

패키징 인쇄를 위한 병렬 오프셋 인쇄 공정의 스케줄링

문재경 · 태현철^{1*}

한국생산기술연구원 디지털헬스케어연구부

Scheduling of Parallel Offset Printing Process for Packaging Printing

Jaekyeong Moon and Hyunchul Tae*

Department of Digital Healthcare Research, Korea Institute of Industrial Technology

Abstract With the growth of the packaging industry, demand on the packaging printing comes in various forms. Customers' orders are diversifying and the standards for quality are increasing. Offset printing is mainly used in the packaging printing since it is easy to print in large quantities. However, productivity of the offset printing decreases when printing various order. This is because it takes time to change colors for each printing unit. Therefore, scheduling that minimizes the color replacement time and shortens the overall makespan is required. By the existing manual method based on workers' experience or intuition, scheduling results may vary for workers and this uncertainty increase the production cost. In this study, we propose an automated scheduling method of parallel offset printing process for packaging printing. We decompose the original problem into assigning and sequencing orders, and ink arrangement for printing problems. Vehicle routing problem and assignment problem are applied to each part. Mixed integer programming is used to model the problem mathematically. But it needs a lot of computational time to solve as the size of the problem grows. So guided local search algorithm is used to solve the problem. Through actual data experiments, we reviewed our method's applicability and role in the field.

Keywords Packaging printing, Offset printing, Scheduling, Assignment problem, Vehicle routing problem

서 론

포장(Packaging)이란 '물품의 수송, 보관, 취급, 사용 등에 있어서 그것의 가치 및 상태를 보호하기 위하여 적절한 재료, 용기 등을 물품에 부여하는 기술 또는 그 상태'를 말한다¹⁾. 포장은 제품을 생산하는 제조업과 제품을 유통하고 부가가치를 창출하는 서비스업을 연결하는 매개체의 역할을 수행하고, 때문에 이를 담당하는 패키징 산업은 2.5차 산업으로도 불린다. 기술 및 소재의 기능이 미흡했던 과거에는 주안점이 제품을 보호하는 기본적인 기능에 있었다. 그러나 현대 사회에서 기술의 발전과 정보의 다양화로 인해 제품의 기능만으로 경쟁적 우위를 점하기 어려워졌고, 기업들은 제품 차별화를 위한 마케팅 수단으로 포장의 디

자인적 기능에 집중하고 있다²⁾. 예를 들면, 치약이나 핸드워시 등과 같은 저관여 일용소비재 상품들(Fast Moving Consumer Goods: FMCG)이 브랜드 이미지를 프리미엄화하기 위해 선택하는 고급스러운 느낌의 패키지들이 그것이다.

세계적으로 패키징 시장은 꾸준히 성장해 온 분야이다. 특히, 코로나 팬데믹은 일용소비재와 의료용품에 대한 수요를 증가시켜 패키징 시장의 성장을 가속시켰다. 디지털 플랫폼에서 소비자가 선택 가능한 맞춤형 패키징의 등장도 시장 성장을 뒷받침하였다. 이러한 성장세는 앞으로도 지속될 것으로 예상된다. 이미 예전부터 글로벌 물류기업들은 신규 고객층 확보를 위해 패키징 혁신을 시도해 왔으며, 최근에는 친환경 소비 트렌드에 맞춰 친환경 패키징을 위한 소재와 기술 개발에 주목하고 있다. 국내의 경우, 1인 가구 및 맞벌이 비율 증가, 고령화와 같은 사회 변화에 따라 소형 포장, 편의성 증대 포장, 약제 포장, 특수 포장 등의 다양한 형태의 포장 수요가 지속적으로 증가하고 있다³⁾.

이러한 패키징 산업의 성장에 맞춰 포장재를 인쇄하는 인쇄 산업도 급속한 환경 변화를 맞고 있다. 기존의 소품종

*Corresponding Author: Hyunchul Tae
Department of Digital Healthcare Research of Korea Institute of Industrial Technology
Tel: +82-41-589-8455, Fax: +82-41-589-8480
E-mail: sage@kitech.re.kr

대량생산 위주의 패키징 인쇄 시장은 색상 표현에 있어 다양화가 빠르게 진행되고 있고, 색에 대한 이해도가 높아지면서 인쇄 품질 관리 기준 또한 높아지고 있다. 그러나 국내 인쇄 중소기업은 품질 관리 측면에서 기존의 수기 방식을 답습하고 있다. 의사결정을 위한 데이터 수집부터 결과에 대한 피드백까지 작업자에 의존하므로 전반적인 의사결정에 작업자의 경험×직관×능력에 따른 불확실성이 존재하는 상황이다. 불확실한 의사결정은 생산성과 품질 저하를 초래하기에 기존 방식을 개선할 필요성이 있다.

패키징 인쇄는 생활용품, 화장품 등을 포장하는 종이 박스 포장재를 인쇄하는 기술로, 국내 패키징 인쇄의 대부분은 오프셋 인쇄 방식을 이용하고 있다. 오프셋 인쇄는 알루미늄 인쇄판 위에서 물과 기름(잉크)이 서로 섞이지 않는 성질을 이용한 평판 인쇄 방식이다. 평판 인쇄 방식에서 인쇄판은 잉크가 묻는 부분과 묻지 않는 부분을 나누기 위해 인쇄판에 특수 도료가 칠해지는데, 이 인쇄판이 거친 종이에 직접 닿게 되면 칠해진 특수 도료가 쉽게 벗겨지는 문제가 있다. 이에 오프셋 인쇄는 이름처럼 인쇄판이 종이와의 직접적인 접촉 없이 잉크를 전달할 수 있도록 고무 롤러를 사용한다. 인쇄판이 회전하면서 칠해진 잉크를 고무 롤러에 전달하면 종이가 이 고무 롤러 사이를 통과하면서 종이에 잉크가 인쇄되는 것이다. 표면이 상대적으로 매끄러운 고무 롤러를 통해 잉크를 전달하기 때문에 인쇄판의 손상이 적어 대량 인쇄가 가능하고 고무의 탄성으로 더 좋은 인쇄 품질을 얻을 수 있는 것이 오프셋 인쇄의 특징이다. 상업적으로 대량의 컬러 인쇄가 필요한 패키징 인쇄 분야에서 전통적인 오프셋 인쇄 방식은 그 장점이 명확하기에, 디지털 인쇄 방식이 많이 발달한 현재 그리고 앞으로도 국내 및 글로벌 시장의 요구에 대응하기 위해서 두 방식의 혼용 운용은 계속될 것으로 예상된다.

