

복수 개 · 보수 프로젝트의 자금제약하 프로그램 관리

- 자원제약 마스터-일정계획을 중심으로 -

Organizational Program Management of Multiple Maintenance Projects Under Fund Constraints

구 교 진*
Koo, Kyo-Jin

요 약

다수의 시설물을 보유한 기관에서는 유지보수와 리모델링(M/R) 프로젝트들이 동시에 진행되고 있다. 내부기능인력에 의해 프로젝트를 수행하는 자원제약 환경에서 프로젝트 관리자는 각 프로젝트의 완료일과 기관내 가용한 자원에 대한 제한을 고려하며 프로그램을 계획하고 관리하게 된다. 다중 자원제약 M/R 프로그램의 효과적 관리를 위해 이 논문은 장기적인 프로그램 계획축과 단기적인 일정창으로 구성되는 2단계 계층관리모형을 제안한다. 프로그램 플랜을 위한 연동계획기법은 연속되는 프로젝트 요청의 불확실성에 대처하기 위해 제안되었다. 일정창 속에서 마스터-일정계획을 수립하기 위해 개발된 PCR 일정계획 알고리즘은 M/R 프로그램에서의 작업반내 기능인력의 배치계획을 제공한다. 스프레드시트 인터페이스를 활용하여 M/R 프로그램의 일정계획 시스템을 개발하였고, 연속된 3개의 일정창으로 설계된 시뮬레이션 실험을 통해 개발된 일정계획 알고리즘과 기존의 자원제약 일정계획 휴리스틱의 성능을 비교하였다.

키워드 : 유지보수 · 리모델링, 복수 프로젝트, 기관 프로그램, 자금제약, 마스터 일정계획

1. 서론

시설물의 유지보수와 리모델링(Maintenance & Remodeling, 이하 M/R)은 시설의 사용가능성을 지속하기 위하여 성능을 회복시키고 변화된 요구에 맞게 개선하는 행위이다. 다수의 시설물을 보유한 기관(예를 들면, 대학, 군부대, 정부기관, 대기업) 등은 신규건축 뿐만 아니라 M/R 프로젝트를 발주하는 대형 발주자이다. 대형 발주기관에는 시설물의 신축 및 M/R을 관리하는 조직(예, 시설과)이 존재하고 있으며, 이 조직은 상시고용 내부직원과 더불어 외부인원을 계약에 의해 활용하게 된다. 대형 발주기관 중 미국의 위스콘신 대학과 같은 경우에는 330개가 넘는 교육시설과 운영시설을 소유하고 있으며, 10개의 공종별 작업반(예를 들면 목공반, 전기반)에 소속된 내부 기능인력에 의해 다수의 M/R 프로젝트를 수행하고 있다. 이 경우 각 공종별 작업반에 소속된 기능인력을 어떻게 각각의 프로젝트를

구성하는 액티비티에 배치하고 관리하여야 하는가를 해결해야 한다.

다수의 시설물을 소유한 기관에서 다중 상관관계를 갖는 M/R 프로젝트들을 효과적으로 계획하고 조정하기 위한 체계로서 Koo & Russell(2000)은 'M/R 프로그램 관리 모델'의 개념을 소개하였다. 이 관리모형을 실현하는 과정에서는 예산상의 제약을 받게 되고, 이러한 자금제약으로 인해 프로젝트들을 수행하는 비용의 규모 및 내용을 계획할 필요가 있게 된다. M/R 프로그램 환경에서는 각 프로젝트에 소요되는 재료비와 작업반에 소속된 기능인력의 인건비가 프로그램의 운영비용의 대부분을 차지하게 된다. 프로그램 관리자는 프로젝트의 긴급성 및 중요성을 고려하여 우선순위를 결정하고 그에 따른 예산을 확보하게 된다. 예산이 확보되어 앞으로 시공하여야 할 프로젝트들이 결정되면, 이미 진행되어온 프로젝트들과 새로이 프로그램의 관리영역 안으로 진입한 프로젝트들의 작업 액티비티와 이를 수행하는 내부 기능인력의 제한조건을 고려하여 프로그램 플랜을 확정하게 된다.

이 논문에서는 이러한 자금제약하의 M/R 프로그램 관리를 위

* 일반회원, 서울시립대학교 건축공학과 조교수, 공학박사
이 논문은 2002년도 서울시립대학교 학술연구조성비에 의하여 연구되었음

한 연구의 첫 번째 단계로서 외부적 불확실성으로부터 프로그램 플랜을 안정화하는 방법과 내부인력자원의 제약하에 다수 프로젝트들의 일정을 계획하는 알고리즘을 제시하고자 한다.

2. 프로그램 플랜의 안정화

M/R 관리부서가 다수 프로젝트를 동시에 관리 할 때, 프로그램 계획의 어려움은 M/R 환경의 동적인 특성에 의해 증가된다. 현실적으로 프로젝트에 관한 요청은 동일한 시점에서 발생하지 않고 시간의 흐름과 더불어 불규칙하게 프로그램 관리영역으로 진입하게 된다. 각 프로젝트의 시작점의 불확실성은 프로젝트의 주요일정을 예측하기 어렵게 하며, 프로그램 관리자는 자금조달의 계획, 내부 인력의 효율적인 운영 등을 위해 프로그램 계획을 안정화시킬 수 있는 방법을 필요하게 된다.

