

고정정반 블록조립 작업장의 일정계획 시스템 개발

고시근¹ · 박주철² · 최용선³ · 주철민⁴

¹부경대학교 산업공학과/²울산대학교 산업공학과/³인제대학교 산업시스템공학과
⁴동서대학교 정보시스템공학부

Development of a Block Assembly Scheduling System for Shipbuilding Company

Shie-Gheun Koh¹ · Ju-Chull Park² · Yong-Sun Choi³ · Chul-Min Joo⁴

This paper introduces a production scheduling system for the block assembly shop in a shipbuilding company. We are to make a schedule for the assembly shop and related subassembly production shop. The objectives of the scheduling system are balancing of the workload of the subassembly shops as well as the assembly shop. There are a number of technological and resource constraints including assembly space restriction which is most important and critical resource in the shop. It is very hard and time consuming to consider the two problems, the workload balancing and the spatial allocation problem, simultaneously, and hence, we analyze the two problems independently. The first problem has already been introduced(고시근, 1996), and the overall appearance of the system and the spatial allocation algorithm are presented in this paper.

1. 서 론

본 연구에서는 H 중공업 조선사업본부에서 효율적인 중기일정계획을 위해 개발한 선각일정계획 최적화시스템(HYPROSS-II)(박주철 외, 1997)의 일부인 곡블록의 생산일정계획 시스템을 소개한다.

선박의 건조에 있어 기본단위가 되는 블록은 철판의 절단, 성형, 소조립, 중조립, 대조립 등의 여러 과정을 거치면서 조립되어 도크에서 선박의 형태로 탑재된다. 블록은 조립작업시 바닥면의 형상에 따라 평블록과 곡블록으로 분류되는데 블록의 조립과정도 이에 따라 구분된다. 평블록은 바닥이 평면이므로 Roller 위를 이동해 가며 조립되는 흐름생산방식이 사용되고 곡블록은 바닥면이 곡면이므로 높이를 조절할 수 있는 Pipe Jig가 설치된 고정된 위치(이하 고정정반으로 표기)에서 조립되고 있다. 전체 시스템과 본 시스템의 관계를 <그림 1>에서 보여주고 있다.

본 시스템에서 대상으로 하고 있는 곡블록 공장은 그 내부에 곡블록 조립장뿐 아니라 성형, 판계, 평판 중조립, 곡판 중조립 등의 곡블록 조립에 필요한 선행 작업장들을 포함하고 있다. 성형은 평평한 철판을 원하는 형태로 구부리기 위해 프레스나 열처리를 하는 공정이고, 판계는 여러 개의 철판을 용접하여

더 넓은 철판을 만드는 공정이며 중조립은 블록을 구성하는 단위블록(한 개의 블록은 1~4개의 단위블록으로 구성됨)을 조립하는 공정이다. 그런데 곡블록이라 하더라도 그 블록을 분해해 보면 곡판과 평판의 결합으로 되어 있으므로 곡블록을 위한 중조립공정은 평판의 중조립공정과 곡판의 중조립공정으로 구분되는 것이다. 평판의 중조립 작업은 평블록의 조립공정과 마찬가지로 곡블록 생산공장 내의 흐름생산부서에서 수행되며, 곡판의 중조립은 고정정반으로 이루어진 중조립부서에서 이루어진다.

곡블록의 조립은 전술한 바와 같이 정반에서 이루어지며 각각의 정반은 U1-Bay, U2-Bay 등과 같이 Bay라고 부른다. 반면 조립에 필요한 선행 작업장(성형, 판계, 중조립)은 Subbay라고 부른다.

고정정반 작업장은 정반의 Space가 가장 중요한 생산자원이며, 따라서 블록별 조립완료 요구일(납기)을 준수하며 곡블록 공장의 가장 효율적인 운영방법을 추구하는 본 시스템의 가장 중요한 제약조건은 정반사용에 관련된 것이다. 즉, 블록탑재(Erection)계획에서 생성된 각 블록의 조립완료 요구일을 만족하며 고정정반의 Space제약과 Subbay의 능력제약을 고려한 효율적인 곡블록공장의 생산일정을 수립하는 것이 본 시스템의 목적이다.