패키징 인쇄는 인쇄에 앞서 생산 시안 디자인 공정과 조색 관리 공정을 거친다. 특히, 조색 관리 공정은 인쇄에 필요한 잉크를 만드는 공정으로 작업자로 인한 불확실성이 존재한다. 이에 Tae⁴⁾는 조색 관리 공정에서 잉크 배합 비율과 소모량 계산 자동화의 필요성을 제시하였다. Hyun, et al.⁵⁾은 인쇄 환경에 영향을 받는 오프셋 인쇄의 색상 재현에 있어 Kubelka-Munk 모델의 적용성을 검증하였고, Seo, et al.⁶⁾과 Kim, et al.⁷⁾은 각각 스테킹 양상불 기법과 DNN (Deep Neural Network)을 이용하여 효율적인 별색 제조 방법을 제안하였다. 또한, Tae⁸⁾는 조색에 소모되는 잉크량을 예측하는 머신 러닝(machine learning) 기반 알고리즘을 개발하였다.

한편, 주문들은 CMYK에서 별색까지 다양한 색상을 활용하고 각각 필요로 하는 색상의 가짓수가 달라 그 조합이 매우 많은 반면, 라인에서는 동시에 여러 주문을 인쇄할 수 없기 때문에 한 라인에서 여러 주문을 인쇄하게 된다. 그리

고 라인을 구성하는 인쇄 유닛도 동시에 여러 색상을 사용할 수 없지만 활용할 수 있는 색상 수 만큼 설치될 수 없다. 따라서 인쇄를 준비하는 과정에서 색상 변경 작업이 수반된다. 색상 변경에 있어 핵심은 잉크 주입구의 세척이다. 미흡한 세척 작업은 인쇄 품질 저하로 이어져 앞선 조색 과정에서의 노력이 무력해질 수 있기 때문에 충분한 시간이 필요하다. 그러나 다른 준비 과정과 달리, 세척은 인쇄 유닛의 가동 중에는 수행될 수 없어 작업 시간을 증가시킨다. 뿐만 아니라 잦은 세척 작업은 생산 비용 또한 증가시키므로, 이를 최소화하는 작업 일정 계획이 필요하다.

이와 관련된 연구로는 Moon, et al.⁹⁾의 잉크 색상 변화가 존재하는 인쇄 공정의 스케줄링이 있다. Burger, et al.¹⁰⁾은 잉크 색상에 따른 교체 시간이 일정한 인쇄 공정에 대해 exact와 heuristic solution을 구하는 방법을 제안하였고, Bae¹¹⁾는 색상에 따라 변경 시간이 다른 인쇄 공정의 작업 순서를 결정하는 규칙을 제시하였다. Kim, et al.¹²⁾은 인쇄 공정과 유사한 염색가공 공정에 대해 유전 알고리즘 기반 스케줄링 방법을 제안하였고, Rifai, et al.¹³⁾은 유사한 공정들을 job sequencing and tool switching problem으로 정의하여 메타 휴리스틱을 적용하였다.

이러한 연구들은 단일 인쇄 라인에서의 적용을 위해 문제에 접근하였고, 하나의 라인에 배정된 주문들에 대한 스케줄링 방법을 제시하였다. 그러나 복수의 라인이 존재하는 생산 환경에서는 이러한 방법을 여러 라인에 각각 적용하는 것이 효율성을 보장하기 어렵다. 또한, 주문이 많아지면 주문을 라인에 배정하는 단계부터 스케줄링 결과에 불확실성이 발생한다. 이에 본 연구에서는 병렬 설비가 존재하는 인쇄 공정의 스케줄링에서 작업자로 인한 불확실성 해소 개선에 기여할 수 있는 스케줄링 방법을 제시한다.

재료 및 방법

1. 문제 소개

본 연구에서는 패키징 인쇄를 위한 오프셋 인쇄 라인이 병렬로 존재하는 인쇄 환경을 다루고자 한다. 모든 주문은 각각 시안 디자인과 함께 정해진 색상 및 인쇄 매수를 필요로 한다. 주문들을 인쇄하기 위해 몇 개의 인쇄 유닛으로 이뤄진 복수의 라인이 존재하고 각 라인의 인쇄 용량은 동일하다. 라인은 한 번의 가동에서 하나의 주문만 인쇄할 수 있고, 라인을 구성하는 인쇄 유닛들은 한 가지 색상만 사용할 수 있다. 따라서 동일한 색상을 계속 사용하는 경우에는 세척이 필요 없지만, 색상을 변경하려면 세척 작업이 선행된다. 세척은 인쇄 유닛 단위로 수행되기 때문에 세척에 소요되는 시간은 해당 인쇄 유닛에서 사용 중인, 즉 세척의 대상이 되는 색상과 다음 인쇄를 위해 주입될 색상의 조합에 의해 결정된다. 여기서 일반적인 규칙은 밝은 색을 어두

Table 1. Illustration of color

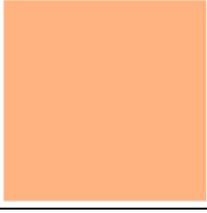
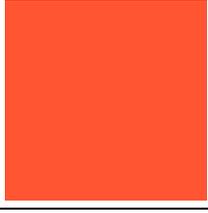
	
Color A ($L^*:87, a^*:27, b^*:-34$)	Color B ($L^*:35, a^*:27, b^*:-34$)
	
Color C ($L^*:100, a^*:80, b^*:-67$)	Color D ($L^*:88, a^*:80, b^*:67$)

Table 2. Illustration of cleaning time by color

Color	A	B	C	D
A	0	4	2	5
B	6	0	3	2
C	7	4	0	3
D	1	2	5	0

운 색으로 교체하는데 소요되는 시간이 어두운 색에서 밝은 색으로 교체하는데 소요되는 시간보다 적고, 연한 색에서 진한 색으로 교체하는데 소요되는 시간이 진한 색에서 연한 색으로 교체하는데 소요되는 시간보다 적다는 것이다. 그 이유는 색상 품질과 관련이 있는데, 예를 들어, Table 1의 Color A, C를 각각 Color B, D로 변경하는 경우에 반대의 경우보다 충분한 세척 시간이 확보되어야 한다.