프로그램을 계획하는 방법으로 M/R 관리부서에서 일반적으로 사용하는 선착순(First-Come-First-Service : FCFS)방법의 알고리즘은 다음과 같다. 한 프로젝트가 요청될 때마다 (1) 프로그램 내에서 진행 중인 기존 프로젝트들에 배치되지 않은 기능인력이 요청된 프로젝트의 구성 액티비티들을 수행하는데 충분한가를 검토하고, (2-1) 기능인력에 여유가 있는 경우는 기존 프로젝트의 완료시점 뒤에 요청된 프로젝트가 위치하게 되지만, (2-2) 인력이 배치될 수 없으면 그 프로젝트의 연기되거나 연장된다. 결과적으로 FCFS 기법은 먼저 요청된 프로젝트를 우선적으로 계획하는 임시방편적 원칙으로 인해 장기적으로 운영되는 M/R 프로그램에 전체적인 최적해를 제공하지 못하게 된다. FCFS기법의 다른 결점으로는 프로그램 관리자에게 계속해서 재계획하는 부담을 주게 되며, 또한 발주기관이 기획하는 전체적 운영계획의 큰 틀과 상호작용하는 프로그램에서는 어느 정도의 예측가능성이 확보되어야 하는데 이의 부족으로 프로그램 및 발주기관 전체의 관리가능성을 손상시키게 된다는 점이다 (Church & Uzsoy 1992).

FCFS에 의한 계획과 비교하여 프로그램의 안정성을 향상하기 위해 이 연구에서는 연동계획(Rolling Horizon: RHZ)방법을 프로그램 계획에 적용하였다. RHZ의 계획체계는 M/R 환경의 동적인 불확실성에 대처하기위해 장기적 '계획축' (Planning Horizon)을 몇 개의 단기적 '계획창' (Scheduling Windows)으로 구분하고, 하위문제인 계획창을 계획·종합하여 '프로그램 플랜' (Program Master Plan)을 연속적으로 수립하게 된다.

즉, 첫 번째 일정창에 위치한 프로젝트들만을 대상으로 계획안을 도출한 후, 이 기간 내에 실행되지 못한 프로젝트 및 프로젝트의 작업 액티비티들을 다음 일정창에서 추가로 요청된 프로젝트와 연동하여 두 번째 계획안을 수립한다. 이러한 분할계획방법에 의해 FCFS보다 적은 빈도의 프로그램 계획으로 관리자가 보다 중요한 관리행위에 초점을 맞출 수가 있게 되며, 임시방편적 국부적 해결안이 아닌 예측가능성이 향상된 전체적 해결안을 얻을 수 있게 된다. 현실적으로 프로그램 관리자는 프로젝트의 요청과 시공 사이에 존재하는 중간과정(기획, 설계 등)의 기간을 조정하여 프로젝트들의 시공 개시점을 프로그램 계획축을 따라 일정한 간격으로 구획하여 대응하는 일정창에 배치하게 된다. 즉, 프로그램 관리자는 적극적인 계약정책을 수행함으로써 연속적인 M/R 요청의 동적 특성을 정적인 하위문제의 연속체로 전환하게 된다.

이 연구에서 대상으로 하는 대학의 M/R 프로그램에서는 공사 요청의 피크가 1년을 주기로 나타나기 때문에(예를 들면 여름방학과 명절기간 중 공사요청), 프로그램의 장기 계획축은 12개월로 설정하였다. 장기 계획축을 구성하는 단기 일정창의 기간이 길면 일정계획의 빈도를 줄일 수 있다. 그러나 일정창을 길게 하는 것은 창으로 구획된 기간에서 내부적 불확실성을 증가시키고 계획기간 내 완공 가능성을 감소시킨다. 일정창의 길이는 한달에 한번 건설 프로그램 계획을 수정하는 관례에 따라 1개월로 설정하였다(Kerzner 1994).

3. 일정창 속의 마스터-일정계획

3.1 계획 목표

프로그램 관리자는 특정한 일정창에서 다수 프로젝트의 일정을 계획할 때, 정해진 '품질'과 '예산' 범위 내에서 가능한 '빨리'라는 목표를 기준으로 한다. 이러한 목표는 '고객 만족'이라는 표현으로 압축될 수 있으며, M/R 프로그램 관리의 외부적 목표에 해당한다. 반면에 프로그램의 '생산성'과 기능인력의 '균형적 활용'이라는 내부 목표의 측면에서 기능인력의 작업대기시간은 감소되어야 하고(인력활용을 증가), 각 작업반의 인력활용율은 장기계획축의 연장선상에서 안정적으로 유지되어야 한다. 프로그램 내 재고관리의 측면에서 작은 '진행중 프로젝트수' (Projects-in-Progress : PIP)는 관리자원이 분산되지 않게 한다. PIP가 크게 되면 프로그램으로부터 완공되는 시설물의 수가 늘어나지 않으면서 프로그램으로 입력되는 자원의 양이 증가하게 되어, 기관 내에 머무르고 있는 미활용 자산의 규모가 증가하고 조직의 효율성은 감소하게 된다. 한편, M/R 프로그램의 동적인 특징은 각 작업반에 소속된 상시고용 인력의 적정규모를 결

1) 해당 프로젝트는 하도급계약에 의해 외부인력에 의해 수행될 수도 있지만, 내부의 상시인력이 고용되어 있는 상황에서는 외주는 프로그램에 추가비용을 부담시키게 되므로 프로젝트 수행이 긴급하게 요청되는 상황으로 한정된다.

정하기 어렵게 한다. 이 논문에서는 내부 기능인력의 보유규모에 대한 조정 가능성을 향후 연구과제로 남겨두고, 여러 작업반에 소속된 기능인력의 활용율 증대와 복수 프로젝트의 공사기간 단축에 초점을 맞추기로 한다. 결국 이러한 문제는 다중자원 제약 하 복수 프로젝트의 일정계획으로 귀결되게 된다.