고정정반의 활용에 관련된 연구로는 박경철 등(1995)이 특

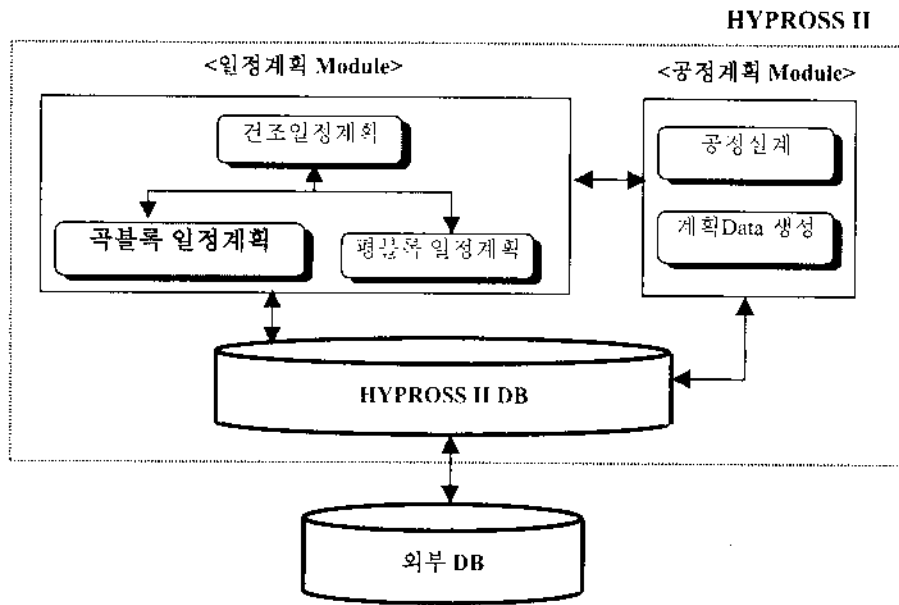


그림 1. HYPROSS II 시스템의 구조.

수한 형태의 정반에서 이루어지는 조립작업에 대해 개발한 시스템이 있다. 그들의 연구대상 작업장은 작업장의 폭이 좁아 블록들을 한 줄로 나열하는 방식을 택하고 있다. 반면 Lee *et al.*(1996)이 개발한 시스템은 2차원 평면 위에 블록들을 자유롭게 배치하는 방식을 따르고 있다. 또한 Lozano-Perez(1983)는 하나의 블록다각형(Convex Polygon)이 기존에 배치되어 있는 다른 블록다각형에 겹쳐지지 않게 놓여지는 방법을 연구하였고, O'Rourke *et al.*(1982)은 두개의 블록다각형이 겹쳐지는 부분을 찾는 알고리즘을 연구하였으며 Preparata and Shamos(1985)는 관련된 여러 가지 기하학적인 문제를 다루고 있다. 본 시스템의 대상이 되는 작업장도 이들이 다른 시스템과 유사한 형태로 블록들의 배치가 이루어진다. 그러나 본 시스템에서는 정반의 면적활용률뿐 아니라 조립 Bay의 부하, Subbay의 부하를 포함한 다양한 요인들을 고려하고 있다.

본 논문의 구성은 2장에서 시스템의 구조 및 기능을 결정변수와 제약조건 및 목적을 통해 정의한다. 2차원 고정정반에 블록다각형으로 가정된 블록을 배치해 나가는 정반배치 Module에 대해서는 3장에서 설명하며 4장에서는 시스템의 개발결과를 실제 컴퓨터 화면을 사용해 간략히 소개한다.

2. 시스템의 구조 및 기능

본 시스템의 개략적인 구조가 <그림 2>에 나타나 있다. HYPROSS II DB로부터 블록별 조립완료 요구일과 각종 블록관련정보를 전달받아 고정정반에의 배치가능성과 Subbay의 부하를 고려하여 곡블록공장의 전체적인 생산일정을 자동 또는 수동으로 계획하는 것이다. 계획결과는 각종 형식의 화면 및 인쇄물로

출력할 수 있고 DB 또는 File에 저장하여 추후에 사용할 수 있다.

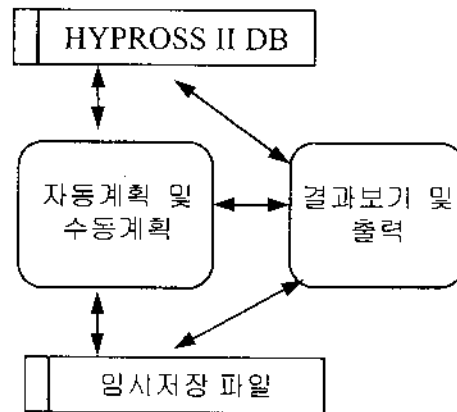


그림 2. 본 시스템의 구조.

수동계획은 본 시스템에서 제공하는 각종 정보(부하상황, 정반사용현황 등)를 참조하여 사용자가 직접 마우스 작업을 사용해 특정블록을 특정일정으로 특정정반의 특정위치에 배치하는 것이다. 특정기간내 전체블록을 수동계획으로 처리할 수 있고 일부만 수동계획 후 자동계획을 실행할 수도 있다.

자동계획은 자동일정계획과 자동정반배치의 2개의 Module로 구성되어 있다. 조립일정계획 Module은 유전알고리즘을 이용해 Subbay 및 조립 Bay의 부하가 균형화되는 작업일정을 작성해준다(고사근, 1996). 이 과정에서 다른 제약조건들은 고려되지만 작업장 배치가능성 제약은 직접 고려되지 못하고 전체 정반 면적 대비 블록면적의 비율을 제약조건으로 도입해 간접적으로 고려된다. 따라서 조립일정계획 모듈에서 작성된 작업일정은 부하균형화 측면에서는 가장 좋은 일정이나 배치가 불

가능한 경우가 발생한다. 따라서 이 일정을 기초로 하여 자동 정반배치모듈에서 배치가능성을 확인하여야 한다. 정반배치 모듈에서는 가능한 한 일정계획모듈의 결과일정을 준수하도록 블록이 과연 그 시점에 어떤 정반에 투입가능할 것인지, 가능하다면 그 정반 내에서 어느 위치에 어떤 형태(블록은 정규 상태를 비롯하여 90, 180, 270만큼 회전배치 가능)로 배치되는 것이 유리할 것인지를 분석하는 것이다.

