

# 물류 과학 기술 연구

Journal of Logistics Science & Technology

2020년 12월 1권 1호

THE SOCIETY OF LOGISTICS  
SCIENCE AND TECHNOLOGY

# 물류과학기술연구

Journal of Logistics Science & Technology

제 1권 제 1호

2020년 12월

---

## 목 차

### 1. 딥러닝과 스테레오 카메라를 이용한 객체 거리 추출 방법

박민홍, 원종운, 조재훈, 김용태 / 3

### 2. 온라인 리테일러에서 수요 패턴 예측을 위한 컨볼루션 LSTM 적용 방안

정재원, 임성택, 이승민, 공형준 / 15

### 3. 이열 배치설계를 위한 Cuckoo Search 기반의 최적화 알고리즘 개발

윤영민, 채준재 / 31

### 4. Bass Model 기반의 접이식 컨테이너 수요 예측

김학성, 하현구 / 49

# 딥러닝과 스테레오 카메라를 이용한 객체 거리 추출 방법

박민홍<sup>1</sup>, 원종운<sup>2</sup>, 조재훈<sup>3</sup>, 김용태<sup>1</sup>

<sup>1</sup>한경대학교 ICT로봇기계공학부

<sup>2</sup>한국철도기술연구원

<sup>3</sup>한경대학교 스마트물류기술연구센터

## Object distance extraction method using deep learning and stereo camera

Min-Hong Park<sup>1</sup>, Jong-Un Won<sup>2</sup>, Jae-hoon Cho<sup>3</sup>, Yong-Tae Kim<sup>1</sup>

<sup>1</sup>Dept. of ICT, Robot and Mechanical Engineering, Hankyong National University

<sup>2</sup>Korea Railroad Research Institute, Logistics System Research Division

<sup>3</sup>Smart Logistics Technology Institute, Hankyong National University, Korea

In this paper, we proposed a method to calculate the linear distance value according to the location of the delivery box and the stereo camera. The proposed method recognizes individual delivery boxes using YOLOv4, and calculates the vertical plane position of the camera and the linear distance between the boxes using the IMU sensor mounted on the stereo camera. In order to evaluate the performance of the proposed algorithm, a laboratory-scale experimental space was constructed, and the measured distance of the actual delivery box and the output distance of the proposed algorithm were compared.

**Keywords:** YOLOv4, IMU sensor, delivery boxes, stereo camera.

### 1. 서론

최근 코로나 19 사태는 사람과 화물의 이동패턴에 큰 영향을 미치고 있다. 사람과 화물의 이동에 있어서 검역과 보안기능이 강화되어 과거보다 이동의 자유가 제한되고 더 많은 시간과 비용 소요될 것으로 전망된다. 한편 유통의 온라인화는 더 빠른 속도로 진행되면서 택배와 같은 도시 물류수요가 크게 증가하고 있다(Lee, 2020).

온라인 쇼핑시장은 2017년 78조2000억 원으로 집계 되었으며, 5년마다 약 2배 이상 성장하고 있다. 2022년에는 176조2000억 원에서 189조8000억 원으로 늘어날 전망이다. 또한, 전자상거래 택배 물량도 2배 성장해 10억 박스가 넘을 것으로 예상된다(Soh et al., 2020). 이러한 택배 시장의 변화는 택배 회사들로 물동량 처리 효율을 높이기 위해서 인공지능, 자동화 시설의 필요성을 증가 시키고 있다. 인공지능 및 딥러닝을 이용한 창고관리는 물론 상품의 인식 및 분류 등 다양하게 적용되어지고 있다.

최근 택배 물류 기술 분야에서는 노동력 비중에 큰 상·하역 작업에 자동화 시스템을 도입하는 연구 개발들이 활발히 진행되어지고 있다. 특히 자동 하역장비에서 카메라와 딥러닝 기술을 이용한 박스와 화물을

인식하는 기술들은 택배 물류의 자동화를 위한 중요한 기술 중의 하나이다(Won et al., 2019).

딥러닝 기술은 합성곱 신경망을 이용한 다양한 구조의 딥러닝 기술들이 개발되어 지고 있다. 합성곱 신경망이 제안된 이후로 객체가 존재하는 영역을 먼저 구분하여 성능을 개선한 R-CNN이 개발되었으며 R-CNN의 연산시간을 개선시킨, Fast R-CNN(Girshick, 2015), Faster R-CNN(Ren, 2015) 등의 구조들이 개발되어 졌다. 또한 정확성을 높이기 위한 Masked R-CNN(He, 2017)의 개발은 산업현장에 적용할 만한 성능을 보였다. 이후 다양한 대기업들이 딥러닝 기술 연구에 참여하면서 산업현장에서의 딥러닝 기술이 적용이 빠르게 확산되어지고 있다. 특히 구글에서 개발하고 오픈소스로 제공되는 YOLO모델은 객체 인식의 속도와 성능은 다양한 산업분야에서 성공적인 적용이 이루어지고 있다(Choi et al., 2019).

한편 딥러닝 기술과 함께 카메라 기술 또한 빠르게 발전하고 있으며 최근에는 카메라를 이용한 객체의 거리값을 추출하는 저가형 스테레오 카메라 시장이 빠르게 발전하면서 딥러닝 기술과 카메라를 이용한 기술들이 다양하게 연구되어지고 있다. 스테레오 카메라는 객체와 카메라 사이의 거리를 알 수 있는 장점이 있다. 그러나 카메라의 높이와 기울기에 따라 거리값이 달라지기 때문에 적용분야에 따라 적절한 보상 알고리즘 개발이 필요하다. 자동 택배하역 장비의 경우 하역장비의 초기 위치에서 목표 박스 및 화물로 이동하여 성공적인 작업을 수행하기 위해서는 하역장비의 초기 좌표와 카메라의 설치 위치, 높이 그리고 기울기에 따른 객체의 직선거리를 계산할 필요가 있다.

본 논문에서는 택배박스과 카메라 위치(높이 및 기울기)에 따른 직선 거리값을 계산하는 방법을 제안한다. 제안한 방법은 딥러닝을 기술 중 YOLO-v4(Alexey, 2020)를 이용하여 개별 택배 박스들을 인식하고 인식된 박스들을 스테레오 카메라에 장착된 IMU(Inertial Measurement Units)센서를 이용하여 카메라의 수직 평면 위치와 박스들 간의 직선거리를 계산한다. 제안된 알고리즘의 성능을 평가하기 위하여 연구실 규모의 택배 화물 공간을 구축하여 실험을 진행하였으며 실제 택배 박스의 실측거리와 제안된 알고리즘의 출력 거리값을 비교하였다. 실험결과 제안된 알고리즘의 출력 거리값이 만족할만한 성능을 보임을 확인할 수 있었다.