이로 인해 만약 세척이 되는 색상과 새로 주입될 색상이 복수로 존재하면, 어느 인쇄 유닛을 세척하고 어떤 색을 주입하는지에 있어 가능한 의사결정의 경우의 수가 증가하고 선택에 따라 주문 간 세척 시간이 달라진다. 예를 들어, 색상 간 세척 시간이 Table 2와 같을 때 임의의 두 주문을 인쇄한다고 하자. 두 주문의 연속적인 인쇄에서 세척의 대상이 되는 색상을 A와 B, 새로 주입될 색상을 C와 D라 하면, 주문 간 세척 시간에는 Table 3와 같이 경우의 수가 존재한다.

Table 3의 두 경우 모두 A, B를 C, D로 교체한다. 그러나 교체하는 과정에서 어느 색상 잉크를 담고 있는 인쇄 유닛에 배치하는지에 따라 교체에 소요되는 세척 시간이 4와 8로 차이가 있다. 이러한 차이는 주문이 늘어남에 따라 그 영향이 쌓여 전체 작업 완료 시간의 계산을 어렵게 한다.

이처럼 인쇄 공정의 스케줄링에서 주문의 인쇄를 위한 색

Table 3. Variation of cleaning time by color change

# of cases	Color change	Cleaning time
1	A → C, B → D	2 + 2 = 4
2	A → D, B → C	5 + 3 = 8

상 잉크 배치 방식은 중요한 요인으로 작용하며, 스케줄링 전반에 영향을 미친다. 하지만 작업자마다 성향이나 경험, 능력 등이 다르기 때문에, 의사결정에 차이가 발생하고 스케줄링 결과에 불확실성이 존재하게 된다. 따라서 이러한 불확실성을 해소하고 일관된 결과를 제공할 수 있도록 최적화된 스케줄링 방법이 필요하다.

2. 해결 방법

본 연구에서는 문제를 각 주문들이 인쇄되는 라인 및 작업 순서를 결정하는 부분과 연속적인 인쇄에서 주문 간 준비 방식을 결정하는 부분의 두 부분으로 구분하였다. 라인 및 작업 순서를 결정하는 부분 문제는 인쇄 라인을 차량으로, 각 주문을 방문해야 하는 노드로 간주하여 일종의 차량 경로 문제(Vehicle Routing Problem: VRP)의 형태로 모형화가 가능하다. 이에 따라 주문의 연속적인 인쇄에서 발생하는 준비 시간을 VRP 모형의 노드 간 거리로 가정하였으며, 최적화된 의사결정을 도출하기 위해 할당 문제(Assignment Problem)를 적용하였다.

2.1. 할당 문제

할당 문제는 다수의 작업자와 다수의 작업이 존재하는 상황에서 각 작업자를 작업들에 배정하는 문제이다. 한 명의 작업자는 하나의 작업만 수행할 수 있고, 작업자마다 작업을 수행하는 시간이 다르다. 할당 문제의 목표는 작업 시간 총합의 최소화이다.

본 연구에서는 주문의 연속적인 인쇄를 위해 색상 잉크 배치 방식을 결정하는 문제에 대해 할당 문제를 적용하여 모형화 하였다. 선행 주문의 인쇄 유닛 세팅을 작업자로, 후행 주문의 인쇄 유닛 세팅을 할당이 필요한 작업으로 가정하였으며, 색상 조합에 따른 세척 시간을 작업자별 작업 수행 시간으로 가정하였다. 또한, 매 인쇄에서 사용하는 색상만이 인쇄 유닛에 존재하도록 세척 작업이 수행되는 생산 환경을 가정하여, 할당 문제의 결과가 전체 작업 순서의 영향을 받지 않도록 모형화를 진행하였다.

Fig. 1은 본 연구에서 적용한 할당 문제를 도식화한 형태로, P_0 와 P_1 은 각각 선행 주문과 후행 주문의 인쇄를 위한 설비 세팅을 나타낸다. 각 노드(node)는 주어진 색상을 사용하는 인쇄 유닛을 의미하는데, 'blank'는 사용되지 않고 비어 있음을 의미한다. 노드를 연결하는 아크(arc)는 이어진 세척 의사 결정을 의미하고 그 비용은 색상에 따른 세척 시간이다. 할당 문제의 목표는 총 비용을 최소화하는 조합 탐

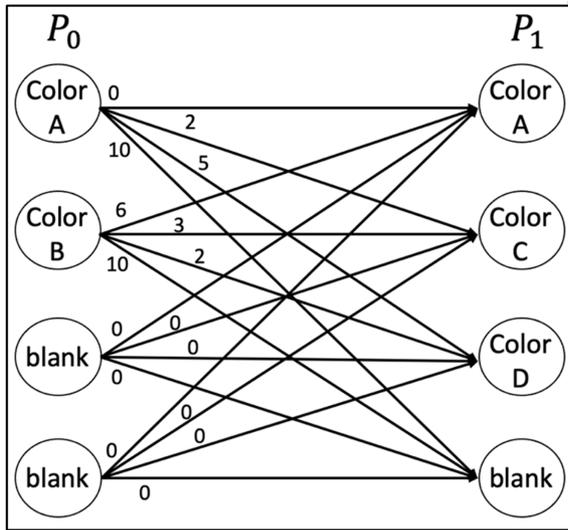


Fig. 1. Illustration of assignment problem.

색에 있다. 따라서 가능한 모든 주문의 순서쌍에 대해 할당 문제를 적용하면, 해당 주문들을 연속해서 인쇄할 때 세척 시간을 최소화하는 색상 배치 방식을 도출할 수 있다. 이러한 할당 문제를 혼합정수계획법(Mixed Integer Programming: MIP)을 이용하여 수리모형으로 나타내면 다음과 같다.

<집합>

P_0 : 선행 주문 인쇄 유닛 집합, $P_0 = \{p_1, \dots, p_m\}$

P_1 : 후행 주문 인쇄 유닛 집합, $P_1 = \{p_1, \dots, p_m\}$

<파라미터>

c_{uv} : 인쇄 유닛의 세팅을 u 에서 v 로 변경하는데 필요한 시간

<결정 변수>

w_{uv} : 인쇄 유닛의 세팅을 u 에서 v 로 변경하면 1, 그렇지 않으면 0.

$$\text{minimize } \sum_{u \in P_0} \sum_{v \in P_1} c_{uv} w_{uv} \quad (1)$$

$$\sum_{u \in \emptyset} w_{uv} = 1 \quad \forall v \in P_1 \quad (2)$$

$$\sum_{v \in Y} w_{uv} = 1 \quad \forall u \in P_0 \quad (3)$$

할당 문제의 규모는 라인이 보유한 인쇄 유닛 수에 영향을 받는다. 반면, 인쇄에 사용되는 색상 수인 인쇄 도수는 일반적으로 10도를 넘지 않으므로, 모든 주문의 순서쌍에 적용함에 있어 각 순서쌍에 대한 결과는 합리적인 시간 내에 계산이 가능하다. 따라서 선형 최적화 솔버(linear optimization solver)인 GLOP을 이용하여 수리 모형의 최적해를

탐색하였다.

