중일정계획을 고려한 선체 생산설계 작업할당 시뮬레이션

손명조¹ · 김태완^{1,2,†}

서울대학교 해양시스템공학연구소¹

서울대학교 조선해양공학과²

Job Assignment Simulation of Ship Hull Production Design in Consideration of Mid-Term Schedule

Myeong-Jo Son¹ · Tae-Wan Kim^{1,2,†}

Research Institute of Marine Systems Engineering, Seoul National University¹

Department of Naval Architecture and Ocean Engineering, Seoul National University²

This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this paper, we analyze the procedure of the design manager for the enhancement of the hull production design process by use of the simulation method. Normally, design manager assigns design jobs according to various methods and estimates the corresponding results. When the construction drawing which is the output of the detail design where a design is dealt by zones, the design manager identifies blocks and analyzes their work difficulties, and assigns jobs to design engineers who are different in capabilities. These processes including the design engineer who can be modeled with man-hours evaluation model are represented in detail as a simulation model. As the high-level modeling for the discrete-event system, we use Event Graph model. And we implemented the simulation using Simkit which is open simulation engine for the discrete-event system. We made the simulation scenario to be written by a user in the scenario generator which is separated from the simulation model, and made the simulation result to be visualized in the form of Gantt chart in a Web. In the scenario of the irregular issuance for various construction drawings which contain different numbers of blocks, we performed the Monte-Carlo simulation according to various assignment methods to find the assignment result that satisfies the mid-term schedule.

Keywords : Hull production design(선체 생산설계), Discrete-event simulation(이산사건 시뮬레이션), Event graph modeling(이벤트 그래프 모델링), Job assignment(작업할당), Mid-term schedule(중일정계획)

1. 서 론

1.1 조선 설계 분류와 생산설계의 특징

조선 설계업무는 선후관계, 설계의 완성도를 기준으로 크게 기본설계, 상세설계, 생산설계로 구분할 수 있다. 기능적으로는 선체 분야와 선실, 선장, 기장, 배관 등을 포함하는 의장 분야로 구분할 수 있다. 본 논문에서는 선체 분야의 생산설계를 대상으로 한다.

선체 기본설계는 선박 전체를 대상으로 하여, 종양단면도(midship section), 일반구조도(general construction)와 같은 기본도, 사양서(spec.), 설계 표준, 표준 부품 형상과 기호 등을 정의, 설계한다. 상세설계는 이를 기반으로, 구조적 또는 기능적으로 의미가 있는 각 구획에 대한 구조도를 설계, 확정하고, 예량을

산정한다. 생산설계는 구조도와 블록분할도(block division)에 근거하여 각 블록별로, 생산을 할 수 있도록 정교하고 구체적인 공작도를 작성하고, 네스팅을 통한 절단도를 작성한다. BOM(Bill of Material)관점에서 조선 설계 단계를 비교하면 Fig. 1과 같다. 앞서 언급한 생산설계는 조선전용 3차원 캐드(CAD; Computer-Aided Design)를 이용하여, 3차원 선체 블록 모델을 생성하고 2차원 뷰(view)를 불러오는 기능(Son, et al., 2013)을 활용하여, 공작도를 작성한다. 또한, 3차원 선체 블록 모델은 네스팅용 부재 모델로 변환되어 네스팅 및 절단도 작성에도 사용된다. 이외에도 네스팅 결과와 현장 자재 정보를 연계하는 소요강재표, 블록의 모든 부재의 재질, 무게, 위치, 수량, 위상관계(hierarchy)를 포함하는 M-BOM(Manufacturing BOM), 리프팅러그도면(lifting lug plan), 족장도(scaffolding plan)를 포함하는 생산기술도가 한 블록에 포함되어야 할 생산설계의 산출물이다.

Fig. 1 Feature of each design phase in the shipbuilding regarding BOM structure

상세설계의 단위인 구획은 일반적으로 여러 블록으로 구성되어 있다. 따라서, 상세설계 부서에서 각 구획별 구조도가 출도되면, 그 구획에 포함된 블록을 블록분할도를 근거로 분류하고, 그 블록별로 생산설계가 수행된다. 생산설계는 구조도 분석과 모든 부재 라벨링 (labeling), 모델링, 공작도 작성 등이 서로 선후관계에 종속적이기 때문에 한 명의 설계자가 특정 블록에 대한 생산설계를 모두 수행한다. 이러한 특정 블록에 대한 생산설계는 평균적으로 한 달 내외의 시간이 소요된다. 또한, 구조도는 서로 다른 개수의 블록을 포함하고 있고, 불규칙적인 시간 간격을 두고 완성되어 생산설계 부서에 도착하게 된다. 따라서, 한정된 설계 인원이 투입될 때, 작업할당을 어떻게 하느냐에 따라 총 소요시수와 프로젝트 종료일이 달라지게 된다. 이러한 문제를 해결하는 작업할당에 대해서 다음 절에서 살펴보도록 한다.