## 2. 딥러닝 기반 객체 인식

이미지나 영상에서 객체인식을 기술들은 다양한 방법으로 연구되어져 왔다. 초기의 객체인식 알고리즘들은 히스토그램이나 푸리에 변환과 같은 방법들에 의해서 특징점들을 추출하고 각각의 특징점들과 가장 유사한 객체들을 판별하는 기술이었으며 신경망 및 SVM(Support Vector machine)등과 같은 분류기 등을 사용하였다. 2012년 ILSVRC(Large Scale Visual Recognition Challenge)에서 딥러닝에 기반한 AlexNet이 기존의 방법들에 비해 우수한 성능을 보임으로써 딥러닝에 의한 객체인식 기술이 큰 관심을 받기 시작하였다(Krizhevsky et al., 2012). 이후 2013의 ZF Net(Zeiler et al., 2014), 2014년에 GoogLeNet(Szegedy, et al., 2015), 2015년 ResNet(He et al., 2016) 등의 딥러닝 기반의 기술들이 ILSVRC에서 우승하면서 딥러닝 기술의 성능을 입증하였다.

최근 구글은 오픈 딥러닝 기술인 YOLO모델들을 연달아 발표하면서 지속적인 성능 개선이 이루어지고 있다. 본 논문에서는 YOLO-v4를 이용한 객체인식 기술을 기반으로 객체를 인식하고 제안된 알고리즘을 적용하였다.

### 2.1 객체 인식을 위한 데이터셋 구축

본 연구에서는 다양한 택배 화물 중 박스를 대상 객체로 선정하였으며 딥러닝 기술을 이용하여 박스 인식을 수행하기 위하여 5,102장의 학습 이미지 수집하여 이용하였다. 다양한 택배환경에서 박스들은 단일 박스로 존재하기 보다는 하나의 이미지에 다양한 위치와 형태로 적재되어진 경우가 많다. 또한 외부의 힘에 의해 박스의 일그러짐 등의 왜곡들에 대해서 정확한 위치와 인식을 위하여 이미지의 라벨링을 다각형 형태로 표시하였다. 사용된 데이터셋의 예시는 아래와 같다.



(a) example of box data



(b) example of box data

Figure 1. example of box data

객체에 대한 영역을 표시하기 위하여 라벨링 기법의 하나인 VIA(Vgg Image Annotator)를 이용하였다. 박스에 대한 데이터셋은 하나의 객체(박스)의 영역을 세부적으로 표현하기 위하여 다각형의 점들로 표현하였으며 하나의 이미지에 박스의 개수와 하나의 객체가 갖는 이미지상의 위치 및 하나의 객체를 표현하는 사각형의 경계박스(Bounding Box)로 구성하였다.



Number of object : 6

Bounding Box (x, y, width, height)

Each area of object

Figure 2. Information of dataset of box image

## 2.2 객체 인식을 위한 딥러닝 모델

객체 인식을 위한 모델은 YOLOv4 모델을 이용하여 학습을 진행하였다. YOLOv4의 전체적인 구조는 아래와 같은 구조를 보인다.

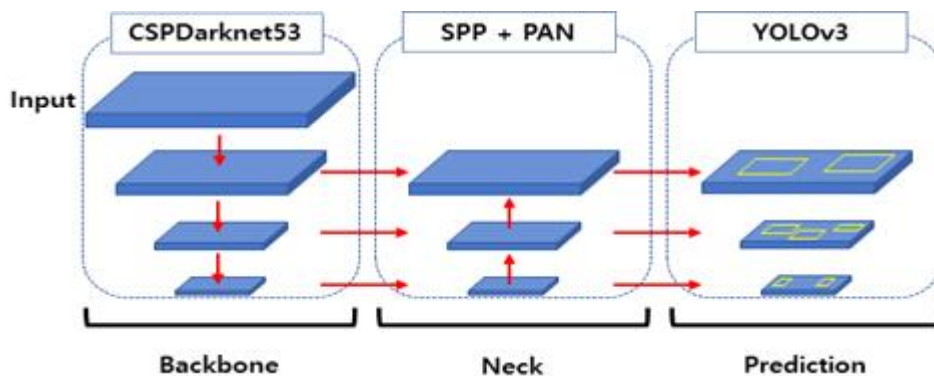


Figure 3. Structure of learning model

YOLO는 객체 인식의 대한 문제를 회귀 문제로 접근하여 객체가 있는 위치 및 크기에 대하여 추출하는 네트워크를 따로 갖지 않는다. 입력 크기에 따른 최종 출력은 격자 형태에 정보를 포함한 형태를 갖고 있는 것이 특징이다. 객체가 있는 위치를 사전에 학습하거나 먼저 연산하는 과정이 제외되어 RCNN 계열의 모델보다 연산 속도가 빠르다는 장점을 갖는다. 또한 YOLOv4에서는 CSPNet을 추가하여 기존의 ResNet과 같은 깊이가 100층 이상의 네트워크에서도 학습 시 전체 연산량을 줄였다. 연산량을 줄임으로 출력되는 속도가 향상되는데 이는 카메라 영상의 FPS 성능을 올렸다. YOLO는 다른 객체 인식 방법들의 모델들 보

다 빠른 속도를 가지며 최종출력을 통하여 각 격자의 정보를 통해 해당 객체의 위치나 크기를 알 수 있다. 따라서 객체의 정보를 직관적으로 알 수 있기에 객체의 실제 위치 추정에 적합하다.

### 3. 제안된 깊이 영상에 따른 객체의 위치 추정 방법

거리에 대한 정보를 나타내는 영상이라고 할 수 있는 깊이 영상은 3차원 공간상의 깊이 정보(Depth Map) 즉 3차원 공간상의 피사체의 거리 값을 갖는 정보이다. 이러한 3차원 깊이 정보는 스테레오 카메라를 이용하거나 레이저 스캐너, TOF(Time of Flight)를 이용하는 방법들이 있다. 본 논문에서는 객체를 인식하고 인식된 객체의 3차원 공간상의 거리와 높이를 추정하기 위하여 스테레오 카메라를 사용하였다.

스테레오 카메라는 사람의 눈과 유사한 방식으로 카메라에 입력된 영상을 통해 관측된 공간상의 거리 및 관측 물체의 형태를 감지 할 수 있다. 스테레오 카메라는 영상에서의 임의의 패턴을 다른 카메라 영상에서 찾아낸다. 이러한 패턴의 관계를 바탕으로 삼각기법을 이용하여 3차원 공간상의 위치를 도출 할 수 있다. 원리는 아래 그림과 같이 임의의 base line의 거리를 두고 위치한 카메라의 영상을 통해 피사체의 변위를 구하고 렌즈의 초점 거리를 통하여 스테레오 카메라로부터 피사체의 거리를 다음 식(1)과 같이 추출한다.

$$1) \text{ distance} = \text{Focal} \frac{b}{\text{diparity}} \quad (1)$$

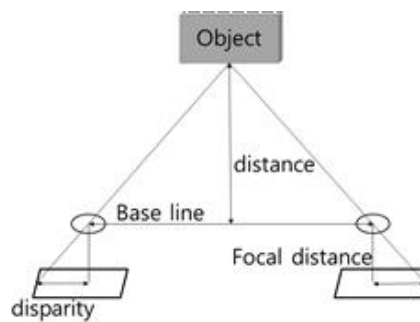


Figure 4. Geometry of the stereo matching system

Distance는 피사체와 카메라의 거리, Base line은 두 카메라 렌즈의 거리를 뜻하며, disparity는 두 개의 이미지에서 피사체 간 거리 차, Focal distance는 렌즈의 초점 길이이다. 이를 통해 카메라 영상을 통해 피사체를 아래 그림과 같이 거리 정보로 표현할 수 있는 것이다.

