서 시뮬레이션 툴 사용 보다 우수하였다.

QUEST는 단독으로도 사용이 가능하나, Delmia사의 다른 툴의 모델을 불러 통합적인 제조 라인을 구성할 수 있다. 예를 들어 Delmia/IGRIP 이나 Virtual NC 같은 다른 Delmia 솔루션에서 만들어진 시뮬레이션 모델을 읽어 들여 그대로 사용하는 것도 가능하다. Virtual NC와 IGRIP을 활용하여 단위 작업장의 Cell 모델을 구축하고 이를 QUEST와 연계하여 시뮬레이션 하면 정확도의 향상은 물론이고, 진정한 의미의 Digital Ship Yard 완성의 토대가 될 수 있을 것이다.

참고 문헌

- [1] 이상범, “현대 생산·운영 관리”
- [2] Wallace J. Hopp and Mark L. Spearman, "Factory Physics : Foundations of Manufacturing Management"
- [3] 신종계, 이장현, 우종훈, “디지털 선박생산 (**Digital Shipbuilding**) 개념”, 대한조선학회논문집, 제38권 제1호, pp. 54-62, 2001
- [4] 우종훈, 이광국, 정호림, 이장현, 신종계, 권영대, “디지털 조선소 구축을 위한 물류 모델 프레임워크”, 대한조선학회 춘계 학술대회 논문집pp. 436-448, 2004
- [6] **Gary B. Shelly and Thomas J. Cashman and Harry J. Rosenblatt**, "**Systems Analysis and Design**"
- [7] 강현진(2005), “유전자 알고리즘을 이용한 조선 소조립 로봇용접 공정 최적화 및 3D 시뮬레이션을 통한 검증”
- [8] 이창환(2005), “가상 시뮬레이션 기술을 이용한 일일 탑재 공정 연구”
- [9] 김탁곤(1999), “시스템적 모델링 방법을 이용한 이산사건 시스템 모델링”

감사의 글

대학원 생활이 막바지에 접어듭니다. 돌이켜 보면 힘든 시기도 많았지만 약간의 지식을 떠나 저에게는 더 큰 무언가를 배우고 내면적으로 성장하게 한 시기인 것 같습니다. 부족한 논문을 마무리하면서 일상에서 잠시나마 벗어나 제 자신을 돌이켜 보고, 지면을 빌어서 그동안 고마웠던 분들에게 감사의 글을 올리고자 합니다.

우선 부모님께 가장 먼저 감사의 뜻을 전하고 싶네요. 오늘의 제가 있도록 해주셨고, 학부 4년도 모자라 대학원 공부를 더 한다고 했을 때 선뜻 그러라고 하시며 격려하고 지원해 주신 점 정말 고맙습니다. 독립의 기로에 서서 사회에 나갈 준비를 하지만 항상 부모님에 제게 가르쳐 주신 뜻에 어긋나지 않는 아들이 되도록 노력 하겠습니다. 부모님 언제나 사랑합니다.

이 논문이 나오기까지 실질적으로 저를 지도하시고 후원하신 박주용 교수님께 진심으로 감사하다는 말을 전하고 싶습니다. 실험실에서 가장 오래 생활하고, 다른 뛰어난 선배들과 달리 애(?)를 많이 태운 제자로 기억에 남을 듯 하네요. 언제나 모자란 저에게 애정 어린 충고와 격려 해주신 점은 영원히 잊지 못할 것 같습니다. 논문을 심사해주신 조효제 교수님과 박석주 교수님, 여러 년 동안 지도하신 교수님들께도 감사드립니다. 2년간 같이 과제를 수행하면서 물적·정신적으로 지원을 아끼지 않았던 삼성중공업의 김세환 부장님에게도 진심어린 고마움을 표합니다. 오늘의 후배 동환이가 있기까지 힘쓴 여러 선배님들, 위로 최우현 부장님, 경철형, 현철형, 상수형, 대호형, 장곤형, 태인형, 재용형, 현진에게도 감사드리고 아울러 잘 따르고 많이 도와준 창섭이와 실험실 가족들에게도 고맙단 말을 전하고 싶네요. 특히 곁에서 짓궂은 장난을 잘 받아주고, 말동무가 되었던 두원아... 정말 고맙다.

과제와 논문 준비를 위해 삼성중공업에서 많이 도와주신 한상동 책임, 정석찬 기사, 정보시스템의 권영대 대리, XINNOS 멤버인 종훈형, 춘재형, 광국형이 있었기에 오늘의 논문이 완성될 수 있었습니다. 정말 감사드립니다.

항상 삭막한 공부에 감성적으로 도움을 주신 Violinist 한행래 선생님, BAO 회원님들, 대학교 98학번 동기들, 태양촌 동문인들 그리고 주변 모두에게 감사드리고 싶네요. 이제 사회에 첫발을 내디디는데 항상 그래왔던 것처럼 자신 있고 힘차게 전진해 나가겠습니다. 계속 성장해 가는 모습 지켜봐 주세요.

마지막으로 장소를 몇 차례 옮기고 이름도 바뀌었지만 6년간 저와 함께한 실험실과 지면에 미처 실지 못한 분들과 이 작은 기쁨을 함께 하고 싶습니다.

2006년 1월 9일 실험실에서
동환 드림