1.2 작업할당 문제

작업자와 할당할 작업의 수가 동일할 때 최적할당은 Hungarian 알고리즘으로 해결된다 (Kuhn, 1955). Hungarian 알고리즘은 Kuhn이 1955년에 개발, 발표한 개념으로, 할당문제를 다항식의 수준에서 해결할 수 있는 결합 최적화 알고리즘이다 (Burkard, et al., 2009).

관련 연구로는 간단한 비즈니스 프로세스에 대하여, Hungarian 알고리즘을 사용하여 작업할당을 하고, 시뮬레이션으로 검증한 연구가 있다 (Shen, et al., 2003; Kamrani, et al., 2009). 또한, 본 연구의 선행 연구로서 조선 선체 생산설계 작업할당 문제에 대하여 블록 난이도를 평가하는 블록 표준 설계시수 개념과 설계자 능력을 고려한 설계자 가용시수 개념을 적용하여 Hungarian 알고리즘으로 최적할당을 한 연구가 있다 (Son & Kim, 2012b).

선행 연구에서는 여섯 개의 블록을 포함하는 구조도가 출도되었을 경우, 네 명의 설계자에게 어떠한 블록을 할당을 해야 총 소요시수가 최소가 되는지를 Hungarian 알고리즘으로 문제를 해결하였다. 본 연구에서는 더욱 현실에 맞는 문제로 시나리오를 확

장하였다. 구체적으로, 서로 다른 개수의 블록을 포함하는 네 개의 구조도가 불규칙한 시간 간격을 두고 출도되는 동적인 상황에서 네 명의 설계자에게 작업을 할당하고, 각 설계자가 작업이 종료되면 새로운 블록을 할당하는 과정을 예측하고 분석하는 것을 목적으로 한다. 이러한 경우에 총 소요시수가 최소가 되거나 전체 설계 종료에 걸리는 시간이 최단 시간이 되도록 하면서 동시에 중립적 계획을 만족하는 작업할당을 위해 이산사건 시뮬레이션을 이용하여 설계 업무 할당 프로세스를 상세 모델링하고자 한다.

1.3 논문의 구조

2장에서는 이산사건 시뮬레이션을 상세하게 모델링할 수 있는 이벤트 그래프 모델링에 대해 설명하고, 이를 이용하여 선체 생산설계 작업할당 시뮬레이션 모델을 구성한다. 3장에서는 본 논문에서 대상으로 하는 작업할당 업무에 대한 시뮬레이션 시나리오에 대해서 설명한다. 4장에서는 공개용 이산사건 엔진인 Simkit을 이용하여 구현한 시뮬레이션의 구성에 대해서 설명하고, 5장에서는 시뮬레이션 결과를 분석한다. 마지막으로, 6장에서 결론 및 향후 연구에 대해서 설명한다.

2. 이벤트 그래프 모델링을 이용한 상세 이산사건 모델링

2.1 이벤트 그래프 모델링

이벤트 그래프 (event graph) 모델링은 이산사건 시뮬레이션의 사건 처리 (processing), 사건 계획 (scheduling) 등의 관계를 그래프를 이용하여, 가시적으로 표현하는 방법이다 (Schruben, 1983; Buss, 1996). 이러한 개념을 A.H. Buss가 Java로 구현하고, 이산사건 시뮬레이션 엔진인 Simkit을 공개용으로 배포하였다 (Buss, 2002). 이러한 Event Graph 모델링의 핵심 개념은 다음과 같다.

Fig. 2 좌측 상단에서는 이벤트 그래프 모델링에서 'Scheduling Edge'의 개념을 보이고 있다. 사건 A와 사건 B 사이에, 촉발 (triggering) 관계가 있을 경우, 이를 시간과 함께 표현하는 방법으로, 사건 A가 발생하면, 항상 사건 B가 발생할 경우 적용하는 방법이다. 이 때 'time delay'는 사건 A가 발생한 후, 얼마 이후에 사건 B가 발생하느냐는 개념으로, 즉시 사건이 발생하는 경우가 많기 때문에, 'zero'가 많이 사용된다. 또한 'time delay'는 상수가 될 수도 있고, 변수가 될 수도 있다.