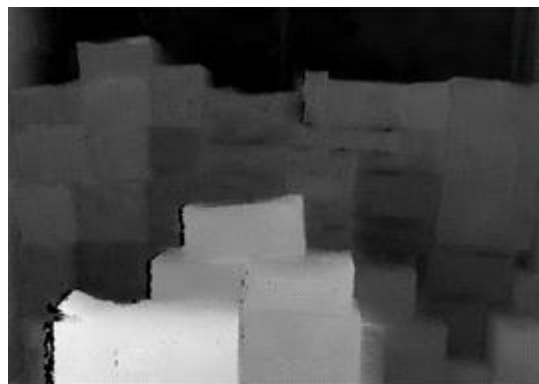


Figure 5. Example of depth map

#### 3.1 깊이 영상 추출 및 위치 계산

본 논문에서는 인식된 박스의 바닥면에서 높이와 카메라가 위치한 평면에서의 직선거리를 추출한다. 스

스테레오 카메라는 지면 또는 이동하는 면에서 높이가 존재하고 카메라를 통해 나타나는 영상도 정확한 수평을 이룰 수 없다. 이러한 요소로 인하여 스테레오 카메라는 영상에 입력된 피사체의 직선거리를 계산하고 도출하게 된다. 도출된 직선거리는 카메라의 위치에서부터 떨어진 거리를 나타내므로 인식된 피사체의 바닥 평면에서 높이, 카메라가 위치한 평면에서 피사체까지의 직선거리는 표현될 수 없다. 본 논문에서는 3차원 공간에서 카메라의 각도와 무관하게 피사체의 수직, 수평 거리를 계산하는 방법에 대해 제안한다. 피사체의 수직, 수평거리를 계산하게 된다면 인식된 객체를 적재하거나 하역 시 기준을 제시할 수 있다. 제안된 방법은 다음 아래 그림과 같이 카메라의 위치를 직교좌표의 원점으로 가정한다.

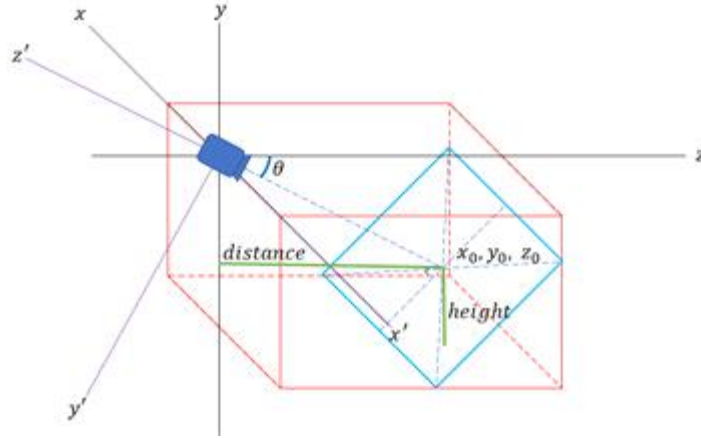


Figure 6. Distance and height extraction using depth information

수식 (2)과 (3) 같이 카메라의 영상의 중심점을 기준으로 카메라의 roll( $\phi$ ), yaw( $\psi$ ) 회전각을 계산하여 카메라 영상의 중심점을 회전 변환을 한다. 회전 변환된 중심점과 pitch( $\theta$ ) 회전각을 이용하여 카메라가 위치한 평면의 법선 벡터를 구한다.

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (2)$$

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} \cos\psi & 0 & \sin\psi \\ 0 & 1 & 0 \\ -\sin\psi & 0 & \cos\psi \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (3)$$

도출된 법선 벡터를 통하여 기준좌표계의 평면의 방정식을 다음 식(4)과 같이 나타낼 수 있다. 평면의 방정식을 통하여 스테레오 카메라를 통해 얻은 공간상의 한 점을 바탕으로 평면의 수직인 직선거리를 다음 식(5)과 같이 나타낼 수 있다.

$$t(x-x_0)+p(y-y_0)+q(z-z_0)=0 \quad (4)$$

$$distance = \frac{|tx_0+py_0+qz_0+d|}{\sqrt{t^2+p^2+q^2}} \quad (5)$$

위 과정과 같이 카메라가 위치한 기준 좌표의 z축으로부터 직선거리를 도출하여 지면에서의 높이 또한 측정가능 하며 높이는 바닥면에서 카메라의 높이의 기준이 있어야한다. 본 논문에서는 이러한 과정을 통해 인식된 객체의 중심점을 기준으로 해당 객체의 위치가 직교좌표계에서 어떤 위치를 갖는지 추정할 수 있다.

### 3.2 인식된 객체의 작업 영역

본 논문에서는 박스를 딥러닝을 통해 인식하여 해당 박스의 위치를 스테레오 카메라 영상의 정보를 통

해 좌표를 계산한다. 이러한 좌표들은 많은 박스가 쌓여있는 물류창고나 택배하역장과 같은 곳에서는 박스가 적층된 위치에 따라 순서대로 하역이 요구되는 자동 하역장비에 적용될 수 있으며 적층된 깊이에 따라 박스를 하역하는데 있어서 작업하고자 하는 위치의 우선순위가 변동되는 시스템에도 적용되어질 수 있다. 또한 제안된 알고리즘의 특징은 딥러닝을 통해 인식된 객체의 거리 및 높이 정보를 이용하여 하역하는 기준에 따라 작업 영역을 구분할 수 있다. 하역하는 기준이라 함은 작업 시 박스를 하역할 수 있는 면적이다. 기준에 따라 박스를 작업하는 공간 영역이 달라지므로 하역 면적에 따라 구분할 수 있는 알고리즘은 다음 흐름도와 같다.



Figure 7. Flowchart of workspace division

연산 순서는 첫째로 딥러닝을 통해 객체인식을 하여 인식된 객체의 이미지상 좌표정보를 통해 카메라에서 직선거리를 추출한다. 추출된 직선거리와 카메라에서 보는 객체의 좌표값을 통하여 객체 중심점의 3.1절에서 언급한 방법으로 객체의 높이와 거리를 추출한다. 다음으로 추출된 거리와 높이 및 이미지상 좌표를 통해 각각의 객체가 갖는 정보를 정렬한다. 객체의 각 거리 즉 카메라가 위치한 평면에서의 직선거리에 따라 내림차순 정렬한다. 정렬된 객체의 거리값을 통해 해당 객체간의 차이를 통해 적재된 위치가 다른 곳을 추출한다. 마지막으로 적재 깊이가 유사한 부분은 산출하여 유사한 부분의 객체들의 높이를 통해 작업할 수 있는 공간 안에 들어오는 높이를 산출하여 최종적으로 작업영역을 산출한다.