3.2 자원제약하 일정계획

(1) 수학적 최적화 기법

자원이 제한된 상황에서 프로젝트의 일정계획에 대한 선형 연구들은 수학적 최적화기법과 휴리스틱 기법으로 대별된다²⁾. 수학적 기법은 일정계획 문제에 대한 유일한 최적해를 찾는 것을 목적으로 하며 선형계획법(Linear Programming), 정수계획법(Integer Programming), 동적계획법(Dynamic Programming), 부분열거법(Partial Enumeration)과 분기한계법(Branch and Bound) 등을 포함한다. 유감스럽게도 최적화기법에 의한 수학적 해결은 단지 소규모의 프로젝트에만 가능하며, 소규모 프로젝트에 대해서조차도 수학적 기법을 구현하는 것은 현실적으로 어려움이 있다. 예를 들면, 4가지 인력자원과 55개의 액티비티로 구성된 프로젝트의 계획문제를 선형계획법으로 모델링하기 위해서는 5,000개 이상의 등식과 1,600개 이상의 변수를 요구한다(Wiest 1964). 또 다른 수학적 모델링의 예는 4개의 프로젝트, 40개의 액티비티, 그리고 9개의 자원에 대한 Chen(1994)의 연구이다. 비록 Chen의 연구가 1057개의 0-1 결정변수와 1360개의 제약조건으로 구성되는 일반적 모델을 104개 변수와 53개 제약조건으로 단순화하였지만, 이러한 수학적 모델링 및 해법은 복수 M/R 프로젝트의 마스터-일정계획을 수립하는데 사용하기에는 현실적으로 거리가 있다고 할 것이다.

(2) 휴리스틱 기법

수학적 최적화 기법의 비현실성으로 인하여 휴리스틱 규칙들이 자원제약 프로젝트의 일정계획 분야에 활용되어져 왔다. 이 중 일부는 상업화된 프로젝트관리 소프트웨어에 채택되어 네트워크 공정표를 작성하는 기본원칙이 되고 있다. Hegazy & El-Zamzamy (1998)는 하나의 단순한 프로젝트 네트워크를 대상으로 상업적으로 판매되는 프로젝트 계획 소프트웨어들(예를 들면, Primavera Project Planner²⁾ (P3²⁾, Microsoft²⁾ Project)의 자원계획 기능을 비교했다. 이 연구에서는 휴리스틱

규칙의 구현방법이 소프트웨어에 따라 약간씩 상이하기는 하지만 결과물인 일정계획표는 최적안과는 상당한 거리가 있다고 결론지우고 있다. 또한 De Wit과 Herroelen(1990)은 대부분의 상업 패키지들의 계획능력을 “매우 초보적일 뿐만 아니라 심하게 왜곡되어 있다”라고 비판하면서(p. 116), 일정계획 소프트웨어의 대폭적인 성능향상을 요구하였다.

(3) M/R 프로그램 계획에 휴리스틱 적용

M/R 프로그램 환경에서 휴리스틱을 적용하기 위해서는 복수 프로젝트로 구성된 네트워크의 구조적 특성을 고려해야 한다. 그러나, 조사된 문헌의 대부분에서는 모든 프로젝트들이 동일한 시점에 프로그램에 진입하는 정적인(static) 환경을 가정하였다. 다수의 연구자들(예를 들면 Mohanty & Siddiq 1989, Meredith & Mantel 1995)은 복수의 프로젝트를 단일의 대형 프로젝트로 간주하고 CPM기반 휴리스틱 기법을 네트워크에 적용하였다. 그러나 더미 액티비티에 의해 인위적으로 묶여진 가상적 메가 프로젝트 모델은 여러 프로젝트들이 공통의 착공일과 완공일을 공유하지 않는 M/R 프로젝트의 개별성을 수용하는데 제한된 역량을 보이고 있다. 이러한 특성을 고려하지 않으면 일부 프로젝트 또는 그 프로젝트의 부분인 특정 액티비티는 일정계획의 매 순간마다 반복적으로 연기되어 해당 프로젝트의 순작업일수에 비해 완공하는데 소요되는 전체기간이 길어질 수도 있다. 이러한 현상은 M/R 프로그램의 전체적 관리목표를 달성하는데 저해요인으로 작용하게 된다.

4. PCR 일정계획 알고리즘

이상에서 검토한 M/R 프로그램의 특성과 기존의 계획방법에 대한 이해를 바탕으로 다음에서는 다중자원 제약하 복수 프로젝트의 관리를 위해 이 연구에서 제안하는 프로그램 제약자원(Program Constraint Resources, PCR)기반 일정계획 알고리즘을 소개하기로 한다. PCR 일정계획 알고리즘은 단위 일정창속에서 마스터-일정계획(Master Construction Schedule)을 수립하기 위해서 활용된다. 이 알고리즘은 기존의 일정계획 연구에서 주안점이 되었던 프로젝트와 액티비티 보다는 프로그램 내에 상주하는 자원, 즉 기능인력의 작업흐름에 관리의 초점을 맞추고 있다. 알고리즘의 6단계에 대한 설명을 쉽게 하기 위해 4개 M/R 프로젝트와 3개 작업반으로 구성된 간단한 프로그램 네트워크를 대상으로 마스터-일정계획을 수립하는 과정을 기술한다. 기술의 간결성을 위해 각 작업반의 명칭은 작업반 E(전기), M(기계), C(목공)로 표기하고, 각 작업반에는 한명의 기능인력이 있는 것으로 가정하였다.

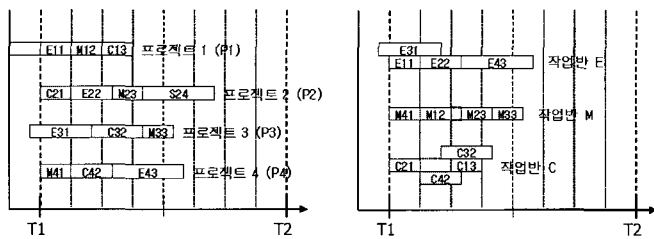
2) 최근 자원제약 일정계획에 적용되고 있는 인공지능기반 탐색기법(Genetic Algorithm, Simulated Annealing Algorithm)들은 임의탐색(Random Search)을 활용한 변종 휴리스틱 기법으로 분류될 수 있다.

4.1 초기 왼쪽정렬 배치

특정 일정창 속에서 마스터-일정계획을 수립하는 출발점으로 서 우선 각 프로젝트를 구성하는 작업 액티비티들과 이들 사이의 선후관계를 포함하는 초기배치 계획안을 준비한다³⁾. 일정창 T1T2(작업일 1~작업일 20)에 4개 프로젝트들을 왼쪽정렬한 초기배치를 [그림 1-(a)]에 나타냈다

단계 1 : 일정창에서 프로그램 네트워크를 준비한다. 각 프로젝트의 작업 액티비티를 확정하고, 각 액티비티의 예상 기간과 소요자원을 규정한다.