대상 시스템을 보다 명확히 정의하기 위해 시스템의 결정변수, 제약조건, 목적을 설명하면 다음과 같다.

[결정변수]

본 시스템이 결정해야 할 변수에는 각 블록별 조립작업 착수 일자, 조립작업 위치 (특정 정반 및 정반내 위치), 조립블록의 배치형태(회전여부), Subbay 일정 등이 있다

[제약조건]

- 1) 2차원 정반 위에 블록의 배치가능성(Space 제약)
- 2) 블록의 중량, 크기, 작업상에 따른 특정정반의 특정 블록에 대한 작업가능성
- 3) 블록의 납기(조립완료 요구일)
- 4) 스톱기간: 블록의 조립완료 후 후속작업까지 기다리는 시간제약
- 5) P블록과 S블록 조건 : 좌현(Port) 및 우현(Starboard)블록은 관리의 편의를 위해 동일정반에서 같은 날짜에 조립착수하여 같은 날짜에 반출되어야 한다.
- 6) 정반가장자리에 배치되는 블록의 배치형태 : 정반의 옆쪽(통로변)에는 가능하면 대형블록을 배치하여야 하며 또한 작업의 편의를 위해 꼭부위가 정반안쪽으로 향하게 배치되어야 한다.

[목적]

- 1) 계획기간에 걸친 일자별 부하의 평준화
 - 2) 조립작업장간의 부하평준화
 - 3) 각 Subbay에 대한 일자별 부하의 평준화
- 보는 바와 같이 본 시스템은 수많은 결정변수와 제약조건, 그리고 복잡한 목적함수를 갖는 방대한 최적화문제이다. 이 문제를 한꺼번에 푼다는 것은 거의 불가능하다고 보아므로 전술한 바와 같이 본 시스템에서는 문제를 두 단계로 분리하여 해결하기로 하였다. 즉, 배치가능성제약을 완화하여 전체 면적에 대한 제약으로 변형한 뒤 블록별 조립착수일과 작업정반을 결정하고 그 결과에 의거하여 정반배치작업을 수행하는 것이다.

3. 정반배치 알고리즘

3.1 블록투입 알고리즘

본 절에서는 시스템에서 사용되는 정반배치 Module의 일부로 기존의 작업중인 블록들 사이에 새로운 블록의 조립공간을 확보하는 알고리즘에 대해 다음과 같은 5단계로 나누어 설명한다.

(1) 블록의 블록다각형화

다양한 형태의 입체물인 하나의 꼭지점을 조립작업이 행해지는 고정정반에서 그 블록이 차지하는 투영형상에 따라 평면으로 가정하고 그 평면 형상을 본 알고리즘에서 사용 가능하도록 블록다각형으로 다시 가정하여 <그림 3>과 같이 다각형의 꼭지점 중에서 가장 하부(하부의 점이 두 개일 때는 둘 중의 좌측)점을 기준점(Reference vertex)으로 하여 반시계방향으로 각 꼭지점을 기억함으로써 블록을 표현한다. 물론 다각형이 아닌(모서리가 곡선) 블록도 있으나 이 경우에는 유사한 다각형으로 전환해주는 과정을 거치면 된다.

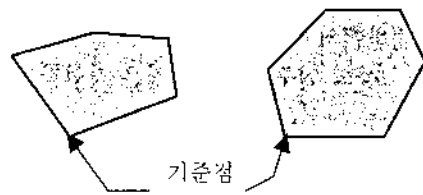
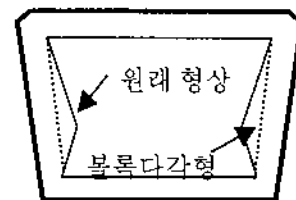


그림 3. 블록의 표현방법.

(2) 기 배치블록에 안전면적 추가

블록의 형태가 불록형이 아닌 경우도 소수이지만 존재한다. 이 경우는 <그림 4>의 점선과 같이 불록형으로 전환해준다. 또한 블록과 블록 사이에는 작업상의 편의와 안전성을 확보하기 위해 1 m 정도의 안전거리를 둔다. 따라서 새로운 블록의 투입시 기 배치블록과 안전거리를 확보하기 위해 기 배치블록은 원래 형상보다 1 m 정도 확대된 형상으로 가정한다.



안전거리 추가한 블록다각형

그림 4. 블록의 불록다각형화 및 안전거리 확보.

(3) 정반형상의 표현

고정정반은 직사각형이지만 정반에 따라 정반 내부에 Dead Space가 있는 경우가 있다. 이 경우에는 <그림 5>와 같이 정반의 형태를 직사각형이라고 가정하고, Dead Space에는 직사각형의 가상블록들이 항상 배치되어 있다고 생각한다.