## 4. 실험 결과

제안된 방법의 성능을 평가하기 위하여 연구실 규모의 테스트 베드를 구축하였으며 딥러닝 모델의 정확도 평가 및 실험, 인식된 객체의 중심점의 연산된 높이와 거리의 실제 거리와의 오차를 실험 및 계산된 작업영역과 실제 작업영역을 비교하여 진행하였다. 실험에 사용된 박스는 실제 우체국 택배 박스 및 다양한 업체의 박스를 이용했으며 실험실 또는 컨테이너 내부에서 촬영된 이미지가 사용되었으며, 스테레오 카메라 모델은 ZED 스테레오 카메라를 사용하였다.

### 4.1 객체 인식 학습

객체인식 및 분류를 회귀문제로 접근하는 YOLO는 합성곱 형태의 출력을 나타낸다. 이러한 형태는 평균 30FPS(Frame Per Second) 이상의 카메라 영상을 표현할 수 있는 장점을 갖고 있다. 본 논문에는 5,102장의 박스 사진에 대하여 학습을 진행하였다. 학습을 위하여 이미지 데이터셋은 직접 촬영과 구글 이미지를 통해 구축하였다. 데이터셋은 단일 박스 또는 하나의 이미지에 복수의 박스가 존재하는 이미지들을 사용하였다.





Figure 8. Box image used in the experiment

학습 데이터는 전체 데이터셋의 90%(4,592장)를 사용하였으며 평가 데이터셋은 10%(510장)를 사용하였다. 또한 전체 epoch 수는 2,000, 메모리 크기를 고려하여 batch size는 2로 설정하였다. 활성화함수는 Mish를 이용하여 기존의 활성화함수보다 가중치를 갱신하고 Mosaic augmentation 기법을 통해 성능을 개선하였다. 그림 9는 제안된 방법의 박스 인식 정확도를 나타낸다. 실험결과 박스에 대한 단일 객체의 정확도는 약 97%의 성능을 보여주고 있다.

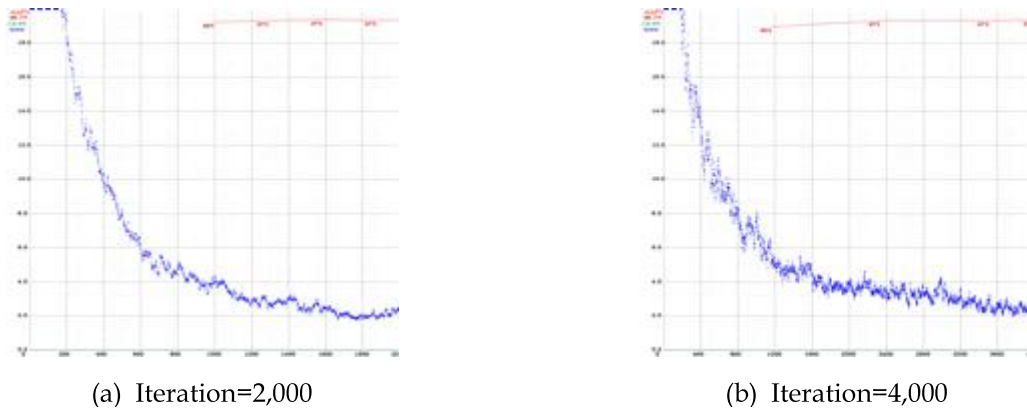


Figure 9. Learning result of box dataset

#### 4.2 스테레오 카메라를 통한 인식된 객체 거리 비교 실험

그림 10과 11은 실제 실험에서 박스를 인식을 결과 영상이며 표1~4는 앞서 학습된 결과를 바탕으로 입력된 영상에서 해당 객체의 중심점을 계산하여 추출된 거리와 높이와 실제 거리와 높이를 비교 하였다. 또한 결과값의 신뢰성을 위해서 6번의 다른 환경에서 각각 8회씩 48회의 실험을 진행하였으며 오차율은 mm 단위까지 인식하도록 하였다. 실험결과 거리에 대한 평균오차는  $\pm 13.6\text{mm}$ , 높이에 대한 평균 오차는  $\pm 12.6\text{mm}$ 로 만족할만한 성능을 보였다.



(a) Pitch angle = 10



(b) Pitch angle = 15

Figure 10. Result of object detection at deferent angle

Table 1. Result of figure 10.(a)

지점	예측 거리	실측 거리	거리 오차	예측 높이	실측 높이	높이 오차
1	2760	2780	-20	1119	1140	-21
2	2770	2755	+15	1709	1698	+11
3	2776	2760	+16	228	245	-17
4	2719	2705	+14	1130	1148	-18
5	2754	2735	+19	685	695	-10
6	2756	2740	+16	216	225	-09
7	2750	2735	+15	773	795	-22
8	2876	2890	-14	1793	1782	+11

Table 2. Result of figure 10.(b)

지점	예측 거리	실측 거리	거리 오차	예측 높이	실측 높이	높이 오차
1	2761	2780	-19	1123	1140	-17
2	2770	2755	+15	1707	1698	+09
3	2775	2760	+15	225	245	-20
4	2715	2705	+10	1131	1148	-17
5	2750	2735	+15	686	695	-09
6	2751	2740	+11	215	225	-10
7	2748	2735	+13	780	795	-15
8	2878	2890	-12	1792	1782	+10



(a) Result of move right to 1m



(b) Result of move left to 1m

Figure 11. Result of object detection at deferent position

Table 3. Result of figure 11.(a)

지점	예측 거리	실측 거리	거리 오차	예측 높이	실측 높이	높이 오차
1	2685	2701	-14	226	240	-14
2	2676	2658	+18	206	218	-12
3	2712	2691	+11	330	311	+19
4	2691	2681	+10	1004	1010	-06
5	2676	2670	+06	883	891	-08
6	2693	2705	-12	1175	1163	+12
7	2701	2708	-09	1431	1442	-11
8	2707	2685	+22	1640	1622	+18

Table 4. Result of figure 11.(b)

지점	예측 거리	실측 거리	오차 거리	예측 높이	실측 높이	오차 거리
1	2688	2701	-13	229	240	-11
2	2673	2658	+15	210	218	-08
3	2683	2691	-08	327	311	+16
4	2665	2681	-16	1001	1010	-09
5	2682	2670	+12	888	891	-03
6	2690	2705	-25	1172	1163	+09
7	2700	2708	-08	1429	1442	-13
8	2704	2685	+19	1638	1622	+16