위와 같이, 사건 B가 무조건 일어나는 경우보다는 어떠한 조건을 만족시켜야지만, 사건 B가 생성되는 경우가 더 많다. 이러한 조건 개념을 추가한 것이 'Scheduling Edge with Boolean Condition' 개념이다. 이때는, Fig. 2의 좌측 가운데와 같이,

Boolean 조건 'i'를 물결표시 위에 추가한 내용으로 표현한다.

복잡한 시뮬레이션을 수행하면, 사건들이 어느 특정 순간에 동시에 발생하는 경우가 있다. 이 때, 사건 간의 선후관계가 반드시 필요한 경우가 있다. 즉, 특정 사건에게 우선권 (priority)를 옵션으로 부여할 수 있고, 시간이 같을 경우는 우선권이 높은 사건을 먼저 스케줄링하는 개념이다. Event Graph 모델링 개념으로는 Fig. 2의 우측 상단과 같이 우선권이 있을 경우, 'scheduling edge' 위에 원안에 그 우선권을 ("HIGH", "HIGHER", "HIGHEST", "LOW", "LOWER", "LOWEST") 가운데 하나로 적어 주면 된다.

Fig. 2 Model representations of event graph modeling

사건이 발생하여 다른 사건을 촉발할 때는, 추가적인 메시지나 데이터 등을 전달해야 하는 경우가 많다. 즉, 같은 사건이라도 인스턴스 (instance)가 달라지는 개념이다. Fig. 2의 우측 가운데에서 볼 수 있는 것과 같이, 이벤트 그래프 모델링에서는 박스 안에 전달된 메시지 또는 데이터 변수명을 기입한다. 이 때는, 이를 수신할 사건에도 전달받을 메시지 형태 또는 프로그래밍 언어의 데이터 타입을 일치시켜야 한다.

시뮬레이션 모델이 복잡해질수록 모델간에 전달되어야 하는 메시지는 늘어난다. 이를 반영하여, 'Scheduling Edge with Multiple Arguments'가 Fig. 2의 좌측 하단과 같이 정의된다. 서로 다른 자료형의 인수 (argument)를 보내고 받는 인터페이스를 만들어서 전달하는 개념이다. 이 때, 인수의 개수와 그 자료형은 제한이 없다. 단, 이를 전달받는 사건에 데이터 타입, 순서, 개수를 정확히 일치시켜야 한다.

위와 같이 특정 사건들이 스케줄링되어 있는 상태에서 (즉, 'time delay'가 있는 상황), 어떠한 사건이 발생되었을 경우에 특

정 사건을 취소하는 경우도 있다. 이를 'Canceling Edge'라고 하고, Fig. 2의 우측 하단과 같이 점선으로 표시된 화살표로 표현한다. 또한, 'Canceling Edge'는 우선권과 'time delay'를 가지지 않는다. 하지만, 'Scheduling Edge'에서 인수가 사용되었다면 마찬가지로 인수를 적어주어야 하고, 불 (Boolean) 조건도 적용할 수 있다.

실제적인 간단한 예를 살펴보기로 한다. Fig. 3는 "For Loop" 구현을 보여주고 있다. 시뮬레이션이 시작되자마자 사건 'Run'이 호출된다. 사건 'Run'이 발생하면, 곧 바로 정수 '0'와 함께 'Init (0)' 이 발생한다. 이는 정해진 서버의 개수인 m만큼 반복 수행된다. 즉, 시뮬레이션 시작과 동시에 다른 ID를 갖는 m개의 서버가 생성된 것이다.

Fig. 3 "For Loop" modeling example in event graph

DEVS (Discrete Event System Specification) coupled 모델 개념처럼 (Zeigler, et al., 2000), 두 개의 단위 모델을 연계시켜 주는 개념도 이벤트 그래프 모델로 표현 가능하다. 이벤트 그래프 모델로 개발된 단위 모델을 컴포넌트 (component)라고 한다. Fig. 4와 같이 두 개의 컴포넌트가 있을 때, 발신자 (source) 역할을 수행하는 컴포넌트의 사건들을 가운데 수신자 (listener) 역할을 수행하는 컴포넌트에 동일한 이름과 형태 (signature)의 사건이 있을 경우, 발신자 컴포넌트에서 그 사건이 발생되면 수신자 컴포넌트의 동일 사건도 스케줄링되는 가능하다. 즉, 발신자 컴포넌트의 특정 사건을 수신자 컴포넌트 사건의 촉발 사건으로 사용이 가능하다. Fig. 4에서는 사건 B(k)가 촉발 사건에 해당된다.