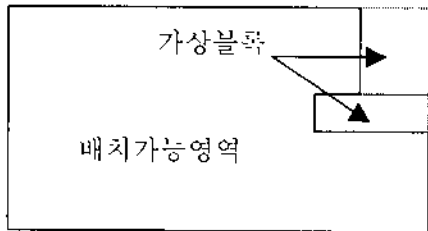


그림 5. 조립 작업장의 형태.

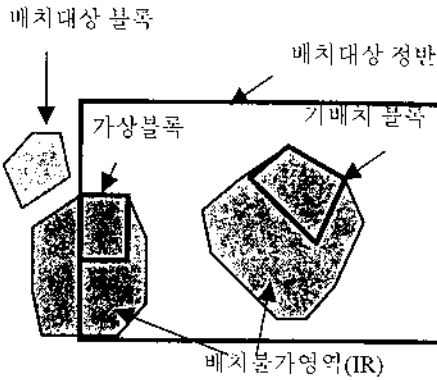


그림 7. 기존블록을 고려한 블록배치가능 영역(IR).

(4) 새로운 블록의 배치가능 영역 찾기

새로운 블록을 배치하고자 할 때에는 정반의 형상(가로와 세로길이, Dead Space)과 기 배치블록의 배치형태 등을 고려하여 새로운 블록이 배치될 수 있는 위치(새로운 블록의 기준점이 놓여질 수 있는 영역)를 찾아야 하며 그 절차는 다음과 같다.

① 정반형상(Dead Space를 포함한 직사각형)에 기준한 배치가능영역(PR: Possible Region) 계산.

새로운 블록을 배치하려고 하는 정반의 형태를 고려하여 새로운 블록의 기준점이 놓여질 수 있는 영역을 계산하는데 그 방법은 블록의 기준점으로부터 각 방향(좌우상하)의 끝점까지의 거리를 계산하여 정반의 각 변으로부터 그 거리를 빼주는 것이다(<그림 6> 참조).

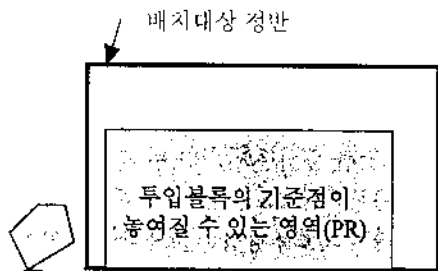


그림 6. 정반형상을 고려한 블록배치가능영역(PR).

② 기 배치블록(가상블록도 포함)에 기준한 배치불가능영역(IR: Impossible Region) 계산.

정반의 Dead Space를 표현하는 가상블록(직사각형)과

이미 배치된 실제 블록(안전거리가 더해진 블록다각형)을 고려하여 새로운 블록의 기준점이 놓여질 수 없는 영역을 Lozano-Percz(1983)가 제안한 방법을 이용해 계산한다(<그림 7> 참조).

③ 위의 두 종류의 영역을 고려한 실제 배치가능영역(FR: Feasible Region) 계산.

<그림 6>의 배치가능영역(PR)에서 <그림 7>의 배치불가능영역(IR)에 속하는 부분을 제외시키면 그 영역이 실제로 블록의 기준점이 놓여질 수 있는 영역(FR)이 된다(<그림 8> 참조).

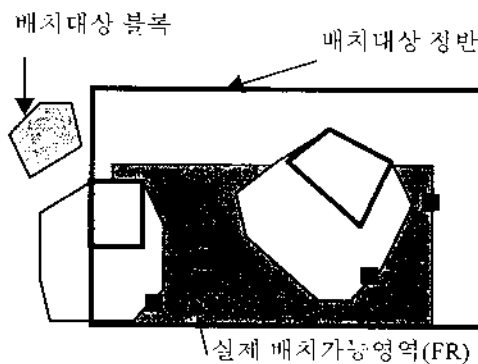


그림 8. 투입블록의 실제 배치가능위치.

(5) 새로운 블록의 배치 후보점

<그림 8>에서 보는 바와 같이 새로운 블록의 기준점이 놓여질 수 있는 후보지는 영역으로 나타나기 때문에 새로운 블록의 배치 후보점은 무수히 많다. 그런데 이 후보점은 영문자로 표시한 바와 같이 다음의 4가지 유형으로 분류할 수 있다.

A 유형은 정반의 Dead Space를 표시하는 가상블록과 관

계된 점들로서 이 유형의 점에 새로운 블록의 기준점을 위치시키면 새로운 투입블록이 정반의 Dead Space에 붙어서 놓여지게 된다.

B 유형은 FR의 내부점(Interior Point)으로 이 유형의 점에 새로운 블록의 기준점을 위치시키면 새로운 투입블록이 정반의 가장자리와 기존의 블록에서 떨어져서 놓여지게 된다. 따라서 정반의 변적활용률이 떨어지는 결과를 초래하게 되므로 우리의 시스템에서는 이러한 유형의 배치 후보점을 고려대상에서 제외한다.

C 유형은 기존블록과 관계된 점들로서 이 유형의 점에 새로운 블록의 기준점을 위치시키면 새로운 투입블록이 기존 블록과 안전거리만큼 떨어져 놓여지게 된다.