한편, 색상에 따른 세척 시간(CT)은 CIE $L^*a^*b^*$ 색좌표를 기준으로 산정하였다. CIE $L^*a^*b^*$ 색좌표는 국제조명위원회(CIE)가 제정한 색채 표준으로 우리 눈의 색 인지 능력을 숫자로 표현한 좌표계이다. L^* 는 밝기를, a^* 는 빨강(+)과 초록(-)의 정도를, b^* 는 노랑(+)과 파랑(-)의 정도를 나타내고, 두 색상 좌표의 차를 계산하여 색의 차이를 수치적으로 나타낼 수 있다. 본 연구에서는 식 (4)를 이용해 세척 시간이 색의 차이에 비례하면서 밝기의 변화에 영향을 받도록 가중치를 설정하였다.

$$CT = 20 - 0.08 \cdot \Delta L^* + 0.02 \cdot \sqrt{\Delta a^{*2} + \Delta b^{*2}} \quad (4)$$

세척 시간은 사용 중인 색상을 비우거나 다른 색상으로 변경할 때 발생한다. 그 외에 같은 색상을 계속해서 사용하거나 빈 인쇄 유닛에 색상을 주입하는 경우에는 세척이 필요 없으므로 0을, 색상을 비우는 경우에는 이후의 색상을 대비하여 가장 높은 세척 시간 가중치를 배정하였다.

2.2. 차량 경로 문제

차량 경로 문제는 Dantzig and Ramser¹⁴⁾에 의해 처음 소개된 이후, 다양한 제약조건을 반영하면서 파생된 문제들의 연구가 진행되고 있다. 전통적인 차량 경로 문제는 다수의 고객을 방문하는 차량에 대해 비용을 최소화하는 경로 탐색을 목표로 한다. 각 고객에게는 한 대의 차량이 한 번씩만 방문하여 일련의 서비스를 제공하고 각 차량은 디포(depot)에서 출발하여 디포로 돌아와야 한다. 이를 그림으로 도식화하면 다음 Fig. 2와 같이 나타낼 수 있다.

Fig. 2는 3개의 차량으로 10개의 고객을 방문하는 VRP 문제를 도식화한 형태로, 그림에서 노드는 방문해야 하는 고객을 나타낸다. 아크는 노드와 노드를 연결하여 고객을 방문하는 순서를 나타내며 각각에 해당하는 비용이 주어져 있다. 모든 노드는 연결되어 있지만 노드마다 다른 노드로 이

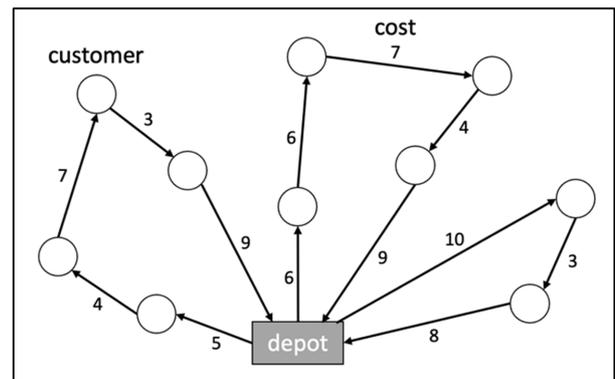


Fig. 2. Illustration of vehicle routing problem.

동하는 비용이 다르므로, VRP에서는 주어진 차량들에 대해 비용 측면에서 최적화된 경로 계획을 도출한다.

이러한 VRP를 적용하여 인쇄 공정에서 주문들이 인쇄되는 라인 및 작업 순서를 결정하는 부분 문제를 모형화 하였다. 노드에 해당되는 주문은 납기가 최적화된 상태로, 순서 변경에 따른 납기 문제가 발생하지 않는 n 개의 주문이 접수되었다고 가정한다. 접수된 주문들을 처리하는 인쇄 라인은 노드를 방문하는 차량으로 l 대가 존재하며, l 의 값에 따라 단일 및 병렬 라인 모두에 적용 가능하다. 차량의 노드 방문은 해당 주문의 처리를 나타내므로 한 노드에서 다른 노드로 이동하는 비용은 한 주문의 인쇄를 마쳤을 때 다른 주문의 인쇄를 위한 준비 작업부터 인쇄를 완료하기까지의 시간으로 볼 수 있다. 이에 따라 노드 간 이동 비용을 주문 간 준비 작업에 소요되는 시간과 후행 주문의 인쇄 시간으로 구성된 시간 함수로 정의하였다. 색상 잉크 배치 및 세척 작업이 수행되는 준비 작업은 작업자로 인한 불확실성이 존재하기 때문에 할당 문제를 통해 최적화된 의사결정 및 그에 따른 소요 시간을 도출하였고, 인쇄 시간은 주문의 인쇄 매수에 따라 값을 갖도록 설정하였다. 또한, 각 라인에 주어진 마지막 주문의 인쇄가 완료되는 시점에 인쇄 유닛이 초기 상태로 존재하지 않아도 되므로 디포로 돌아오는 비용을 0으로 설정하였다. 이처럼 정의된 시간 함수는 공정의 여러 특성을 반영할 수 있도록 변형이 가능하다. 본 연구에서는 인쇄 공정에서 주문들의 기본적인 정보인 사용 색상과 인쇄 매수로 시간 함수를 정의하였지만, 다른 요인의 가중치를 시간의 형태로 정의가 가능하다면 스케줄링에 반영할 수 있다.

이를 바탕으로 병렬 오프셋 인쇄 공정의 스케줄링 모형을 다음과 같은 변수들과 제약식을 통해 정의한다.