프로젝트들을 가장 빠른 개시시각(Earliest Start Time, EST)으로 배치한 초기안을 이 연구에서는 '왼쪽정렬 프로그램 간트 차트'로 이름 지었다. 이전 일정창에서 이미 계획되었으나 해당 일정창으로 이월된 작업이 있는 진행 중 프로젝트(P1 과 P3)를 수행되지 않은 액티비티(E11, M12, C13; E31, C32, M33)⁴⁾로 구성되는 새로운 프로젝트로 재정의 한다. 또한 액티비티가 이전 일정창에서 완료되지 않았다면, 그 액티비티의 가장 빠른 개시시각은 음의 값(예, P3에서 E31의 EST는 -1)이 된다.



(a) 프로그램 간트차트 (b) 자원수요차트
그림 1. 초기 왼쪽정렬 배치

4.2 최우선 PCR의 결정

단계 2 : 자원수요차트를 이용하여 프로그램 제약자원의 우선순위를 결정한다.

자원수요는 M/R 프로그램의 한 일정창에서 각 프로젝트를 구성하는 액티비티의 수행을 위해 공중별 작업반의 기술인력을 요구하는 정도를 의미한다. '자원수요차트'는 각 작업반의 기술인력에게 요구되는 시간에 따른 작업량의 수준변화를 도식적으로 보여준다([그림 1-(b)] 참고). 전체 프로그램의 작업성능은 요구 작업량에 비해 내부자원의 작업가능량의 비율이 가장 낮은 자원

- 3) 소규모 M/R 프로젝트가 수행되는 작업공간은 대부분 하나의 공간으로 구획되며, 통상 이 작업장에 한 작업반의 인원이 투입되면 다른 작업반은 작업을 수행하지 않는다. 결과적으로 프로젝트는 연속되는 작업들로 구성되는 선형성을 갖는 특성이 있게 된다.
- 4) 액티비티의 표기에서 알파벳 문자는 수행하는 작업반의 약자, 첫 번째 아래첨자는 프로젝트의 식별번호, 두 번째는 그 프로젝트에서 액티비티의 작업순서에 따른 식별번호이다.

의 흐름에 의해 결정된다(Newbold 1998). 프로그램의 내부자원은 각 작업반의 기능인력이 되며, 이 기능인력의 흐름이 해당 계획창 내 전체 프로젝트들의 수행을 제한하는 '프로그램 제약자원(Program Constraint Resources, PCR)'으로 작용한다. 이 중 전체 프로그램의 성능에 영향을 주는 순서를 확인하고, 작업반의 영향이 큰 순서로 자원활용율이 높도록 우선적으로 각 프로젝트에 배치하도록 한다. 이에 의해 각 프로젝트에서의 시간여유(Time Float)에 대한 고려뿐만 아니라 각 작업반의 '인력여유(Capacity Float)'를 고려한 마스터-일정계획을 수립할 수 있게 된다. 자원수요차트의 내용을 참고하여 각 PCR이 마스터-일정계획의 성능에 영향을 주는 정도(Criticality)는 다음 규칙에 따라 결정한다.

첫째, 각 작업반 내 인력흐름에서 작업대기시간이 없고, 현재 작업반 정원을 기준으로 자원수요에 대한 인력여유가 없다.

둘째, 인력여유가 있는 경우, 자원활용의 합계($\sum (\lambda \times d)$)가 가장 크다. 여기서 λ 는 각 작업일의 자원활용율을, d 는 작업일수를 나타낸다.

끝으로 해결되지 않은 우선순위는 작업 액티비티의 번호나 무작위 선택에 의해 결정한다.

이 규칙들에 기초하여 작업반 E는 일정창 T1T3에서 최우선 PCR이다. 최우선 PCR은 작업반 C이며, 그 다음 우선순위는 작업반 M이 된다. [그림 1]에 나타난 프로그램 사례의 경우에 최우선 PCR인 작업반 E의 기능인력에 의해 수행되는 액티비티 P11, P22, P31, P43은 다른 PCR에 의해 수행되는 액티비티에 비해 다음에 설명되는 자원배치 휴리스틱 규칙에 따라 가능한 빨리 마스터-일정계획 상에 배치된다.

4.3 휴리스틱 규칙의 적용

단계 3 : 다음의 휴리스틱 규칙을 순서대로 적용하여 특정 PCR의 작업흐름에서 액티비티들의 작업순서를 결정한다.

규칙 1 : 이른 액티비티 우선

규칙 2 : 짧은 후속체인 우선

후속체인(After-Chain)의 길이는 대상 액티비티의 작업기간을 포함하여 완공일까지의 기간이다. 짧은 후속체인의 규칙은 진행중-프로젝트수를 적게 한다는 마스터-일정계획의 작성목표에 기반한다.

규칙 3 : 짧은 작업기간 우선

남아 있는 다른 작업반들의 모든 액티비티가 계획될 때까지 위의 규칙들을 순서대로 반복 적용한다.

4.4 기술적 작업선행조건의 적용

단계 4 : 전체 PCR 흐름에서 액티비티들 사이의 기술적인 선

행조건을 적용한다.

일정창의 시작점(T1)에서부터 시작하여 먼저 최우선 PCR(E)의 액티비티를 단계 3에서 결정된 순서로 배치한다. 이후 차순위 PCR(C)과 그 다음 PCR(M)에 의해 수행되는 액티비티들이 배치되는데, 이 때 하위 PCR 액티비티는 소속된 프로젝트에서 상위 PCR의 작업흐름 중 기술적으로 선행작업이 되는 액티비티가 완료되는 시점이후까지 지연된다. 예를 들면 작업반 M의 작업흐름 속에 있는 액티비티 M12의 개시일은 프로젝트 P1에서의 선행 액티비티인 E11이 완료되는 시점의 다음 작업일까지 우측이동된다(그림 2 참조). 우선순위가 가장 낮은 PCR의 마지막 액티비티(M33)에 대한 기술적 작업선행관계에 대한 검토와 우측이동을 수행한 후에 최우선 PCR부터 추가적인 액티비티의 우측이동이 없을 때까지 이 절차를 반복한다. [그림 2-(a)]는 예시된 M/R 프로젝트 문제에 대하여 단계 4까지의 절차를 거쳐서 수립된 자원계획차트를 나타낸다. 작업반 E, C, M에서 자원흐름의 길이는 작업대기일을 포함하여 각각 15일, 13일, 그리고 15로 계획되었다.