D 유형은 <그림 6>에 표시된 PR의 가장자리에 놓여진 점으로 이 유형의 점에 새로운 블록의 기준점을 위치시키면 새로운 투입블록이 정반의 가장자리에 붙어서 놓여지게 된다. 따라서 A 유형과 D 유형은 같은 종류라고 할 수 있다.

비록 B 유형을 제외하였지만 나머지 유형의 점들도 FR의 경계선을 따라 무한히 많은 수의 점들을 포함하고 있으므로 분석의 편의를 위해 유한개의 점들만 고려하기로 한다. 고려대상점(Feasible Point)은 FR의 경계점(Boundary Point)중 꼭지점(Extreme Point)과 꼭지점 사이의 경계선을 따라 1m 간격으로 추출하였다. 이 유한개의 후보점을 찾는 방법은 다음과 같다.

- ① PR의 4개 꼭지점에서 IR에 속하는 점 제외
- ② PR과 IR의 교점 중 다른 IR에 속하는 점 제외
- ③ PR에 속하는 IR의 꼭지점 중 다른 IR에 속하는 점 제외
- ④ PR에 속하는 IR들의 교점 중 다른 IR에 속하는 점 제외
- ⑤ PR의 모서리를 따라 1m간격의 점을 추출해 IR에 속하는 점 제외
- ⑥ PR에 속하는 IR의 모서리를 따라 1m 간격의 점을 추출해 다른 IR에 속하는 점 제외

(6) 배치 후보점 중에서 한 점의 선택

유한개의 배치 후보점 중에서 실제로 새로운 블록의 기준점을 위치시킬 한 개의 점을 선택한다. 그 방법은 여러 가지가 있을 수 있겠으나 다음과 같은 세 가지의 규칙을 우선 고려해 본다.

① 정반가장자리에 우선배치

정반 안쪽의 기존 블록들 사이에 새로운 블록을 배치했다고 했을 때 몇일 후 주위에 있던 블록들이 반출되었다면 다른 블록을 투입할 때 그 블록이 걸려서 투입이 곤란한 경우가 생길 수 있으므로 가능한 한 정반의 가장자리에 새로운 블록을 배치하는 것이 후속 블록의 투입을 용이하게 하는 결과를 가져올 수 있다. 또한 본 시스템의 제약조건 중 대형 블록을 정반 가장자리에 배치하여야 한다는 규칙을 만족시키기 위해서는 이 방

법을 사용해야 한다. 배치 후보점을 찾는 6개 방법 중 ①, ②의 3개 방법에 의한 후보점과 ③, ④의 3개 방법에서는 정반의 Dead Space를 표시하는 가상블록에 의해 생기는 IR만 고려해 후보점을 찾으면 이 규칙이 자연스럽게 충족된다.

② 좁은 위치에 우선배치

가능하면 기존블록들이 밀집된 위치에 블록을 배치하는 것이 다른 블록의 투입공간을 넓게 해주므로 유리한 것이다. 그러나 위의 가장자리 우선배치에서 설명했듯이 주위에 있던 블록들이 반출되었을 때 다른 블록의 투입에 장애물이 될 수도 있다는 점을 간과할 수는 없겠다. 또한 기존의 작업블록에 붙여 배치한 경우 좁은 위치 규칙은 어느 정도 자연스럽게 고려된다고 생각하고 본 시스템에서 이 원칙은 제외하기로 하였다.

③ 반출 후 넓은 공간 확보위치에 우선배치

새로 배치되는 블록이 반출되었을 때 그 주위의 공간이 넓어지면 후속 블록의 투입이 용이해질 것이므로 투입될 블록과 완료일이 같은 블록 옆에 들어갈 자리가 있다면 그 위치에 새로운 블록을 배치하는 것이 유리한 것이다. 이 규칙을 고려하기 위해 본 시스템에서는 배치 후보점을 찾을 때 배치대상블록과 완료일이 같은 기존 블록들에 의한 IR을 우선 고려한다.

3.2 자동배치용 2-Phase 알고리즘

위의 블록배치알고리즘을 적용해 내소형블록의 구분없이 계획을 수립하면 제2장의 제약조건 중 6번째 제약조건이 지켜지지 않는 현상이 많이 나타나게 된다. 따라서 본 시스템에서는 블록을 면적에 따라 대형과 소형으로 분류해 대형 블록들을 1차적으로 계획기간에 걸쳐 정반의 가장자리에 배치한 후 Phase 2에서 소형블록들을 빈자리에 배치하는 방법을 사용하였다.

Phase 1에서는 계획대상블록 중 일정 크기를 초과하는 블록들만 골라 배치 후보점을 찾는 6개 방법 중 ①, ②의 3개 방법에 의한 후보점에 배치하여 계획대상기간 전체의 계획을 작성한다. 그 이후에 정반 중심부와 기타 빈 자리에 블록들을 배치하는 Phase 2가 수행된다. Phase 2에서는 후보점을 찾는 6개 방법을 모두 사용하며 그 6개 방법을 사용하는 순서에 따라 정반 가장자리에 우선 배치하거나 반출시 넓은 공간확보원칙을 실현하게 된다.