<집합>

V : 주문 집합; 0은 초기 상태, $V = \{0, 1, 2, \dots, n\}$

K : 라인 집합, $K = \{0, 1, 2, \dots, l\}$

<매개 변수>

C_{ij} : 주문 i 의 인쇄 직후에 주문 j 의 인쇄가 완료되는데 필요한 시간

λ : 가장 늦게 완료되는 라인의 작업 시간에 대한 가중치

<결정 변수>

x_{ij}^k 라인 k 에서 주문 i 를 인쇄한 직후에 j 를 인쇄하면 1, 그렇지 않으면 0.

T_k : 라인 k 의 작업 완료 시간

$$\text{minimize } \lambda \cdot \max_k(T_k) + \sum_{k \in K} T_k \tag{5}$$

$$\sum_{i \in V} \sum_{k \in K} x_{ij}^k = 1 \quad \forall j \in V - \{0\} \tag{6}$$

$$\sum_{i \in V} x_{ih}^k - \sum_{j \in V} x_{hj}^k = 0 \quad \forall h \in V, k \in K \tag{7}$$

$$\sum_{j \in V - \{0\}} x_{0j}^k = 1 \quad \forall k \in K \tag{8}$$

$$\sum x_{ii}^k = 0 \quad \forall i \in V - \{0\} \tag{9}$$

Table 4. Actual order data used in the experiment

# of Orders	PLCODE	INKCODE List	Production Quantity
1	PL210402036	P1850, 020120002, 020120007, 020120014, P7644, 020120009, 020120015	266
2	PL211008085	DS209, 020220009, 020220007	2182
3	PL211008075	DS209, 020220009, 020220007	2657
4	PL210331060	베이지별색19, 020220009, 020220015, 020220014, 020220016, P199K	983
5	PL211019001	020220013, 020220009, P200Y	70834
6	PL210326037	DS333, P479A, 020120002, P412E, DS334, P569	600
7	PL210709002	020120002, 020120007, 020120014, 020120015	2482
8	PL210407036	020120002, 020120007, 020120014, P151F, 020120015	7011
9	PL211019004	020120002, 020120007, 020120014, P9501	42013
10	PL210916009	020220009, P297C, P5555	977
11	PL210923025	020220013, 020220064, 020220015, 020220022, 020220023, P3415B	19039
12	PL211103002	020220013, 020220009, DS293	2011
13	PL210402037	DS475, 회색2, 020220009, 020220015, 020220014, 020220016, 핑크베다2, P410	400
14	PL211103003	020220013, 020220009, DS293	2427
15	PL210408134	002박색15, 020220009, 020220015, 020220014, 020220016, P221B, P207N, 020220013	2800
16	PL211019003	020120002, 020120014, 020120007, 020120015	35936
17	PL210915058	P7533C, 020120002, 020120007, 020120014, P7530A, DS376, 020120015	3302

$$u_i - u_j + l \sum_{k \in V} x_{ij}^k \leq l - 1 \quad \forall i, j \in V, \quad \begin{matrix} 1 \leq i \neq j \leq l \\ u_i, u_j: \text{arbitrary} \end{matrix} \quad (10)$$

$$T_k = \sum_{i \in V} \sum_{j \in V} C_{ij} x_{ij}^k \quad \forall k \in K \quad (11)$$

$$x_{ij}^k \in \{0, 1\} \quad \forall i, j \in V, k \in K \quad (12)$$

$$T_k \geq 0 \quad \forall k \in K \quad (13)$$

목적함수 (5)는 두 개의 항으로 구성된다. 첫 번째 항은 가장 늦게 작업이 완료되는 라인의 작업 시간을 구하고, 두 번째 항을 통해 작업 시간의 총합을 계산한다. 본 연구에서는 작업 완료 시간이 가장 늦은 라인의 작업 시간을 최소화하여 모든 라인의 작업이 최대한 빨리 완료되도록 한다. 제약식 (6)은 모든 주문이 한 번만 인쇄되도록 하고, 제약식 (7)은 작업의 연속적인 흐름을 나타낸다. 제약식 (8)은 모든 라인이 초기 상태에서 시작하고 증도에 복귀하지 못하도록 제한한다. 제약식 (9)와 (10)은 각각 동일 주문에서의 이동과 부분 경로의 형성을 방지한다. 제약식 (11)을 통해 라인 별 작업 시간을 계산하며, 제약식 (12)와 (13)은 결정 변수에 관한 식이다.

2.2.1. 해법

위 수리모형을 풀면 최적해를 얻을 수 있지만, 주문 수가 늘어날수록 계산 시간이 크게 증가하여 규모가 큰 문제에 대해서는 합리적인 시간 내에 결과를 얻기 어렵다. 이에 본 연

구에서는 합리적인 시간 내에 좋은 결과를 얻기 위해 메타 휴리스틱(meta-heuristics) 방법 중 하나인 지역 안내 탐색(Guided Local Search: GLS)을 적용하여 해결하고자 한다.

GLS는 현재해를 수정하여 만든 이웃해를 기반으로 해를 개선해 나가는 지역 탐색(Local Search) 방법¹⁵⁾의 하나로, 지역 탐색 알고리즘이 지역 최적에 수렴하는 방지하기 위해 증강 비용 함수(augmented cost function)를 사용한다. 이는 지역 최적에 수렴하면 그 때의 비용을 주변 탐색 공간보다 높게 만들어 지역 최적에서 벗어나도록 유도한다. GLS의 구체적인 알고리즘의 내용은 다음과 같다¹⁶⁾.

Algorithm 1 Pseudocode of Guided Local Search

```

1:  input:  $g, \lambda, [I_1, \dots, I_M], [c_1, \dots, c_M], M$ 
2:   $k \leftarrow 0$ 
3:   $s_0 \leftarrow$  random or heuristically generated solution
4:  for  $i = 1$  to  $M$  do
5:     $p_i \leftarrow 0$ 
6:  endfor
7:  while (! StoppingCriterion)
8:     $h \leftarrow g + \lambda \sum p_i I_i$ 
9:     $s_{k+1} \leftarrow$  LocalSearch( $s_k, h$ )
10:   for  $i = 1$  to  $M$  do
11:      $util_i \leftarrow I_i(s_{k+1}) * c_i / (1 + p_i)$ 
12:   endfor
13:   for each  $i$  such that  $util_i$  is maximum do
14:      $p_i \leftarrow p_i + 1$ 
15:   endfor
16:    $k \leftarrow k + 1$ 
17: endwhile
18:  $s^* \leftarrow$  the best solution found with respect to  $g$ 
19: return  $s^*$ 
    
```