Fig. 4 'Source-Listener' relationship between Event Graph components

Fig. 5 Overall event graph modeling of ship hull production design process simulation

2.2 선체 생산설계 프로세스의 이벤트 그래프 모델

선체 상세설계 프로세스를 간단하게 모델링하고, 상세설계의 출도 사건에 의해서 생산설계 프로세스가 실행되도록 모델링하였다. 생산설계 프로세스는 구조도를 분석하고 각 블록별 난이도를 평가하여 가용한 설계자에게 할당하는 설계 관리자 모델과, 할당받은 블록을 설계하는 설계자 모델로 구분하여 모델링하였다. 전체 모델 구성도는 Fig. 5와 같다. 시뮬레이션 시나리오를 사용자가 모델 외부에서 정의하면, 구조도 및 소속 블록 정보, 설계자 정보, 시뮬레이션 일반 설정이 각각 XML 파일로 생성되어 시뮬레이션을 실행할 때 각 모델로 전달된다. 이러한 시나리오 생성기를 이용하면 시뮬레이션 모델의 수정없이 반복 시뮬레이션이 가능하다. 또한, 구조도 접수, 작업할당, 설계 시작과 종료 등의 모든 정보는 시뮬레이션 조정자 모델 (simulation coordinator)에서 취합하고, 웹 가시화를 위한 간트 (Gantt) 차트 형태로 데이터를 가공한다.

Fig. 6은 선체 상세설계 모델을 나타내고 있다. 상세설계 모델은 소속 블록 정보를 포함하고 있는 구조도를 언제 출도할 지를 'IssueConstDWG (constDWG)'사건에 정의하고 있다. 이 사건은 생산설계 관리자 모델에 동일하게 정의되어 있고, 수신자 역할을 하도록 설계되어 있다. 구조도 정보는 이름과 소속 블록, 표준설계시수, 출도일을 사용자 정의 클래스에 저장하고 각 이벤트간에 인수로서 전달된다. 상세설계 모델에서는 사용자가 정의한 시나리오에 따라, 구조도의 개수만큼 정해진 출도시간에 'IssueConstDWG (constDWG)'사건이 스케줄링된다.

Fig. 7은 설계 관리자 모델의 이벤트 그래프 표현을 나타내고 있다. 시뮬레이션 시작과 동시에 시뮬레이션 엔진에 의해, 'Run'

Fig. 6 Event graph model representation of 'Hull Design Process'

사건이 실행되어서 초기화되고, 시나리오 생성기로부터 시뮬레이션에 참여하는 설계자 정보를 읽어 들인다. 대기하고 있는 도중, 상세설계 모델에서 'IssueConstDWG (constDWG)'사건이 발생되면, 즉시 'DWGAnalysis (blkNameMH, actInfo)' 사건을 스케줄링한다. 이 사건 스케줄링은 'HIGH' 우선권으로 이루어지고, 인수로는 상세설계 모델에서 넘어온 'BlockInfo' 클래스 형식의 인수인 'blockToBeModeled'중에서, 소속 블록 이름과 그 표준설계시수만을 맵 (map) 자료형으로 만든 값을 넘겨준다. 두 번째 인수로는 결과의 가시화와, 시뮬레이션 분석을 위한 모니터링 데이터를 사용자 정의 클래스 형태로 전달한다.

Fig. 7 Event Graph model representation of 'Design Manager Process'

또한, 'DWGAnalysis (blkNameMH, actInfo)'에서는 일정한 분석 시간인 $t_{analysis}$ 이후에 'Job Assignment (blkNames)' 사건을 발생시키는데, 이 때 $t_{analysis}$ 는 구조도 분석 시간을 모델링 한 것으로, 블록의 개수와 상관없이 구조도를 분석하는 기본 소요 시간 0.5 일로 가정하고, 블록의 개수에 따라서 각 블록의 표준 설계 시수 계산을 위한 입력값 파악을 위해 0.2 일씩 추가하도록 설정하였다. 이 이벤트에서 인수로 전달되는 'blockName'은 리스트 (list) 자료형으로, 현재 구조도 분석결과 식별된 모든 블록의 이름들을 나열한 것이다. 또한, 설계 관리자가 'actInfo'에 어떠한 구조도 분석을 언제부터 착수해서, 언제 할당을 하였는지에 대한 정보를 저장한다. 이 클래스 데이터는 시뮬레이션 조정자 모델이 취합한다.