4. 시스템 구현

본 시스템은 Microsoft Visual C++를 사용해 객체지향프로그램으로 개발되었으며 Windows NT, Windows 95 또는 Windows 98 등에서 작동된다. Windows용 GUI 프로그램을 통해 사용자의

편의성을 강조하였고 객체지향기법의 사용으로 사후 수정을 용이하도록 하였다.

시스템은 DB 또는 파일로부터 자료를 읽어 자동계획 또는 수동계획을 통해 블록의 조립일정을 작성한다. 각 계획은 다음에 소개하는 4개 화면에서 모두 가능하다. 자동계획은 시스템에 내재된 알고리즘을 통해 조립일정, 작업장소 등의 모든 결정변수를 정해준다. 수동계획은 각 화면의 특성에 따라 조금씩 다른 기능을 보유하고 있다.

주기의 용량이 32MB인 233MHz Pentium PC에서 1개월분 320개의 블록을 대상으로 자동계획 시스템을 가동한 결과 자동정반배치 알고리즘에 소요된 시간은 10분 내외였다. 실행시간 측면에서는 상당히 만족할만한 결과로 생각된다. 대형/소형 블록의 구분배치, P/S 블록의 동일정반 및 동시배치와 같은 제약조건을 고려하지 않고 자동계획 알고리즘을 수행하면 정반의 면적활용률이 67% 내외로 상당히 우수한 결과를 제공해주었다. 그러나 이들 제약조건을 고려한 알고리즘은 정반면적률 60% 정도 사용하는 계획안용 작성해 주었다. <그림 9>의 배치결과를 보면 블록들이 상당히 조밀하게 배치되어 있는 것으로 보이나 정반에 따라 56%에서 74% 사이의 면적활용률을 보여주며 전체 정반에 대한 평균면적활용률은 66% 정도이다.

시스템의 개발에는 자동계획뿐 아니라 수동계획에도 많은 노력을 기울였다. 특히 시스템 개발의 원래 목적은 실행계획용이 아닌 3~6개월 이후의 미래에 대한 중기일정계획용이었으나 개발과정에서 생산현장의 강력한 요구에 의해 실행계획용도가 추가되어 사용자에게 의한 자유로운 계획이 가능하도록 다양한 기능들을 추가하는 등 사용의 편의성을 크게 강조하였다.

4.1 배치상황 화면

<그림 9>의 배치상황 화면은 본 시스템의 기본화면으로서 지정일자자의 각 정반별 배치상황과 현재 보유하고 있는 계획대상블록들을 보여준다. 아울러 지정일자 전후에 대한 Subbay의 부하를 화면 중앙부의 표를 통해 사용자에게 제공해 준다.

계획대상블록을 적정위치에 원하는 형태로 배치하거나 기 배치블록의 계획을 취소하여 계획대상블록으로 환원시킬 수 있다. 또한 각 정반을 확대화면으로 볼 수 있는 기능을 갖추고 있다.

4.2 Gantt Chart 화면

지정된 작업장의 지정된 기간에 대한 블록별 일정은 <그림 10>과 같은 Gantt Chart로 볼 수 있다. 그 일정에 의한 부하변화를 수직축과 그래프를 사용해 보여주며 해당기간내에 그 정반에서 작업하는 모든 블록들을 그림으로 표현하고 있다. 마우스 작업에 의해 각 일정막대를 이동 및 확대/축소함으로써 블록의 일정을 조정할 수 있다.

4.3 Mosaic Chart 화면

블록의 일정(블록 조립의 착수 또는 완료일정)을 <그림 11>과 같은 표로 작성해 준다. 지정 기간에 대한 착수(또는 완료) 블록들을 일자별로 정리하여 프로젝트(계약된 배)별 또는 작업장(조립 Bay)별로 보여준다. 여기서도 마우스를 이용한 블록 일정의 조정이 가능하다.

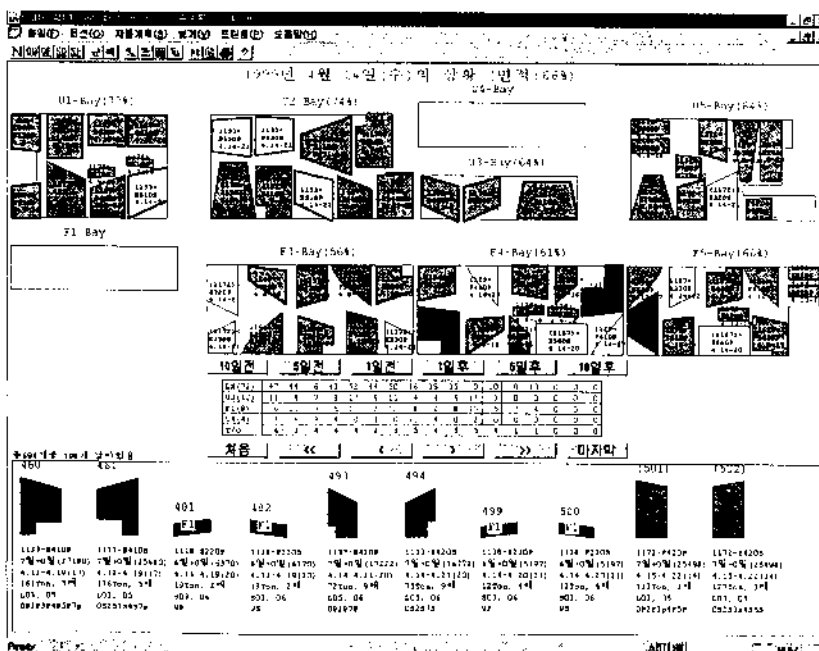


그림 9. UNITPLAN 시스템의 기본화면(배치상황 및 계획대상블록 보기).