Table 5. The time required between orders

# of orders	0	1	2	3	4	5	6	7	8	9	10	11	12	3	14	15	16	17
0	0	1	11	13	5	354	3	12	35	210	5	95	10	2	12	14	180	17
1	0	0	185	187	155	528	138	102	117	322	182	246	185	142	187	153	270	81
2	0	62	0	13	44	395	66	76	99	274	50	155	53	40	55	52	244	77
3	0	62	11	0	44	395	66	76	99	274	50	155	53	40	55	52	244	77
4	0	122	136	138	0	480	130	150	167	350	133	197	137	37	139	49	318	140
5	0	61	51	53	43	0	64	75	98	273	49	136	31	38	33	35	243	76
6	0	99	151	153	126	495	0	132	147	330	149	214	152	114	154	128	300	115
7	0	1	96	98	82	441	63	0	35	232	94	171	98	76	100	89	180	17
8	0	19	125	127	103	470	85	42	0	262	123	194	127	95	129	108	210	35
9	0	17	95	97	81	439	62	33	55	0	93	170	96	75	98	87	201	32
10	0	59	50	52	42	393	62	72	95	270	0	151	49	36	51	49	240	74
11	0	123	152	154	109	477	129	149	168	349	151	0	133	98	135	92	317	139
12	0	62	52	54	44	375	65	75	98	273	48	135	0	38	12	34	243	77
13	0	172	195	197	104	539	184	207	224	407	193	252	196	0	198	95	375	189
14	0	62	52	54	44	375	65	75	98	273	48	135	10	38	0	34	243	77
15	0	173	196	198	105	520	185	208	224	408	194	234	177	84	179	0	376	190
16	0	1	96	98	82	441	63	12	35	232	94	171	98	76	100	89	0	17
17	0	61	182	184	153	526	136	102	115	321	180	243	184	138	186	151	270	0

Table 6. Result of experiments on actual data

Algorithm	Objective Value	CPU Time(sec)	# of Lines	Work Schedule (makespan: min)
MIP	53212*	90	1	9 - 16 - 7 - 8 - 17 (493)
			2	10 - 2 - 3 - 12 - 14 - 11 - 15 - 4 - 13 (502)
			3	5 - 6 - 1 (517**)
SA	54132	10	1	10 - 6 - 16 - 7 - 8 - 17 - 1 (510)
			2	3 - 2 - 5 - 14 - 12 - 15 (496)
			3	9 - 11 - 4 - 13 (526**)
GLS	53636	60	1	10 - 14 - 12 - 5 - 4 - 13 (521**)
			2	9 - 16 - 7 - 17 - 1 (501)
			3	3 - 2 - 8 - 6 - 11 - 15 (514)

*: Optimal solution.

** : Maximum makespan.

Table 7. Illustration of scheduling result obtained by our solution

Line 1								
Order	Unit 1	Unit 2	Unit 3	Unit 4	Unit 5	Unit 6	Unit 7	Unit 8
9	020120002	020120007	020120014	P9501				
16	020120002	020120007	020120014	020120015				
7	020120002	020120007	020120014	020120015				
8	020120002	020120007	020120014	020120015	P151F			
17	020120002	020120007	020120014	020120015	P7533C	P7530A	DS376	
Line 2								
Order	Unit 1	Unit 2	Unit 3	Unit 4	Unit 5	Unit 6	Unit 7	Unit 8
10	020220009	P297C	P5555					
2	020220009	DS209	020220007					
3	020220009	DS209	020220007					
12	020220009	020220013	DS293					
14	020220009	020220013	DS293					
11	020220064	020220013	020220015	020220022	020220023	P3415B		
15	020220009	020220013	020220015	P221B	P207N	020220014	002박색15	020220016
4	020220009	베이지별색 19	020220015			020220014	P199K	020220016
13	020220009	P410	020220015	DS475	핑크베다2	020220014	회색2	020220016
Line 3								
Order	Unit 1	Unit 2	Unit 3	Unit 4	Unit 5	Unit 6	Unit 7	Unit 8
5	020220013	020222009	P200Y					
6	020120002	P412E	P479A	DS333	DS334	P569		
1	020120002	020120014	P7644	020120007	P185O	020120009	020120015	

실험 및 고찰

본 연구에서 제시하는 알고리즘은 할당 문제를 통해 주문 간 준비 작업에서의 최적 의사결정과 소요 시간을 계산하고 이를 입력으로 하는 VRP를 통해 스케줄링 결과를 도출한다. 할당 문제와 달리, VRP는 수리 모형을 통해 최적해를 얻는데 많은 시간이 소요되기 때문에 메타 휴리스틱인 GLS 알고리즘을 적용하였다. 이러한 방법의 현장 적용

성을 검토하기 위해 실용적인 시간 내에 일관된 결과를 제공하는지 실험을 통해 확인한다. 알고리즘은 Python 환경에서 Google OR-Tools 라이브러리를 이용하여 구현하였고, 실험은 Python 3.9를 이용하여 2.6 GHz quad core CPU, 16 GB RAM의 환경에서 수행하였다.

먼저, 병렬 인쇄 설비를 보유한 중소 패키징 인쇄 기업(덕수산업)의 실제 데이터로 small test를 위한 예제를 구성하였다. 전처리를 거쳐 수집한 17개의 주문에 대해 3대의 인

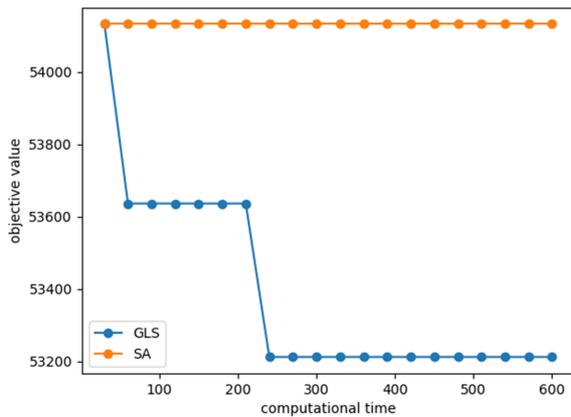


Fig. 3. Objective function change over computation time.