단계 5 : 자원계획차트를 프로그램 간트차트로 전환한다 : [그림 2-(b)] 참조.

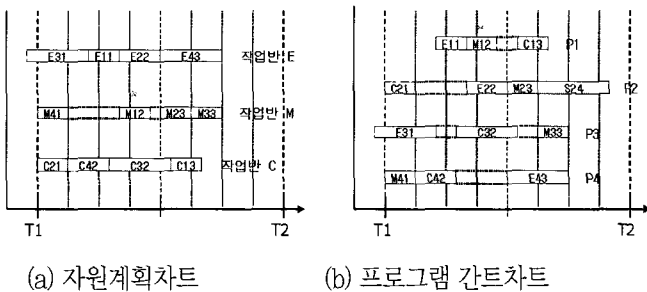


그림 2. 제약조건 적용 후 마스터 일정계획

4.5 계획창의 계획축으로 통합

단계 6 : 계획창에서 수립된 마스터-일정계획을 계획축 상에서 기존 계획창들의 내용과 통합한다.

최종적인 결과물로 PCR 알고리즘은 M/R 프로그램에 대한 공평별 작업반의 흐름관점(Flow View)인 자원계획차트와 프로젝트의 이벤트관점(Event View)인 프로그램 간트차트를 도출한다(그림 3 참조). 이 알고리즘의 기본원리는 조직내 PCR 흐름에 기반하여 액티비티를 실제적으로 계획하고 이것을 프로젝트 관점으로 전환하여 프로젝트의 진행을 관리하는 것이다.

5. 시뮬레이션 환경과 실험 계획

자원이 제약된 프로젝트들에 대한 기존의 3가지 일정계획 휴리스틱과 PCR 일정계획 알고리즘의 상대적인 성능을 검증하기 위해 시뮬레이션 실험을 수행하였다. 일반적으로 소규모 M/R 프로젝트에서는 작업공간을 여러 작업반이 동시에 공유하지 않으므로 대부분의 액티비티는 기존 CPM 분석의 관점에 따르면 주공정(Critical Path) 상에 놓이게 된다. 이러한 이유로 이 시뮬레이션 실험에서 시간여유(Time Float)에 기반한 휴리스틱들(예를 들면 Minimum Slack First, Minimum Project Slack First)은 비교되지 않았다. 실험에서 비교된 휴리스틱은 선행 연구들에서 효율성이 입증된 일정계획 규칙으로서(1) 가장 짧은 프로젝트의 가장 짧은 액티비티가 우선인 SASP(Shortest Activity from Shortest Project), (2) 가장 짧은 후속체인을 갖는 액티비티를 우선 배치하는 SAC(Shortest After-Chain), 그리고 (3) 가장 긴 후속체인을 갖는 액티비티를 우선 배치하는 LAC(Longest After-Chain)이다.

제안된 6단계 PCR 일정계획 알고리즘은 'Visual Basic for

TIME	Sched.Window A																				Sched.Window B																				Sched.Window C																																								
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60																					
E	A11	A11	A11	A11																	A20	B11	B11	B11	B11	B11	B11	B11	B11	B11	B11	B11	B11	B11	B11	B11	B11	B11	B11	C11	C11	C11	C11	C11	C11	C11	C11	C11	C11	C11	C11	C11	C11	C11	C11	C11	C11	C11	C11																						
M																					B21	B21	B21	B21	B21	B21	B21	B21	B21	B21	B21	B21	B21	B21	B21	B21	B21	B21	B21	B31	B31	B31	B31	B31	B31	B31	B31	B31	B31	B31	B31	B31	B31	B31	B31	B31	B31	B31	B31																						
C	A31	A31	A31	A31																	B41	B41	B41	B41	B41	B41	B41	B41	B41	B41	B41	B41	B41	B41	B41	B41	B41	B41	B41	B51	B51	B51	B51	B51	B51	B51	B51	B51	B51	B51	B51	B51	B51	B51	B51	B51	B51	B51	B51																						
S	A41	A41	A41	A41																	B61	B61	B61	B61	B61	B61	B61	B61	B61	B61	B61	B61	B61	B61	B61	B61	B61	B61	B61	B71	B71	B71	B71	B71	B71	B71	B71	B71	B71	B71	B71	B71	B71	B71	B71	B71	B71	B71	B71																						
P	A51	A51	A51	A51																	B81	B81	B81	B81	B81	B81	B81	B81	B81	B81	B81	B81	B81	B81	B81	B81	B81	B81	B81	B91	B91	B91	B91	B91	B91	B91	B91	B91	B91	B91	B91	B91	B91	B91	B91	B91	B91	B91	B91																						