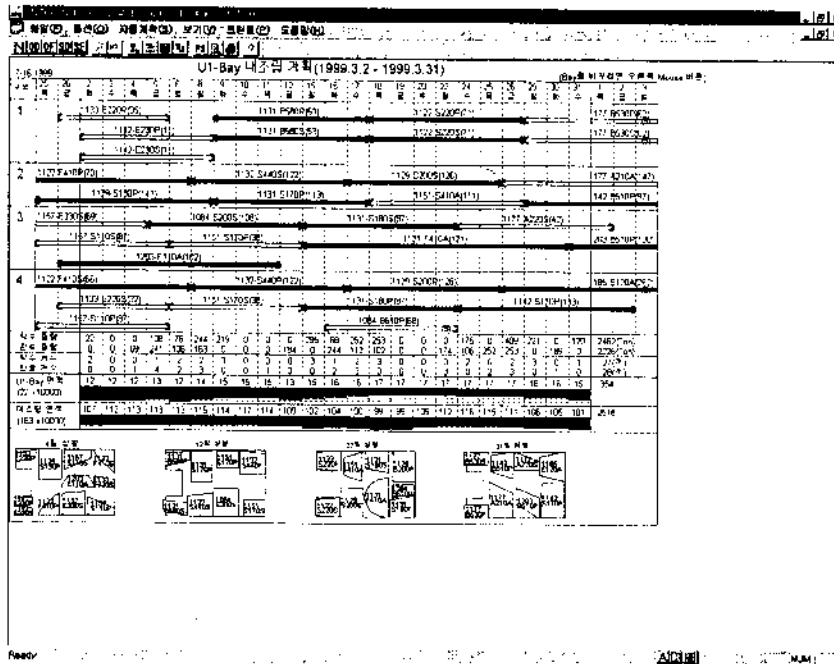


그림 10. Gantt Chart 형태의 화면.

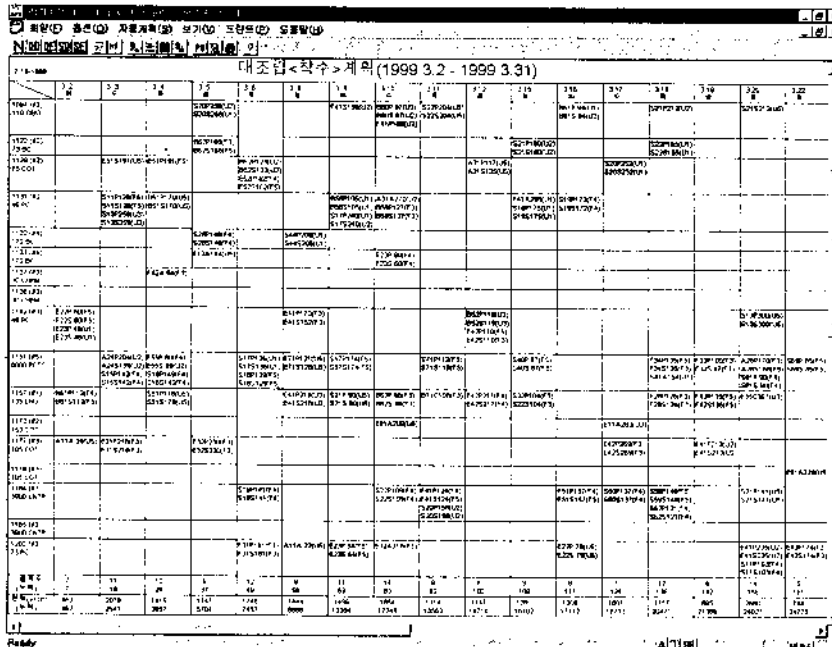


그림 11. Mosaic Chart 형태의 화면.