쇄 라인을 가동하여 전체 작업 완료 시간을 최소화하는 일정 계획을 탐색한다. 각 인쇄 라인은 8개의 인쇄 윗너트로 이뤄져 있고 분당 200매를 인쇄할 수 있다고 가정한다. 각

주문의 사용 색상과 인쇄 매수는 Table 4와 같다. 주문 처리에 소요되는 시간으로 정의하여 노드 간 이동 비용을 계산하고, VRP에 대해 GLS 알고리즘으로 도출한 결과와 MIP 수리 모형의 최적해를 비교한다. 수리 모형의 계산에는 상용 최적화 프로그램인 Gurobi(Ver. 9.1.5)를 이용하였다.

Table 5는 특정 주문의 인쇄가 완료된 상태에서 다른 주문을 완료하기까지 소요되는 시간을 나타낸다. 할당 문제를 통해 연속적인 인쇄에 따른 색상 배치 방식 및 세척 시간을 계산하였고, 이에 인쇄 매수와 비례하는 인쇄 시간을 더해 값을 산출하였다. 주문 0은 색상 잉크가 배치되지 않은 초기 상태를 의미하며, 선행 주문을 기준으로 후행 주문이 사용하는 색상과 인쇄 매수에 따라 시간 가중치가 부여됨을 확인할 수 있다. 예를 들어, 주문 2의 인쇄 이후 연속적으로 동일한 색상을 사용하는 주문 3을 인쇄하는 경우 (C_{23})가 인쇄 매수가 비슷하지만 동일한 색상을 사용하지 않는 주문 7을 인쇄하는 경우(C_{27})보다 낮은 시간 값을 갖는다. 또한, 인쇄 매수가 많은 주문 5가 후행 주문으로 인

Table 8. Result of experiment on generated data

# of lines	# of orders	MIP			GLS		
		Objective value (makespan)	CPU time (sec)	MIP Gap (%)	Average objective value (makespan)	CPU time (sec)	Objective gap ¹⁾ (%)
7	30	57329 (536)	3600	0.9	57335 (536)	300	0
	40	86440 (808)	3600	0.5	86650 (810)	300	0.2
	50	101498 (949)	3600	0.9	101416 (948)	300	-0.1
	60	128663 (1203)	3600	1.3	128378 (1200)	300	-0.2
	70	143121 (1338)	3600	1.6	142595 (1333)	300	-0.3
	80	186116 (1740)	3600	2.0	184106 (1721)	300	-1.1
	90	TIME**	3600	-	204985 (1916)	300	-
	100	242822 (2270)	3600	1.2	241703 (2259)	300	-0.5
10	30	37338 (340)	3600	1.9	37349 (340)	300	0.03
	40	58027 (528)	3600	1.1	57963 (527)	300	-0.1
	50	68567 (624)	3600	1.7	68167 (620)	300	-0.5
	60	87279 (794)	3600	1.2	87299 (794)	300	0.02
	70	98262 (894)	3600	2.3	97414 (886)	300	-0.9
	80	127400 (1159)	3600	1.3	127020 (1155)	300	-0.3
	90	143345 (1304)	3600	1.9	142631 (1297)	300	-0.5
	100	169283 (1540)	3600	1.5	169091 (1538)	300	-0.1
15	30	22699* (199)	109.66	0	22699 (199)	300	0
	40	36865 (322)	3600	3.0	36571 (319)	300	-0.8
	50	43110 (376)	3600	2.7	42821 (373)	300	-0.7
	60	56253 (491)	3600	3.1	55265 (481)	300	-1.8
	70	63129 (550)	3600	2.9	62177 (541)	300	-1.5
	80	84038 (733)	3600	2.9	82954 (722)	300	-1.3
	90	97227 (851)	3600	5.3	94449 (823)	300	-2.9
	100	112806 (983)	3600	2.8	111034 (966)	300	-1.6

Objective gap¹⁾: {(Average objective value of GLS) - (Objective value of MIP)} / (Objective value of MIP)

*: Optimal solution.

TIME**: Any feasible solutions cannot be found during 1 hour.

쇄되는 경우(C_{is})는 전반적으로 높은 시간 값을 갖는다.

Table 6는 Table 5를 입력으로 하는 VRP를 수리모형, GLS가 도출한 결과이며 또 다른 메타 휴리스틱인 시뮬레이션 어닐링(Simulated Annealing: SA)의 결과도 함께 비교하였다. 문제의 규모가 작은 small test에서는 수리모형으로도 약 2분 내에 최적해를 도출할 수 있었다. 이와 비교하여 GLS와 SA는 비교적 짧은 시간 내에 최적에 근사한 해를 도출하였다. 최적해보다는 그 값이 좋지 않지만, 그 차이가 작아 수용 가능한 수준으로 판단된다. Fig. 3는 두 메타 휴리스틱의 계산 시간에 따른 목적 값의 변화를 비교한 결과이다. 두 방법은 동일한 초기해에서 시작하였지만, 10분까지 계산 시간을 단계적으로 증가시키며 적용한 결과에서 GLS가 SA보다 더 짧은 시간 내에 해의 품질을 개선하였음을 확인할 수 있다. 이러한 결과는 현장에서의 적용을 위해 도식화 된 형태로 제공될 필요가 있다. 따라서 최적화된 주문을 인쇄하는 순서와 그에 따라 인쇄 유닛에 색상 잉크를 배치하는 방식을 Table 7과 같이 도식화하였다.

Small test의 결과를 바탕으로, 더 큰 생산 환경에서 더 많은 주문이 주어진 상황에서의 적용성을 확인하는 실험을 진행하였다. 각 인쇄 라인은 8개의 유닛으로 구성됨을 가정하고, 주문은 30~100개, 라인 수는 7~15개로 늘려가며 24개의 예제를 생성하였다. 예제를 구성하는 주문은 텍스산업이 보유한 데이터에 기반하여 3,331가지의 조색 중 무작위로 2~8가지의 색상을 사용하고 (50, 25000)의 값을 인쇄하도록 설정하였다. GLS 알고리즘의 계산 시간은 5분으로 제한하고 반복 10회의 목적 값의 평균으로 최적해와 비교하였다. 수리모형을 통한 최적해 탐색은 문제의 규모가 증가함에 따라 계산에 많은 시간이 소요되는 경우가 있기 때문에 1시간 경과 시 실험을 멈추고 그 때의 값을 최적해로 간주하였다.