(a) 자원수요차트

TIME	Sched.Window A																				Sched.Window B																				Sched.Window C																																									
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60																						
P1	A11	A11	A11	A11	A12	A12	A13	A13	A13												A21	A21	A21	A21	A21	A21	A21	A21	A21	A21	A21	A21	A21	A21	A21	A21	A21	A21	A21																																											
P2																					A31	A31	A31	A31	A31	A31	A31	A31	A31	A31	A31	A31	A31	A31	A31	A31	A31	A31	A31	A41	A41	A41	A41	A41	A41	A41	A41	A41	A41	A41	A41	A41	A41	A41	A41	A41	A41	A41	A41																							
P3																					A51	A51	A51	A51	A51	A51	A51	A51	A51	A51	A51	A51	A51	A51	A51	A51	A51	A51	A51	A61	A61	A61	A61	A61	A61	A61	A61	A61	A61	A61	A61	A61	A61	A61	A61	A61	A61	A61	A61																							
P4																					A71	A71	A71	A71	A71	A71	A71	A71	A71	A71	A71	A71	A71	A71	A71	A71	A71	A71	A71	A81	A81	A81	A81	A81	A81	A81	A81	A81	A81	A81	A81	A81	A81	A81	A81	A81	A81	A81	A81																							
P5																					B11	B11	B11	B11	B11	B11	B11	B11	B11	B11	B11	B11	B11	B11	B11	B11	B11	B11	B11	B21	B21	B21	B21	B21	B21	B21	B21	B21	B21	B21	B21	B21	B21	B21	B21	B21	B21	B21	B21																							
P6																					B31	B31	B31	B31	B31	B31	B31	B31	B31	B31	B31	B31	B31	B31	B31	B31	B31	B31	B31	B41	B41	B41	B41	B41	B41	B41	B41	B41	B41	B41	B41	B41	B41	B41	B41	B41	B41	B41	B41																							
P7																					B51	B51	B51	B51	B51	B51	B51	B51	B51	B51	B51	B51	B51	B51	B51	B51	B51	B51	B51	B61	B61	B61	B61	B61	B61	B61	B61	B61	B61	B61	B61	B61	B61	B61	B61	B61	B61	B61	B61																							
P8																					B71	B71	B71	B71	B71	B71	B71	B71	B71	B71	B71	B71	B71	B71	B71	B71	B71	B71	B71	B81	B81	B81	B81	B81	B81	B81	B81	B81	B81	B81	B81	B81	B81	B81	B81	B81	B81	B81	B81																							
P9																					B91	B91	B91	B91	B91	B91	B91	B91	B91	B91	B91	B91	B91	B91	B91	B91	B91	B91	B91	B101	B101	B101	B101	B101	B101	B101	B101	B101	B101	B101	B101	B101	B101	B101	B101	B101	B101	B101	B101																							
P10																					C11	C11	C11	C11	C11	C11	C11	C11	C11	C11	C11	C11	C11	C11	C11	C11	C11	C11	C11	C21	C21	C21	C21	C21	C21	C21	C21	C21	C21	C21	C21	C21	C21	C21	C21	C21	C21	C21	C21																							
P11																					C31	C31	C31	C31	C31	C31	C31	C31	C31	C31	C31	C31	C31	C31	C31	C31	C31	C31	C31	C41	C41	C41	C41	C41	C41	C41	C41	C41	C41	C41	C41	C41	C41	C41	C41	C41	C41	C41	C41																							
P12																					C51	C51	C51	C51	C51	C51	C51	C51	C51	C51	C51	C51	C51	C51	C51	C51	C51	C51	C51	C61	C61	C61	C61	C61	C61	C61	C61	C61	C61	C61	C61	C61	C61	C61	C61	C61	C61	C61	C61																							

(b) 프로그램 간트차트

그림 3. 3개 일정창의 마스터 일정계획 예

Application(VBA) 언어를 사용하여 Microsoft Excel 환경 하에서 'M/R 프로그램 계획 시스템'으로 구현되었다. 비교되는 세 가지 휴리스틱들도 Brook 알고리즘을 기초로 코딩되어 개발된 M/R 프로그램 계획 시스템에서 활용할 수 있도록 하였다. 원래 Brook 알고리즘은 단일 자원과 단일 프로젝트를 가정하였으므로 다중 자원과 복수 프로젝트로 구성되는 M/R 프로그램의 특성에 맞도록 알고리즘의 일부를 수정하였다.

M/R 관리기관의 전체 계획축에 걸친 동적인 특성, 즉 외부적 불확실성을 반영하기 위하여 연속한 3개의 계획창을 갖도록 시뮬레이션 실험을 설계하였다. 시뮬레이션은 서로 다른 40가지 상황으로 계획된다. 각 상황에서 프로젝트들의 집합체 3개가 각각 3개의 일정창에서 계획된다. 각 계획창에서 각 프로젝트에 대한 요청이 프로그램 안으로 진입하는 시점은 작업일 0~19 사이에서 균등분포를 갖으며 평균값은 10일이다. M/R 환경의 동적인 특성을 반영하기 위해 한 계획창에 배치되는 프로젝트들을 전체 프로젝트들을 유형화한 프로젝트 모델들의 집합체에서 임의로 선택한다. 이 집합체는 3개 액티비티에서 6개 액티비티를 갖는 평균 4.85개의 액티비티로 구성되고 작업반 C, M, E, S, P 중 3에서 5개 작업반의 기능인력에 의해 수행된다. 액티비티의 작업기간은 평균 2.81 작업일이며 2일에서 4일의 범위를 갖는다. 실험을 간소하게 하기 위해 각 액티비티를 수행하는데 필요한 기능인력은 작업반의 구분없이 1인으로 가정한다. 각 M/R 프로젝트의 순 공사기간은 작업일 기준 6일에서 20일의 범위를 가지며, 평균 13.64일을 나타냈다. 시뮬레이션 실험의 진행 중에 실험을 통제하기 위한 어떠한 작위적인 조정이 이루어지지 않았다.

6. 실험 결과 및 분석

연속한 3개 계획창에서 40가지 시뮬레이션 상황에 4개 일정 계획 휴리스틱을 적용한 160개 실험결과에 대해 (1) 완공기간/순작업기간 비율, (2) 작업반 가동율, (3) 일정창 초과 지연 등의 성능평가항목을 분석하였다.