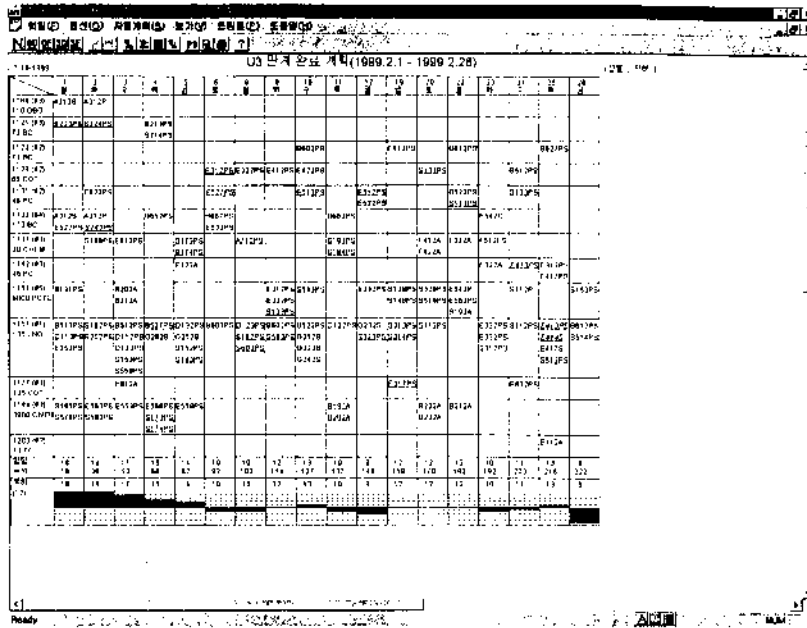


그림 12. Subbay 일정보기 화면.

4.4 Subbay Mosaic Chart 화면

Subbay의 일정과 부하상황을 <그림 12>와 같이 Mosaic Chart의 형태로 보여준다. 각각의 Subbay가 갖는 고유한 특성들을 고려하여 다양한 작업기능을 추가하였다. 여기서도 마우스를 이용한 블록일정의 조정이 가능하나 조립블록의 일정이 우선이므로 조립블록의 일정을 침해하는 조정은 불가능하게 되어 있다.

5. 결 론

본 논문에서는 곡블록공장 생산일정계획의 개관과 고정정반의 효율적인 사용을 위한 정반배치알고리즘을 소개하였다. 이 시스템은 현재 H중공업(주) 조선사업본부 내의 사업기획부서에서 중기 생산일정계획용으로 실제 사용되고 있으며 곡블록의 조립이 이루어지는 대조립공장에서는 1개월 실행 계획용으로 사용이 되고 있다. 이것은 내부의 알고리즘과 같은 학술적인 이슈보다는 현장의 사용편의성을 강조한 결과로 이해된다. 그러나 이러한 사용편의성의 강조로 인해 알고리즘의 개발에 많은 시간을 투입하지 못한 것이 사실이며 추후 이 부분에 대한 개선이 필요할 것으로 생각된다. 예를 들어 자동배

치의 경우 수행시간이 사용자에게 의해 특히 강조되어 좋은 해를 찾기 위한 충분한 탐색을 거치지 못한 결과 면적효율성이 다소 만족스럽지 못한 결과를 초래하였다. 따라서 소요시간 측면에서의 시스템은 상당히 만족할만 하였으나 정반면적활용률 측면에서는 앞으로 개선할 여지가 있을 것으로 생각된다.

참고문헌

고시근(1996), 유전 알고리즘을 이용한 블록조립공장의 생산일정계획, *경영과학* 13(1), 11-12.
 박경천 외 (1995), 고정정반 일정계획시스템의 개발, *산업공학*, 8(2), 95-103.
 박주철 외(1997), *선각일정계획 최적화시스템 개발 최종보고서*, 울산대학교 생산성 연구소.
 Lee, K. J., Lee, J. K. and Choi, S. Y. (1996), A spatial scheduling system and its application to shipbuilding: DASCURVE, *Expert System With Applications*, 10(3/4), 311-324.
 Lozano-Perez, P. (1983), Spatial planning: A configuration space approach, *IEEE Transactions on Computers*, C-32(2), 108-120.
 O'Rourke, J., Chien, C-B., Olson, T. and Naddor, D. (1982), A new linear algorithm for intersecting convex polygons, *Computer Graphics and Image Processing*, Vol.19, pp.384-391, 1982.
 Preparata, F. and Shamos, M. I. (1985), *Computational Geometry: An Introduction*, Springer-Verlag New York Inc.



고시근

1986년 고려대학교 산업공학과 학사
1988년 KAIST 산업공학과 석사
1993년 KAIST 산업공학과 박사
현재: 부경대학교 산업공학과 부교수
관심 분야: 생산 및 물류관리, 생산정보시스템



최용선

1986년 서울대학교 산업공학과 학사
1988년 KAIST 산업공학과 석사
1993년 KAIST 산업공학과 박사
1989년 Stanford Univ. 객원연구원
1991년 동경대학교 객원연구원
현재: 인제대학교 산업시스템공학과 조교수
관심 분야: 기업정보시스템, 전문가시스템,
전자상거래



박주철

1981년 서울대학교 산업공학과 학사
1983년 KAIST 산업공학과 석사
1990년 KAIST 산업공학과 박사
현재: 울산대학교 산업공학과 교수
관심 분야: 경제성 공학, 생산정보시스템



주철민

1988년 고려대학교 산업공학과 학사
1990년 KAIST 산업공학과 석사
1994년 KAIST 산업공학과 박사
현재: 동서대학교 정보시스템공학부 산업
공학전공 교수
관심 분야: 네트워크 최적화, 시뮬레이션 및
ERP 시스템