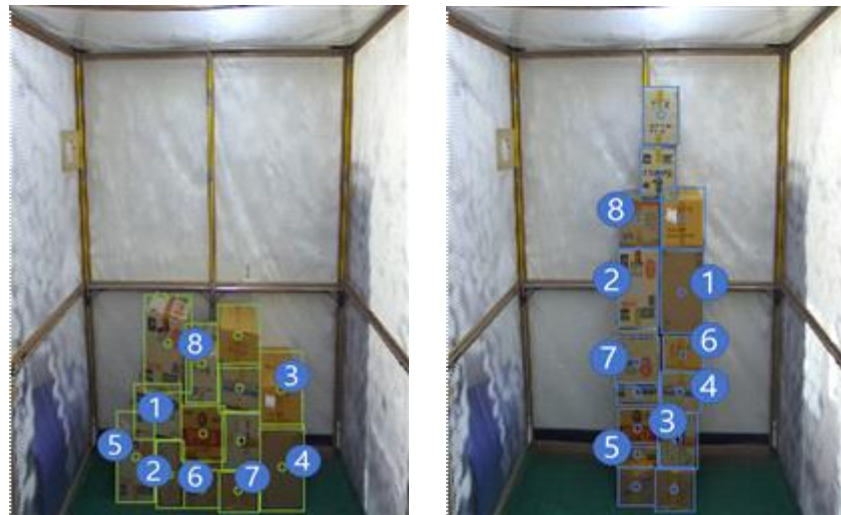


Figure 12. Result of object detection

Table 5. Result of figure 12.

지점	예측 거리	실측 거리	거리 오차	예측 높이	실측 높이	높이 오차
1	2684	2665	+19	602	615	-13
2	2694	2685	+09	230	245	-15
3	2688	2693	-05	718	735	-17
4	2690	2679	+11	328	340	-12
5	2691	2688	+03	394	400	-06
6	2685	2677	+08	189	198	-09
7	2691	2680	+11	187	197	-10
8	2685	2691	-06	828	833	-05

Table 6. Result of figure 13.

지점	예측 거리	실측 거리	거리 오차	예측 높이	실측 높이	높이 오차
1	2781	2765	16	1418	1415	+03
2	2784	2785	1	1395	1397	-02
3	2798	2777	22	555	576	-21
4	2799	2780	19	739	753	-14
5	2792	2775	17	306	323	+17
6	2795	2775	20	1050	1071	-21
7	2798	2785	13	1062	1085	-23
8	2793	2780	13	1613	1608	+05

#### 4.3 작업영역 비교 실험

본 논문에서는 인식된 객체를 바탕으로 400mm\*2,000mm의 로봇 그리퍼의 작업영역을 기준으로 해당 객체들을 하역하는데 필요한 작업 횟수와 중복 영역의 차이를 기준으로 실험을 진행하였다. 실제 박스의 전체 영역을 계산하여 필요한 작업 횟수와 중복 영역을 산출하였다. 딥러닝을 이용한 객체인식을 통하여 도출된 작업영역과 비교를 진행하였다. 아래 그림은 실제 로봇을 작업하기 위한 로봇의 그리퍼 이다.





Figure 14. Real robot gripper shape

컨테이너에서 작업하기 위하여 해당 그리퍼와 동일한 공간을 제작하였으며 제작된 공간과 컨테이너 내부에서의 작업영역을 다음 그림과 같다.



(a) One section

(b) Two sections

Figure 15. Separating the workspace in different situations

## 5. 결론

본 논문에서는 택배박스과 카메라 위치에 따른 직선 거리값을 계산하는 방법을 제안하였다. 제안한 방법은 YOLOv4를 이용하여 개별 택배 박스들을 인식하고 인식된 박스들을 스테레오 카메라에 장착된 IMU 센서를 이용하여 카메라의 수직 평면 위치와 박스들 간의 직선거리를 계산한다. 제안된 알고리즘의 성능을 평가하기 위하여 연구실 규모의 택배 화물 공간을 구축하여 실험을 진행하였으며 실제 택배 박스의 실측 거리와 제안된 알고리즘의 출력 거리값을 비교하였다. 또한 결과값의 신뢰성을 위해서 6번의 다른 환경에서 각각 8회씩 48회의 실험을 진행하였으며 오차율은 mm단위까지 인식하도록 하였다. 실험결과 거리에 대한 평균오차는  $\pm 13.6\text{mm}$ , 높이에 대한 평균 오차는  $\pm 12.6\text{mm}$ 로 만족할만한 성능을 보였다.

## 참고문헌

- Lee, W-H (2020), Strategy for Overcoming Environmental Change Risks Facing the Distribution and Logistics Industry Caused by COVID-19 Pandemic, *Journal&Article Management System*, 23(3), 81-95.
- Soh, S-B, Kim, S-B & Lee, J-P (2020), A Study on the Impact of Product Category on the Expectation of Delivery Service Quality in Online Shopping Malls, *Korean society of supply chain management*, 20(1), 52-66.
- Jiwoong Choi, Dayoung Chun, Hyun Kim, and Hyuk-Jae Lee. (2019), Gaussian YOLOv3: An accurate and fast object detector using localization uncertainty for autonomous driving. *In Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 502-511.
- Cheng-Yang Fu, Mykhailo Shvets, and Alexander C Berg. (2019), RetinaMask: Learning to predict masks improves state-of-the-art single-shot detection for free. *arXiv preprint arXiv:1901.03353*
- Girshick, R. (2015), Fast r-cnn, *In Proceedings of the IEEE international conference on computer vision*, 1440-1448.
- Ren, S., He, K., Girshick, R., & Sun, J. (2015), Faster r-cnn, *Towards real-time object detection with region proposal networks. In Advances in neural information processing systems*, 91-99.
- He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017), Mask r-cnn. *In Proceedings of the IEEE international conference on computer vision*, 2961-2969.
- Won, J-U., Park, M-H., Park, S-W., Cho, J-H., and Kim, Y-T. (2019), Deep Learning Based Cargo Recognition Algorithm for Automatic Cargo Unloading System, *International Journal of Fuzzy Logic and Intelligent Systems*, 29(6), 430-436.
- Alexey Bochkovskiy, Chien-Yao Wang, Hong-Yuan and Mark Liao. (2020), YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv:2004.10934v1 [cs.CV] 23 Apr*
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *In Advances in neural information processing systems*, 1097-1105.
- Zeiler, M. D., & Fergus, R. (2014). Visualizing and understanding convolutional networks. *In European conference on computer vision*, 818-833.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. *In Proceedings of the IEEE conference on computer vision and pattern recognition*, 1-9.
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *In Proceedings of the IEEE conference on computer vision and pattern recognition*, 770-778.

# 온라인 리테일러에서 수요 패턴 예측을 위한 컨볼루션 LSTM 적용 방안

정재원<sup>+</sup>, 임성택, 이승민, 공형준  
한양대학교 산업공학과

## Application of Convolutional LSTM for Predicting Demand Patterns in Online Retailers

J. Jeong<sup>+</sup>, S. Lim, S. Lee, H. Kong  
Department of Industrial Engineering, Hanyang University

Demand pattern forecasting would be surely one of significant modules for successful and profitable on-line retail business. Our work studies the mechanism for demand pattern forecasting at the on-line retailers in a context of e-commerce business. This paper deals with demand pattern forecasting on a basis of time series prediction. We review papers about demand forecasting in a lot of application areas and suggest a new approach to time series prediction using Convolutional LSTM by generating the images for corresponding demand patterns. And it is observed that our proposed approach could elicit more accurate results than the current existing techniques such as seasonal ARIMA and Holt-Winters Technique. Numerical studies using the data for an e-commerce company have been done in order to validate the performance of the suggested demand pattern forecasting mechanism using the Convolutional LSTM. Finally, we provide the meaningful measures ensuring the usefulness and applicability of the proposed demand pattern forecasting mechanism.