6.1 완공기간/순작업기간 비율

완공기간/순작업기간 비율(Completion-time/Duration Ratio, CDR)은 한 프로젝트를 구성하는 액티비티들을 왼쪽 정렬한 초기계획의 순작업기간과 특정 휴리스틱을 적용하여 도출한 마스터-일정계획 안에서 대상 프로젝트의 총완성기간 사이 비율의 평균값이다. CDR을 보완하는 기준으로 완공기간과 순작업기간 차이의 평균(Difference between Completion-time and Duration, DCD)가 일정계획 휴리스틱의 성능을 비교하기

위해 사용되었다. CDR과 DCD는 다음과 같이 계산된다.

$$CDR = \frac{\sum \left(\frac{\text{완공기간}}{\text{순작업기간}} \right)}{N}, \quad (1)$$

$$DCD = \frac{\sum (\text{완공기간} - \text{순작업기간})}{N}, \quad (2)$$

N = 시뮬레이션 실행 수

작은 CDR 또는 DCD 값은 해당 휴리스틱 규칙이 프로젝트들을 조속하게 완료하는 마스터-일정계획을 산출한다는 것을 의미하는 것으로, 발주자의 대기시간과 진행중인 프로젝트 수를 감소한다. CDR과 DCD에 대한 분석을 요약하면, 개발된 PCR 휴리스틱에서 낮은 값을 보이고 있어 두 기준의 입장에서 성능이 가장 우수한 것을 알 수 있다. 반면에 SASP는 가장 높은 CDR과 DCD를 보이고 있으며, LAC는 계획창 3에서 SASP와 SAC 보다 우수한 결과를 보이는 반면 계획창 1에서는 반대의 결과를 나타내고 있다. LAC에 의한 일정계획은 계획창에 따라 가장 넓은 범위의 성능차이를 나타내어 CDR의 값은 2.72~1.93, DCD는 19.32~10.60의 분포를 보이고 있다(표 1 참조).

표 1. 완공기간/순작업기간 비율(CDR) 및 완공기간과 순작업기간 차이(DCD)

일정창	순작업기간	PCR		SASP		SAC		LAC	
		CDR	DCD	CDR	DCD	CDR	DCD	CDR	DCD
1	12.4	1.48	5.95	2.20	17.07	1.89	12.72	2.72	19.32
2	12.6	1.56	7.15	2.51	21.51	2.33	18.67	2.49	17.48
3	12.9	1.58	7.34	2.56	21.57	2.14	15.70	1.93	10.60
평균	12.6	1.54	6.82	2.42	20.05	2.12	15.70	2.38	15.80

한가지 특이한 결과는 LAC를 적용했을 때 나타나는 가장 큰 성능편차이다. LAC는 모든 액티비티들이 특정 계획창 안에서 배치된다면 가장 작은 평균 완성시간을 보인다. 그러나 프로젝트의 한 부분이 다음 일정계획창으로 미루어지면 그 프로젝트는 후속체인의 기간이 새로이 시작하는 프로젝트들보다 짧기 때문에 다음의 계획창에서 더 낮은 우선순위를 갖게 되고 연속해서 그 다음 창으로 지연되는 것으로 분석되었다.

6.2 작업반 가동율

작업반 가동 퍼센트(Shop Utilization Percentage, SUP)는 특정 작업반의 총작업능력에 대한 각 시뮬레이션 실험에서의 계획 작업량의 비율(%)을 평균한 값이다. 작업반 가동비율(Shop Utilization Ratio, SUR)은 각 휴리스틱을 적용한 일정계획에서 각 작업반의 SUP 값을 제안된 PCR 알고리즘의

SUP로 나눈 값이다.

$$SUP = \frac{\sum \left(\frac{\text{계획된 작업량}}{\text{총 작업능력}} \right)}{N} \quad (3)$$

N = 시뮬레이션 실행 수

계획된 작업량 = 각 작업일에서 필요한 인력수 × 3개 일정창에서 계획된 작업일수

작업반 총작업능력 = 전체 인력수 × 3개 일정창 전체기간

$$SUR = \frac{SUP_{\text{비교휴리스틱}}}{SUP_{\text{PCR알고리즘}}} \quad (4)$$

시뮬레이션 실험의 결과로부터 PCR에 의한 작업반 가동율은 LAC에 의한 작업반 M의 가동율을 제외하고 모든 작업반에서 가장 높은 것으로 분석되었다. 비교대상이 된 4개의 휴리스틱 규칙들은 두 그룹으로 분류되어질 수 있다 : PCR과 LAC, 그리고 SASP와 SAC. 첫 번째 그룹은 두 번째 그룹에 비해 SUP와 SUR의 측면에서 더 높은 성능을 보이고 있다. SUR의 평균값에 있어 LAC는 0.970으로 PCR과 유사한 작업반 가동율을 나타내고, SASP와 SAC는 각각 0.688과 0.746으로 낮은 작업반 가동율을 보이고 있다([표 2] 참조).

표 2. 작업반 가동퍼센트(SUP) 및 작업반 가동율(SUR)

작업반	PCR		SASP		SAC		LAC	
	SUP	SUR	SUP	SUR	SUP	SUR	SUP	SUR
E	62.38	1.00	43.21	0.692	47.54	0.762	60.88	0.975
M	37.33	1.00	24.38	0.643	26.21	0.693	37.88	1.021
C	50.21	1.00	33.38	0.667	38.00	0.763	43.92	0.884
S	63.17	1.00	44.71	0.711	47.42	0.755	61.17	0.972
P	37.33	1.00	26.83	0.725	28.13	0.760	37.21	0.999
평균	50.08	1.00	34.50	0.688	37.46	0.746	48.21	0.970

6.3 일정창 초과 지연

일정창 초과 지연(Tardiness Over Window, TOW)은 (1) 세 번째 일정창의 종료시점에서 완공되지 못한 프로젝트들의 수(NOP), (2) 미완공 프로젝트를 완료하기 위해 최소한으로 요구되는 순 작업일수(NWD), 그리고 (3) 프로젝트 계획 완공일과 마지막 일정창의 종료시점 사이의 작업일수(WCE)의 기준에서 비교하였다. 제안된 PCR은 시뮬레이션 실험에서 비교된 4가지 일정계획 휴리스틱 중에서 위의 3개 평가기준에서 가장 우수하였다([표 3] 참조). 비교된 3가지 휴리스틱은 유사한 NOP를 나타내지만 NWD와 WCE의 값에서는 큰 차이를 보이고 있다. 특히 SASP는 비교된 4가지 휴리스틱 중 가장 낮은 성능을 나타내는 것으로 분석되었다.