'Job Assignment (blkNames)'에서는 가용한 설계자가 있는 한, 또한 할당되지 않은 블록이 존재하지 않는 한, 계속 즉시 반복 생성된다. 이 때는 'simSetting.xml'에 있는 작업할당 방법에 따라, 할당 방법을 달리하여 블록을 작업자에게 할당한다. 시뮬레이션이 시작되면 작업할당 방법을 소속 변수로 하는 'Design Manager Process' 모델을 생성하도록 구현하여서, 이처럼 모델의 수정없이 외부 시나리오에 따라 동작원리를 변경할 수 있다. Fig. 8은 설계자 모델의 이벤트 그래프 표현을 나타내고 있다. 관리자가 작업을 할당하면, 이를 할당받은 설계자와 블록 정보를 메시지로 하는 'Start Design (DEName, BlkMH)'사건을 통해 설계자 모델을 촉발하도록 정의하였다. 할당된 모델의 표준설계시수와 설계자의 가용시수를 이용하여 설계 종료시간인 $t_{expectedMH}$ 를 계산하여 'End Design (DEName, BlkMH)' 사건을 스케줄링한다. 그 결과, 설계가 완수되었기 때문에, 즉시 'Job Assignment (blknames)' 사건을 스케줄링하여, 이를 수신하고 있는 설계 관리자 모델을 촉발하도록 설계하였다.

Fig. 8 Event Graph model representation of 'Design Engineer Process'

3. 중일정을 고려한 선체 생산설계 작업할당 시뮬레이션 시나리오

Fig. 9는 본 논문에서 적용한 시뮬레이션 시나리오를 개략적으로 나타내고 있다. 'S02'호선의 생산설계는 설계 관리자 한 명과, 네 명의 설계자들이 참여한다. 상세설계 부서에서는 총 네 개의 구조도가 모두 다른 시간 간격을 두고, 연속적으로 출도된다. 각각의 구조도는 2 ~ 6개까지 서로 다른 개수의 블록들을 포함하고 있다. Fig. 9에서 볼 수 있는 바와 같이, 네 개의 구조도는 'S02-Main Deck Const.', 'S02-Stern Const.', 'S02-Bow Const.', 'S02-Engine Room Const.'이다. 출도 순서는 위의 표기 순이며, 각각의 도면은 시간 간격을 각각 0일, 30일, 20일, 11일의 간격으로 출도된다고 가정한다. 즉, 'S02-Main Deck Construction'은 시뮬레이션이 시작되자마자 출도가 되고, 'S02-Stern Construction'은 시뮬레이션 시간으로 30일 뒤에 출도가 된다. 각각의 구조도는 서로 다른 난이도의 블록들로 구성되어 있다. 블록 난이도 평가는 각 블록의 격벽의 개수, 체적, 경사격벽의 개수, 개구부 개수, 리프팅러그 설치 여부, 족장 설치

여부에 대한 가중치를 적용하여 계산되며, 한 블록을 숙련된 생산 설계자가 설계할 때의 소요되는 난이도를 9 M/H (Man-hours)를 1 M/D (Man-days)로 하는 시수로 나타낸다 (Son & Kim, 2012a). 'S02-Stern Construction'의 예를 들면, 'B#2701' (16.5 M/D), 'B#3701' (22.0 M/D), 'B#2702' (13.7 M/D), 'B#3702' (18.0 M/D)의 네 개의 블록으로 구성되어 있다. 즉, 가용한 설계 자원은 서로 다른 능력을 가진 네 명인데 반해, 설계가 완료되어야 할 블록은 총 16개의 블록이며, 각각의 난이도와 일의 양은 블록 표준 설계시수의 개념으로 살펴 보면, 8.5 M/D ~ 45.2 M/D까지 다양하다. 여기서, 설계자 가용시수는 근속년수, 블록 설계 실적, 최근 실적 5건의 오작, 재검도 여부 등을 이용하여, 100%를 숙련된 설계자의 능력로 하는 효율값으로 계산하였다 (Son & Kim, 2012a). 본 논문에서는 관리자의 최적 할당 목적으로 총 소요시수 최소화, 설계 종료 시간의 최단화를 설정하였다. 또한, 계산된 할당결과가 주어진 중일정계획을 만족하는지를 확인하였다.