Table 8은 두 방법으로 구한 최적해와 계산 시간을 비교하였다. 수리모형에서 MIP Gap은 종료 시점에 보유한 해와 한계값의 상대적인 차이로 0에 가까울수록 최적해에 가깝다고 판단할 수 있고, 본 실험에서는 전반적으로 낮은 값을 보인다. 그러나 최적해를 도출한 1개의 예제를 제외하면, 모두 계산에 1시간이 소요되었고 특정 예제에서는 시간 내에 어떠한 해도 탐색하지 못하였다. 반면, GLS 알고리즘은 모든 예제에 대해 5분의 제한 시간 내에 해를 도출하였다. 10번의 반복에서 해의 품질 또한 수리모형의 최적해와 비교를 통해 그 차이가 크지 않음을 확인할 수 있으며, 일부 예제의 경우 더 개선된 값을 나타내었다.

요 약

본 연구에서는 패키징 인쇄를 위한 병렬 오프셋 인쇄 공정의 스케줄링 문제를 다루었다. 문제에 대해 두 부분으로

구분하여 접근하였고, 각각 할당 문제와 차량 경로 문제를 적용하여 수리적으로 모형화 하였다. 스케줄링 모형의 현장 적용성은 실험을 통해 검토하였다. 실제 데이터로 구성된 작은 규모의 문제에서는 수리모형으로도 실용적인 시간 내에 최적해를 도출할 수 있었고 이와 비교하여 메타 휴리스틱의 성능을 확인하였다. 기업이 보유한 데이터를 바탕으로 문제 규모를 확장한 실험에서는, 수리모형의 최적해와 비교하여 메타 휴리스틱이 해의 품질을 보장하면서 시간적 효율성을 확보할 수 있었다.

본 연구는 수작업 위주의 기존 방식은 주체(작업자)에 따라 스케줄링의 결과에 불확실성이 존재하는 문제에 주목하였다. 이러한 불확실성은 전체 생산 비용의 증가를 가져오기 때문에 이를 개선할 수 있도록 실용적인 시간 내에 일관된 결과를 제공하는 스케줄링 모형을 제시하였다. 제시한 모형은 단일 라인과 병렬 라인 모두에 적용되어 작업자의 경험에 의존하던 기존의 방식을 개선하는데 도움이 될 것으로 판단되며, 시간 함수의 정의를 통해 다른 요인들을 반영하는 연구로의 확장이 가능하다는 의의를 갖는다.

향후 주문의 납기, 복수의 라인에서 동일 주문 인쇄, 동일하지 않은 라인의 인쇄 용량, 조색 난이도 등을 고려하는 연구로의 확장을 통해 패키징 인쇄 분야의 스마트 생산 시스템 도입에 기여할 수 있을 것으로 기대된다.

사 사

본 연구는 산림청(한국임업진흥원) 산림과학기술 연구개발사업(2021383A00-2223-0101)의 지원에 의하여 이루어진 것입니다.

참고문헌

1. Korea Standard, 2017. Glossary of terms for packaging, in KS T 1001.
2. Kim, K.S. and Kang, H.S. 2011. An Effect on Customer Attitude and Purchase Intention of Soap Product Package Design - Focused on Soap Product -, Journal of Korea Design Forum. (30): 33-42.
3. Kang, D.H., Hwang, S.W. and Shim, J.K. 2016. Outline of Korean Packaging Industry, Korean J. Packag. Sci. Tech. 22(3): 135-142.
4. Tae, H.C. 2019. Introduction to packaging printing process and analysis of optimization issues, Proceedings of the Korean Institute of Industrial Engineers Fall Conference, Seoul, Korea, pp. 361-370.
5. Hyun, Y.J., Park, J.S. and Tae, H.C. 2021. Prediction of Color Reproduction using the Scattering and Absorption of Coefficients derived from the Kubelka-Munk model in Package Printing, Korean J. Packag. Sci. Tech. 27(3): 203-210.
6. Seo, J.H. and Tae, H.C. 2020. A Study on Artificial

- Intelligence Algorithm to Determine Basic Ink Mixing Ratio for Color Prediction of Spot Color – Focused on Offset Printing for Spot Color Prediction through Basic Ink Formulation in the Packaging Printing Market, *Journal of Korea Society of Color Studies*. 34(4): 47-55.
7. Kim, Y.S., Uhm, H.S. and Tae, H.C. 2022. Deep Neural Network Modeling for the Prediction of Reflexibility from Printed Outputs in the Spot Color Printing System – Focusing on the Aggregating and Processing of Dataset from the Real-world Printing System, *Journal of Korea Society of Color Studies*. 36(1): 5-12.
 8. Tae, H.C. 2019. Development case of color mixing ink amount prediction algorithm in package printing process, *Proceedings of the Korean Institute of Industrial Engineers Fall Conference*, Seoul, Korea, pp. 3307-3319.
 9. Moon, J.K., Uhm, H.S. and Tae, H.C. 2021. Scheduling of Printing Process in which Ink Color Changes Exist, *Journal of Korea Society of Industrial and Systems Engineering*. 44(4): 32-42.
 10. Burger, A.P., Jacobs, C.G., van Vuuren, J.H. and Visagie, S.E. 2015. Scheduling multi-color print jobs with sequence-dependent setup times, *Journal of Scheduling*. 18(2): 131-145.
 11. Bae, J.H. 2020. Dispatching Rule based on Chromaticity and Color Sequence Priorities for the Gravure Printing Operation, *Journal of Korean Society of Industrial and Systems Engineering*. 43(3): 10-20.
 12. Kim, S.H., Park, S.E., Yoon, Y.S. and Oh, H.W. 2021. A Study on the Scheduling of Dyeing Processing Production Based on Genetic Algorithm, *The Journal of Korean Institute of Communications and Information Sciences*. 46(1): 192-200.
 13. Rifai, A.P., Mara, S.T.W. and Norcahyo, R. 2022. A two-stage heuristic for the sequence-dependent job sequencing and tool switching problem, *Computers & Industrial Engineering*. 163.
 14. Dantzig, G.B. and Ramser, J.H. 1959. The Truck Dispatching Problem, *Management Science*. 6(1): 80-91.
 15. Hwang, J.H. 2021. Neighbor Generation Strategies of Local Search for Permutation-based Combinatorial Optimization, *Journal of The Korea Society of Computer and Information*. 26(10): 27-35.
 16. Voudouris, C. and Tsang, E.P.K. 1999. Guided Local Search and its application to the Travelling Salesman Problem, *European Journal of Operational Research*. 113(2): 469-499.
- 투고: 2022.06.30 / 심사완료: 2022.08.11 / 게재확정: 2022.08.29