표 3. 일정창 초과 지연(TOW)

TOW	PCR	SASP	SAC	LAC
NOP	1.88	4.68	4.03	4.85
NWD	7.65	53.53	44.98	13.15
WCE	10.43	101.60	37.43	32.45

7. 결론

다수의 시설물을 보유한 기관에서 유지보수와 리모델링(M/R) 프로그램을 관리할 때는 프로젝트 발주자의 만족이라는 외부목표와 함께 관리조직의 생산성과 관리가능성의 증대라는 내부적 관리목표를 갖게 된다. 이 연구에서는 M/R 프로그램 관리자가 다중 자원이 제약되는 상황에서 연속되는 프로젝트 요청을 관리·계획할 수 있도록 장기적인 프로그램 계획측과 단기적인 일정창으로 구성되는 2단계 계층관리 모델을 제시하였다. 연동계획기법에 의한 프로그램 마스터플랜은 동적인 M/R환경의 외부 불확실성에 대처하여 프로그램의 예측가능성과 관리가능성을 향상시켰다.

단위 일정창의 마스터 일정계획을 수립하기 위해 개발된 PCR 알고리즘은 프로그램 제약자원인 공종별 작업반에 소속된 기능인력의 흐름관리에 기초하였다. 3개의 연속된 일정창에서의 시뮬레이션 실험을 통하여 제안한 PCR 일정계획 알고리즘이 자원이 제약된 프로젝트에 대한 기존의 일정계획 휴리스틱들(SASP, SAC, LAC)에 비해 세가지 평가기준(CDR, SUR, TOW)에서 우수한 것으로 분석되었다. 다중 자원제약 복수 프로젝트에 대한 프로그램 관리를 위하여 개발한 전산시스템은 제안된 PCR 일정계획 알고리즘과 성능비교된 3가지 휴리스틱 규칙에 의해 일정계획을 수립할 수 있을 뿐만 아니라 다른 알고리즘에 의해 일정계획을 세울 수 있는 환경을 제공하고 있다.

이 논문은 자금제약을 반영한 M/R 프로그램 관리모델을 개발하기 위한 연구의 첫 번째 단계로서 외부적 불확정성으로부터 프로그램 플랜을 안정화하는 방법과 내부 인력자원의 제약하에서 다수 프로젝트의 일정을 계획하는 알고리즘을 제시하였다. 위의 연구내용은 M/R 프로그램으로 요청된 프로젝트들의 긴급성 및 중요성이 동일하다는 가정을 바탕으로 하고 있다. 그러나 동일한 우선순위에 의한 프로그램 계획안은 각 프로젝트의 우선순위의 조정과 예산배정을 위한 재입력자료로 활용될 수 있으며, 조정된 우선순위에 의한 프로그램 계획안을 개발된 전산시스템을 통해 확인할 수 있을 것이다. 이러한 계획 결과에 대한 우선순위 조정과정에서의 재입력 프로세스와 더불어 작업반 기능인력의 정원조정 그리고 외부발주의 대상결정 등의 문제가 보

다 충실한 M/R 프로그램 관리를 위하여 후속적으로 연구되어야 할 것이다.

참고문헌

1. Chen, V. Y. X. (1994). "A 0-1 Goal Programming Model for Scheduling Multiple Maintenance Projects at a Copper Mine." *European Journal of Operational Research*, 76(1), 176-191.
2. Church, L. K., and Uzsoy, R. (1992). "Analysis of Periodic and Event-driven Rescheduling Policies in Dynamic Shops." *International Journal of Computer Integrated Manufacturing*, 5(3), 153-163.
3. De Wit, J., and Herroelen, W. (1990). "An Evaluation of Micro-Computer-Based Software Packages for Project Management." *European Journal of Operational Research*, 49(1), 102-139.
4. Hegazy, T., and El-Zamzamy, H. (1998). "Project Management Software That Meets the Challenge." *Cost Engineering*, 40(5), 25-33.
5. Kerzner, H. (1994). *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*, 5th ed., Van Nostrand Reinhold, New York, NY.
6. Koo, K. J., and Russell, J. S. (2000). "Program Management Model for Multiple Building Maintenance Projects." *Proceedings of the Eighth International Conference (ICCCBE-VIII) on Computing in Civil and Building Engineering*, Fruchter, R. et al., ed., ASCE, 1220-1227.
7. Meredith, J. R., and Mantel S. J. (1995). *Project Management: A Managerial Approach*. John Wiley & Sons, New York, NY.
8. Mohanty, R. P., and Siddiq, M. K. (1989). "Multiple Projects Multiple Resources-Constrained Scheduling: Some Studies." *International Journal of Production Research*, 27(2), 261-280.
9. Newbold, R. C. (1998). *Project Management in the Fast Lane: Applying the Theory of Constraints*. St. Lucie Press, Boca Raton, FL.
10. Wiest, J. D. (1964). "Some Priorities of Schedules for Large Projects with Limited Resources." *Operational Research*, 12, 395-418.

Abstract

In a large owner organization, a program manager of multiple maintenance and remodeling projects has experienced increasing scale and complexity of coordinating the M/R projects with in-house technicians who belong to multiple trade shops. This paper proposes a dual-level hierarchical planning strategy that consists of a program master plan in the long-term horizon and a master construction schedule in an operational scheduling window. A rolling horizon approach to the program master plan is proposed to deal with the external uncertainty of unknown stream of project requests. A resource-constrained scheduling algorithm is developed to generate the master construction schedule in a scheduling window. During development of the algorithm, more emphasis is placed on long-term organizational resource continuity, especially flow management of program constraint resources, than ephemeral events of an individual activity and project. Monte Carlo simulation experiments of three scheduling windows are used to evaluate the relative performance of the proposed scheduling algorithm against three popular scheduling heuristics for resource-constrained multiple projects.

Keywords : maintenance & remodeling, organizational program, multiple projects, fund constraints, master construction schedule, simulation experiment