**Keywords:** Demand pattern forecasting, On-line retailing, Convolutional LSTM, Customer orders, Order management

### 1. 서론

#### 1.1 연구배경 및 목적

온라인 리테일링 운영에 있어서 고객의 수요패턴 예측은 해당 기업의 비즈니스 프로세스 전반에 지대한 영향을 미치는 중요한 요소이다. 즉, 고객의 수요패턴 예측과정에서 발생하는 예측오차는 해당 기업의 매출에 직/간접적으로 막대한 영향을 미치기 때문이다. 또한, 고객수요 패턴에 정확한 예측은 입고, 적재, 오더 피킹, 출고의 기능을 주로 하는 재고관리 시스템의 최적화를 위해 기본적으로 이루어져야 하는 작업이다(Song and Cho, 2015).

본 연구에서는 온라인 리테일 산업의 재고 관리 및 물류 관리 시스템을 위한 수요 예측에 대해 다룬다. 최근 몇 년간 꾸준한 성장세를 보이던 온라인 리테일 산업은 코로나 사태 이후, 더욱 급격하게 성장하게

되었다. 기존에 주로 오프라인으로 판매되던 많은 품목들이 언택트의 필요성에 따라 상당 부분 온라인 판매 품목으로 유입되었고, 그로 인해 온라인 리테일러는 품목의 다양성과 물량증가에 따라 재고 관리에 어려움을 겪게 되었다. 또한, 온라인 리테일 사업의 성장에 따른 물량증가는 물류시스템의 운영과정에서 물류센터에서의 용량 부족 현상으로 나타나고 있다. 예시로, 대한민국 온라인 리테일 사업의 선두주자라고 할 수 있는 쿠팡은 자사 뉴스룸을 통하여 최근 충청북도, 광주광역시, 경상북도에 대규모 첨단물류센터를 설립하기 위해 부지를 매입하였음을 밝혔고, 다른 온라인 리테일사에서도 비슷한 움직임을 보여주고 있다(news.coupang.com, 2020). 실질적인 운영능력의 확보를 위해서는 사전적으로 고객의 수요패턴에 대한 좀 더 정확한 예측기능이 전제가 되어야 하고, 이러한 예측정확도의 개선은 물류시스템의 물리적인 운영용량이 아니라 실질적인 운영역량으로 연계될 수 있을 것이다. 즉, 고객수요의 사전 예측력 제고는 물류시스템의 물리적인 확장에 앞서서 우선적으로 검토되어야 하는 중요한 과제이다. 최근에는 전통적 수요예측 기법인 Holt-윈터스 기법, 계절성 ARIMA(Seasonal ARIMA) 등보다는 인공지능 기반의 수요예측 기법을 많이 사용하는 추세이다. 인공지능 기반 수요예측 기법은 기존 수요예측 기법에 비해 상대적으로 높은 정확도를 보이며 예측실행시간도 덜 소요된다. 또한 독립변수와 종속변수간의 관계를 나타내는 모델을 설정하지 않고 알고리즘을 구축할 수 있다는 장점이 있다(Jeong and Lim, 2019). 이와 같은 장점들이 있기 때문에 수 년 전부터 구글, 패션브랜드, 카드사 등 전 세계 많은 업체에서 더욱 정확한 예측 결과를 위해 LSTM, GRU 등의 인공지능, 인공지능망 기반의 수요예측기법을 이용해오고 있다(Nenni et al., 2013).

본 연구에서는 수리적인 모델에 기반한 수요패턴 예측기법들이 갖는 한계점을 극복할 수 있는 대안으로 딥러닝 (Deep Learning) 기반의 예측방법론을 제안하고자 한다. 따라서, 본 연구에서는 온라인 리테일러의 과거 고객주문 데이터를 패턴화하기 위하여 판매품목별 시간구간별 수요량을 이미지 형식으로 변환하고, 이러한 이미지를 이용하여 딥러닝 시계열 예측 모형을 적용하고자 한다. 연구에서 사용한 고객주문 데이터는 UCI의 Online Retail 데이터이며, 영국의 온라인 리테일 회사의 2010년 12월부터 2011년 12월까지의 54만여건의 데이터이다(<https://archive.ics.uci.edu/ml/datasets/online+retail>). 이미지화된 데이터에는 상용화된 많은 영상, 이미지 분석·예측 알고리즘을 그대로 적용할 수 있다는 장점이 있다. 우리는 공간정보를 활용할 수 있는 시계열 예측 모형인 컨볼루션 LSTM 에 주문리스트 정보를 이미지 데이터로 변환하여 입력하였을 때, 데이터 안에서 픽셀 간의 관계를 고려하는 이미지 학습의 효과가 수요 예측에도 유의미하게 작용할 것이라는 가정 하에 연구를 수행하였다.

본 연구에서는 전통적 예측 기법과 기존의 인공지능 기반의 수요예측 기법에 대한 논문들을 리뷰하고, 이미지화된 데이터에 컨볼루션 LSTM을 적용하여 수요(판매량)를 예측하는 새로운 방법을 제안하고자 한다. 또한, 온라인 리테일러 회사인 A 회사의 12개월간의 주문데이터에 대해 일별 수요 패턴 예측과 시간대별 수요 패턴 예측을 우리가 제안하는 새로운 방식으로 실험한 결과와 기존의 대표적 수요예측기법인 Holt-윈터스 기법, SARIMA, LSTM, GRU 모형으로 실험한 결과의 RMSE, MAE 등을 비교하여 우리가 제안하는 방법의 예측력 향상에 대해 검증하고자 하였다.

## 1.2 이론적 배경

### 1) Holt-윈터스 모형

Holt-윈터스(Holt-Winters) 모형은 Winters가 제안한 방법으로 기법의 간명성과 단순성에도 불구하고 예측능력 만큼은 ARIMA와 같은 모형들에 결코 뒤지지 않는다고 평가받고 있다(Kim, 2009). 또한, 계절성을 추정할 수 없는 일반적인 지수평활 기법과 달리 예측식과 평활식을 이용하여 계절성을 직접 고려하여 시계열의 미래를 예측할 수 있다. 데이터가 가지고 있는 특성에 따라 활용할 수 있는 Holt-윈터스 기법은 승법(multiplicative) 방법과 가법(additive) 방법이 있다(Jeong, 2004). Holt-윈터스 기법에 사용되는 평활식은 평균수준(level), 추세(trend), 계절성(seasonality) 총 3가지 요인들에 대하여 각각 하나씩 총 3개의 평활식으로 이루어져 있다. 아래에서는 Holt-윈터스 기법에 일반적으로 사용되는 용어들을 정의하고 Holt-윈터스 기법의 종류를 설명한다.