선박은 탑재네트워크로부터 역방향으로 각 블록의 생산계획이 작성되고, 중일정계획의 형태로 작성되고 관리된다. 공정계획팀에서 중일정계획이 작성되어 생산설계부서에 공유가 되면, 관리자는 중일정계획을 우선적으로 만족시키는 도면출도계획을 세우게 된다. Fig. 9의 각 블록의 우측 박스에는 시뮬레이션 시작일을 기점으로 한, 각 블록의 생산착수일이 날짜단위로 표시되어 있다. 중일정계획 시나리오는 생산설비가 최대 4개 블록의 동시 생산을 지원한다고 가정하였다. 첫 네 블록의 생산착수일은 시뮬레이션 시작일로부터, 60일 후이며, 그 대상 블록은 첫 번째 출도 구조도면인 'Main Deck Const.'에 포함되어 있는 'B#2/3501', 'B#2/3502'이다. 생산은 블록의 절단부터, 소조립, 중조

립 등을 거쳐, 대조립에서 각 블록이 완성되는 프로세스형 업무이다. 따라서, 다음 네 블록 또한, 앞서 생산된 블록이 모두 생산이 종료된 시점이 아닌 각 블록의 생산 중·후반 부에 생산에 착수되게 된다. 본 시나리오에서는 각 블록의 생산소요일을 일괄 30일로 가정하였고, 25일 간격으로 네 블록씩 순차적으로 생산이 진행된다고 가정하였다.

4. 시뮬레이션 구현

2장에서 설명한 시뮬레이션 모델을 Java기반 공개용 이산사건 엔진인 Simkit 1.4를 이용하여, Groovy 스크립트 (Koenig, et al., 2007)로 작성하였다. 시뮬레이션 모델 가운데, 설계 관리자의 작업할당 전략을 모델로부터 분리시켜 외부에서 정의할 수 있도록 하였다. 이는 동일한 시나리오에서 할당 전략만 변경하여, 시뮬레이션을 수행하고, 결과를 비교하기 위해 시뮬레이션 전술 처리기 (Son, et al., 2010)의 개념을 도입한 결과이다. 이러한 할당 전략 외에도, 시뮬레이션의 시나리오 입력값인 구조도 정보, 소속 블록 정보, 구조도 출도일, 설계자 정보 등을 마찬가지로 시뮬레이션 외부에서 정의할 수 있도록 하였다. 또한, Monte-Carlo기반 시뮬레이션을 위한 시뮬레이션 반복 횟수와, 결과의 가시화 설정, 시뮬레이션 목적 설정 등과 같은 일반 설정 또한 시뮬레이션 외부의 시나리오 생성기에서 정의하도록 하였다. 이러한 정의 파일은 Groovy 스크립트 언어로 사용자가 쉽게 예시를 보며 값을 정의할 수 있고, 정의된 값은 XML 파일로 생성되어, 시뮬레이션 시작과 동시에 입력값으로 투입된다.

Fig. 9 Job assignment simulation scenario considering the mid-term schedule

시뮬레이션 결과는 Java기반의 공개용 간트 차트 가시화 형식인 jsGantt 1.2를 사용하여 구현하였다. 시뮬레이션의 모든 정보를 모니터링 데이터로 취합한 시뮬레이션 조정자 모델에서 jsGantt의 업무 (task)항목에 맞춰, 결과를 가공하고 HTML 형식에 맞춰 결과를 정리한다.

5. 시뮬레이션 결과

3장의 시나리오에 대하여, 설계관리자는 FIFO (First-In First Out), LIFO (Last-In First Out), 랜덤, 최대시수 우선할당, 최소시수 우선할당 방법에 따라 작업할당을 하도록 하였다. 이러한 할당 방법 가운데 대표적으로, FIFO 할당 전략에 따라 시뮬레이션을 진행한 결과가 Fig. 10에 정리되어 있다. 설계 관리자의 구조도 접수와 작업할당, 할당된 블록에 대해 각 설계자가 설계를 시작해서 종료된 시점이 간트 차트 형식으로 정리되어 있고, 새로운 구조도가 접수될 때마다 또는 설계자가 부여받은 블록 설계를 종료할 때마다 관리자가 FIFO 할당 전략에 따라 블록을 할당한 결과를 볼 수 있다. 또한, 블록의 표준설계시수와 사용자 가용시수에 따라 블록 설계가 각기 다르게 종료된 결과가 나타나 있다. 시뮬레이션의 주요한 결과 항목으로 총 소요시수와 주어진 설계가 종료된 시점을 Fig. 10 하단에서 확인할 수 있다. 또한, Fig. 10 우측에는 중일정계획에 따른 생산계획이 설계가 종료된 블록과 선으로 연결되어 나타나있어, 작업할당 결과가 중일정계획을 만족하는지를 가시적으로 확인할 수 있다.

FIFO 할당의 경우, 'B#2501'이 13.1일 설계지연 발생하여, 중일정계획을 만족하지 못하는 것으로 나왔다. 시뮬레이션의 주

요 목적으로 중일정계획을 만족시키는 값을 우선적으로 찾고 그 값이 여러 개 있을 경우, 그 가운데 총 소요시수가 최소가 되는 값을 찾는 할당 전략을 구현하고 이를 'mRANDO_1000'이라고 명명하였다. 기본적으로, 관리자가 작업할당 의사결정을 내릴 때마다 무작위적으로 (randomly) 할당을 하도록 한 시뮬레이션을 여러 번 반복하여 그 가운데 중일정계획을 만족하는 값을 찾고, 그 값이 여러 개 있을 경우 총 소요시수가 최소가 되는 값을 찾는 방법이다. 이 때, 시뮬레이션 반복은 1,000회 수행되었고, 약 3초 이내에 결과가 계산되었다. 시뮬레이션 반복 횟수는 같은 조건에서 횟수를 변경해가며 실험을 수행하여, 계산 시간이 적게 소요되면서도 수렴된 결과를 제시하는 값인 1,000회로 결정하였다 (Son, 2013).

Fig. 11에는 'mRANDO_1000' 할당 전략에 따라 작업할당이 이루어진 시뮬레이션 결과를 볼 수 있다. 모든 블록 설계가 중일정을 만족시켰고, 생산을 위한 준비기간이 최소인 블록의 여유 기간은 17.5일로 나타났다. 즉, 모든 블록이 중일정계획 대비 17.5일 이상의 생산준비기간을 확보할 수 있다. 또한, 총 소요시수도 278 M/D로 FIFO보다 개선된 결과를 얻을 수 있었다.

Fig. 12는 다양한 할당 전략에 따른 총 소요시수와 각 할당 결과의 중일정계획 만족 여부를 나타내고 있다. 총 소요시수 최소화를 위해 자체 개발한 Monte-Carlo기반 시뮬레이션 알고리즘인 'RANDO_1000'방법이 총 소요시수 최소값을 제시하고 있음을 알 수 있다. 그러나 기존의 할당 방법은 모두 중일정계획 제약조건을 만족하지 못하였고, 'mRANDO' 할당 전략만이 모든 블록의 생산설계 계획이 중일정계획을 만족하였음을 알 수 있다.

Fig. 10 Job assignment simulation result with the mid-term schedule scenario - FIFO

Fig. 11 Job assignment simulation result with the mid-term schedule scenario - mRANDO_1000

Fig. 12 Job assignment simulation results according to various assignment methods

6. 결론 및 향후 연구

본 연구에서는 블록단위로, 한 설계자에 의해 한 달 정도의 설계기간이 요구되는 생산설계 프로세스를 분석하였다. 이 가운데, 관리자의 작업할당이 앞선 할당의 결과에 따라, 또는 시간의 진행에 따라 새롭게 할당될 입력값이 투입되는 동적인 시나리오

에서는 기존의 정적인 상황에서 사용된 Hungarian 알고리즘으로 예측-해결하기 어렵다는 상황을 인지하고, 이산사건 시뮬레이션을 적용하였다. 여러 개의 블록 정보를 구성하고 있는 집합인 구조도와 설계자의 정보를 모델링하고, 생산설계 프로세스를 상세하게 모델링할 수 있는 상세 이산사건 모델링 방법인 이벤트 그래프 모델링에 대하여 설명하고, 이를 이용하여 생산설계 작업할당 시뮬레이션 모델을 작성한 결과를 제시하였다. 시뮬레이션은 공개용 이산사건 엔진인 Simkit을 이용하여 구현하였고, 다양한 시나리오는 시뮬레이션 모델과 분리하여 외부에 사용자가 정의할 수 있도록 하였다. 시뮬레이션은 다양한 작업할당 전략에 따라 결과를 예측할 수 있도록 하였고, 그 가운데 FIFO 방법과 중일정 계획을 만족하는 결과를 우선적으로 찾을 수 있는 'mRANDO_1000' 방법을 이용한 시뮬레이션 결과를 간트 차트형식으로 제시하고, 비교하였다. 본 연구에서는 문제 정의와 해결을 위해, 생산설계자와 구조도 및 블록의 수를 최소화하여 시뮬레이션을 수행하였다. 현업 적용을 위해서는 한 호선에 대한 실제 설계일정 계획 및 실적 데이터를 이용하여 시뮬레이션 모델을 보정하여야 한다.