$Y_t$  : t 시점에서 관측된 값  
 $L_t$  : t 시점에서의 시계열 평균수준  
 $b_t$  = 시점 t 에서 시계열 추세성분  
 $S_t$  = 시점 t 에서 시계열 계절성분  
 $F_{t+m} + m$  = 시점 t 에서 예측한 시점 t + m 의 예측값  
 $s$  = 계절성분의 길이  
 $\alpha, \beta, \gamma$  =평활모수

홀트-윈터스 기법은 시계열 데이터가 지닌 계절성의 특성에 따라서 승법적 모형, 가법적 모형으로 나타낸다. 시계열 데이터 전반에 나타나는 계절성(seasonality) 요인이 큰 변동없이 일정하게 나타나는 경우에는 가법적 모형을, 시계열의 수준에 따라서 계절성 요인이 비례하는 경우 승법적 기법을 사용한다.

#### ① 홀트-윈터스 승법적 모형 (Holt-Winters Multiplicative Model)

홀트-윈터스 승법적(multiplicative) 모형에서의 식은 식 (1.1)~(1.4)과 같다.

$$L_t = \alpha \frac{Y_t}{S_{t-s}} + (1 - \alpha)(L_{t-1} + b_{t-1}), \quad (1.1)$$

$$b_t = \beta(L_t - L_{t-1}) + (1 - \beta)b_{t-1}, \quad (1.2)$$

$$S_t = \gamma \frac{Y_t}{L_t} + (1 - \gamma)S_{t-s}, \quad (1.3)$$

$$F_{t+m} = (L_t + b_t m) S_{t-s+m}. \quad (1.4)$$

홀트-윈터스의 승법적 모형은 로그변환을 통해 가법모형으로 변형시킬 수 있다.

#### ② 홀트-윈터스 가법적 모형 (Holt-Winters Additive Model)

홀트-윈터스 가법적(additive) 모형에서의 식은 식 (2.1)~(2.4)과 같다.

$$L_t = \alpha(Y_t - S_{t-s}) + (1 - \alpha)(L_{t-1} + b_{t-1}), \quad (2.1)$$

$$b_t = \beta(L_t - L_{t-1}) + (1 - \beta)b_{t-1}, \quad (2.2)$$

$$S_t = \gamma(Y_t - L_t) + (1 - \gamma)S_{t-s} \quad (2.3)$$

$$F_{t+m} = L_t + b_t m + S_{t-s+m} \quad (2.4)$$

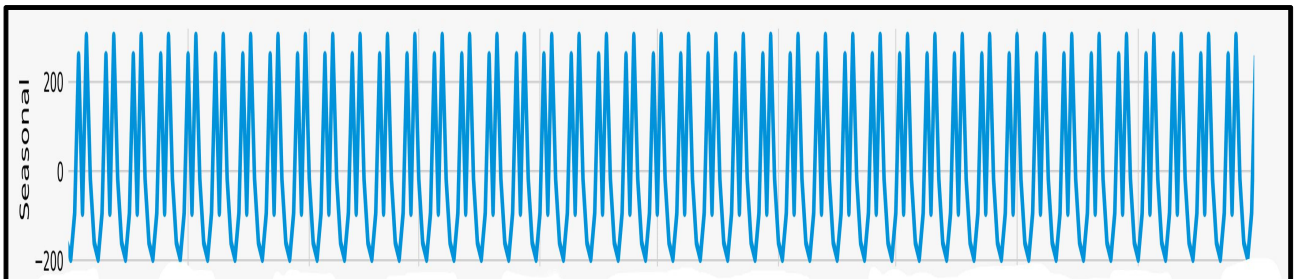


Figure 1 : Seasonality of daily demand of top 77 products



본 연구에서 분석대상이 되는 수요데이터가 계절성이 시계열의 수준에 따라서 비례하는 경우를 가정하여 적용하는 승법적 홀트-윈터스 기법이 더 적합하다고 판단하여 승법적 방법으로 비교군을 생성한다.

## 2) SARIMA (Seasonal Autogressive Integrated Moving Average)

대표적인 확률적 시계열모형에는 AR, MA, 그리고 ARMA 모형이 있다. 이러한 모형은 모두 정상적 시계열 자료, 즉 시점에 관계없이 평균과 분산이 일정한 시계열에 사용한다. 따라서, 비정상 시계열 자료에 대해서는 시계열을 차분하여 정상화하는 ARIMA 모형 등을 이용해야한다. 비정상 시계열  $Y_t$ 를  $d$ 번 차분한 시계열  $y_t$ 를 ARMA(p,q) 모형으로 나타낼 수 있으며, 차분 전 시계열 자료  $Y_t$ 를 적분된 자기회귀 이동평균(Autogressive Integrated Moving Average), 즉 ARIMA(p,d,q) 모형이라고 한다. 일반적으로 ARIMA(p,d,q) 모형은 식(3)와 같이 나타낼 수 있다(Lee and Kwon, 2004).

$$\phi_p(L)(1-L)^d y_t = \theta_q(L)u_t \quad (3)$$

, where

- p = 자기회귀(AR) 항의 차수
- q = 이동평균(MA)의 차수
- d = 정상 시계열이 될 때까지 차분한 횟수
- L = 시차변수(lag operator)
- y = 수요변수
- u = 평균이 0, 분산이  $\sigma^2$ 인 교란변수

일반적으로 본 연구의 실험에서 사용된 데이터와 같이 주기적인 특성을 가지는 시계열 자료에 대해서는 계절성을 고려할 수 있는 Seasonal ARIMA (이하 SARIMA) 모형을 사용하는 것이 적합하다. SARIMA(p, d, q)(P, D, Q)s 모형은 식(4)와 같다.

$$\phi_p(L)\phi_p(L)(1-L)^d(1-L^s)^D y_t = \theta_q(L)\theta_q(L)u_t \quad (4)$$

, where

- s = 분기 수
- D = 계절 단위근(Seasonal Unit Root)의 차수
- P = 계절 자기회귀(Seasonal Autogressive, SAR)항의 차수
- Q = 계절 이동평균(Seasonal Moving Average)항의 차수
- $\theta_q(L)$  = 계절 이동평균항(SMA term)
- $\theta_q(L)$  = 계절 자기회귀항(SAR term)

기존 연구에서 SARIMA 모형은 주로 주기적 특성이나 계절성이 뚜렷한 전력, 교통이나 관광 사업에서의 수요 예측 및 분석에 사용된다. Chang and Liao (2010)의 연구에서는 대만에서 홍콩, 일본, 미국으로 가는 여행객 수의 월별 데이터를 SARIMA를 적용하여 예측하였다. Kim et al.(2014)은 2002년부터 2010년 까지 주간 최대수요 시계열 데이터에 SARIMA 모형을 적용하여 예측결과를 제시하였다. Lee and Kwon (2011)에서는 장래 교통수요 예측에 있어, 국내 교통실적 자료들의 짧은 축적기한으로 인한 데이터 부족을 해결하기 위해 월별 자료를 활용하고, 이에 따라 나타나는 계절성을 고려할 수 있는 SARIMA 모형을 적용하였다. 그러나, ARIMA 또는 SARIMA 모형과 같은 시계열 예측 모델은 기존 데이터의 변동 형태를 파악하여 일정한 패턴이 있는 데이터 값을 예측하기 때문에 데이터가 복잡하거나 비선형성을 가질 때에는 높은 정확도를 기대할 수 없다는 한계점이 있다. 따라서, 복잡하거나 비선형적인 데이터에는 인공지능 기법을 이용한 예측모형을 활용하는 것이 효과적이다 (Lee and Kim, 2020).