향후 연구로는 설계자 총 소요시수 최소화, 시뮬레이션 시간 최단화, 중일정계획 만족 등의 간단한 시뮬레이션 목적에서 벗어나, 긴급출도블록 우선 배치 알고리즘, 제약이론에서의 DBR (Drum-Buffer-Rope) 방법을 이용한 재고를 고려한 작업할당과 같이 현업에서 의미가 있는 목적함수로 최적 시뮬레이션 결과를 제공하는 다양한 시나리오 기반 시뮬레이션을 수행하고자 한다.

감사의 글

본 연구는 제 1저자 손명조의 박사학위 논문의 일부를 발췌, 수정한 것이며, 도약연구과제 “조선 PLM의 BOM (Bills Of Material) 커널 개발에 관한 연구”의 지원으로 이루어진 연구 결과임을 밝히며, 이에 감사 드립니다.

References

- Burkard, R.E. Dell'Amico, M. & Martello, S., 2009. Assignment Problems. *SIAM*, Philadelphia (PA.).
- Buss, A.H., 1996. Modeling with event graphs. In *Proceedings of the 28th conference on Winter simulation/IEEE Computer Society: Coronado*, California, United States, 8–11 December 1996, pp.153–160.
- Buss, A.H., 2002. Component based simulation modeling with simkit. *Proceedings of the Winter Simulation Conference*, San Diego, California, United States, 8–11 December 2002, pp.243–249.
- Kamrani, F. Ayani, R. Moradi, F. & Holm, G., 2009. Estimating performance of a business process model. In *Proceedings of the 2009 Winter Simulation Conference (WSC)/IEEE*, Austin, Texas, United States, 13–16 December 2009, pp.2977–2988.
- Koenig, D. Glover, A. King, P. & Laforge, G., 2007. *Groovy in Action*. Manning Publications.
- Kuhn, H.W., 1955. The Hungarian Method for The Assignment Problem. *Naval Research Logistics Quarterly*, 2(1–2), pp.83–97.
- Schruben, L., 1983. Simulation Modeling with Event Graphs. *Communication of the ACM*, 26(11), pp. 957–963.
- Shen, M. Tzeng, G.H. & Liu, D.R., 2003. Multi-criteria task assignment in workflow management systems. *Proceedings of the 36th Annual Hawaii International Conference on System Sciences*, Hawaii, United States, 6–9 January 2003, pp.202–210.

- Son, M.J. et al., 2010. Modeling and Simulation of Target Motion Analysis for a Submarine Using a Script-based Tactics Manager. *Advances in Engineering Software*, 41(3), pp.506–516.
- Son, M.J., 2013. *Decision support system for job assignment in shipbuilding design using Discrete-event-based business process simulation*. Ph.D. Seoul National University.
- Son, M.J. Kang, H.W. & Kim, T.W., 2013. Scheme-based modeling for the reuse of the block model in AVEVA marine. *Proceedings of the Society of CAD/CAM Engineers Conference*, Pyeongchang, Korea, 30 January–1 February 2013, pp.450–456.
- Son, M.J. & Kim, T.W., 2012a. BPM-based job assignment in the ship hull production design. *Proceedings of the Society of CAD/CAM Engineers Conference*, Pyeongchang, Korea, 1–3 February 2012, pp.49–56.
- Son, M.J. & Kim, T.W., 2012b. Optimal job assignment for design manager in ship hull production design using BPM. *Proceedings of the Korean Institute of Industrial Engineers Conference*, Gyeongju, Korea, 10–11 May 2012, pp.846–853.
- Zeigler, B.P. Praehofer, H. & Kim, T.G., 2000. *Theory of Modeling and Simulation*. 2nd Ed. Academic Press.

손명조

김태완