### 3) LSTM (Long Short-Term Memory)

LSTM 모형은 Hochreiter and Schmidhuber(1997)에 의해 제안된 모형으로 기존의 RNN(Recurrent Neural Network) 모형에서 장기 의존성을 해결하기 위해 만들어진 RNN의 변형 모델이다. RNN의 은닉층에 cell state 구조를 추가한 것으로, 셀 안의 forget gate와 input gate로 입력 정보에 대해 보존 여부를 업데이트하여 학습한다. 이로 인해 RNN에서 사용하는 정보와 사용 지점의 거리가 길 때 학습능력이 저하되는 단점을 극복할 수 있는 대안으로 사용될 수 있다. 일반적인 LSTM 모형의 구조는 다음의 Figure 2 와 같다.

$$f_t = \sigma(W_f \times [h_{t-1}, x_t] + b_f) \quad (5.1)$$

$$i_t = \sigma(W_i \times [h_{t-1}, x_t] + b_i) \quad (5.2)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \quad (5.3)$$

$$C_t = f_t \times C_{t-1} + i_t \times \tilde{C}_t \quad (5.4)$$

$$\sigma_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (5.5)$$

$$h_t = \sigma_t \times \tanh(C_t) \quad (5.6)$$

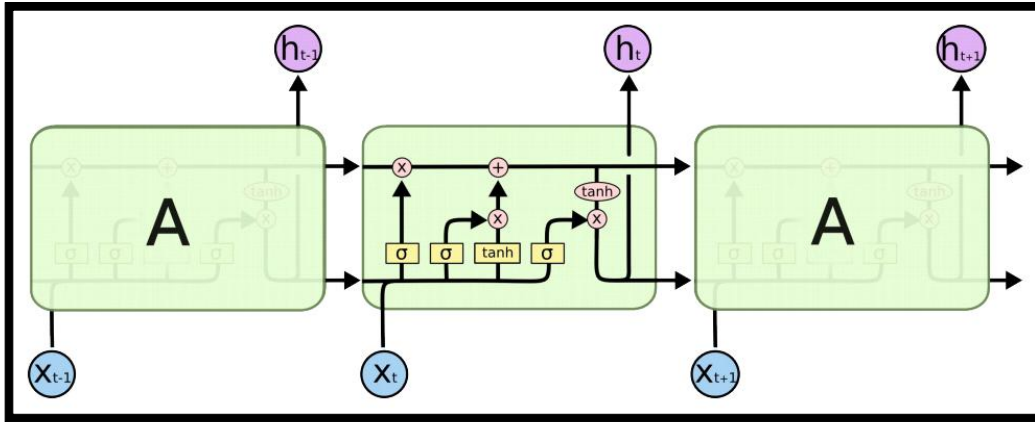


Figure 2. LSTM framework (Olah, 2015)

식 (5.1) ~ (5.6)에서  $W_f$  는 forget gate의 가중치 행렬을 의미하고,  $[h_{t-1}, x_t]$ 는 t-1에서 output이 t에서의 input 으로 반영되는 것을 의미한다.  $f_t$  (forget gate vector)에서는 정보를 제거하는 가중치를 결정하고,  $i_t$  (input gate vector)는 새로운 정보를 획득하는 가중치를 결정한다.  $o_t$ (output gate vector) 는 output의 후보를 결정한다.  $x_t$ 는 입력 벡터,  $h_t$ 는 출력 벡터,  $c_t$ 는 셀 상태 벡터,  $W, U, b$ 는 각각 파라미터 행렬 및 벡터이다(Weng et al., 2019). LSTM 모형은 외부 변수 통합과 자동 특징 추출로 인한 용이성으로 전기사용량, 풍력발전량, 주가, 주택가격 예측 뿐 만 아니라 시퀀스 예측 작업인 음성 인식, 음악 작곡 등 다양한 분야에서 이용된다(Wu et al., 2018). 그러나, 이 모형은 시공간적 특성을 고려하지 못해 이미지나 영상 데이터를 사용할 때는 flatten된 데이터를 사용해야한다는 한계점을 가진다(Ai et al., 2018).

### 4) GRU (Gated Recurrent Unit)

GRU는 LSTM과 같이 게이트의 개념을 이용한 순환 신경망의 일종으로 LSTM의 구조를 단순화하여 연산속도가 LSTM보다 빠르다(Lee et al., 2019). GRU는 LSTM의 망각게이트(forget gate vector,  $f_t$ )와 입력게이트(input gate vector,  $i_t$ )를 갱신게이트( $z_t$ )로 통합하고, 셀 상태( $c_t$ )와 은닉 상태( $h_t$ )를  $h_t$  하나로 통합하

였다 (Kim, 2017). 이처럼 LSTM보다 단순한 구조로 가중치 수가 작으므로 학습이 더 빠르지만, LSTM과 거의 동일한 성능을 보인다(Chaudhuri et al., 2016).

### 1.3 분석 방법론

#### 1) 컨볼류션 LSTM (Convolutional Long Short-Term Memory)

컨볼류션 LSTM (이하 ConvLSTM) 은 FC-LSTM(Fully-Connected Long Short-Term Memory) 을 확장한 모델로 컨볼류션 레이어를 LSTM에 추가한 기법이다(Shi et al., 2015). 기본적으로 FC-LSTM은 셀의 input, output, state가 모두 1차원 벡터이고, FC-LSTM의 기본 구조는 아래의 식 (6.1)~(6.5)과 같다.

$$i_t = \sigma(W_{\xi}X_t + H_{(t-1)} + W_{ci} \circ C_{(t-1)} + b_i) \quad (6.1)$$

$$f_t = \sigma(W_{xf}X_t + W_{hf}H_{t-1} + W_{cf} \circ C_{t-1} + b_f) \quad (6.2)$$

$$C_t = f_t \circ C_{t-1} + i_t \circ \tanh(W_{xc}X_t + W_{hc}H_{t-1} + b_c) \quad (6.3)$$

$$o_t = \sigma(W_{xo}X_t + W_{ho}H_{t-1} + W_{co} \circ C_t + b_o) \quad (6.4)$$

$$H_t = o_t \circ \tanh(C_t) \quad (6.5)$$

이와는 달리 ConvLSTM은 셀의 input, output, state가 3차원 텐서를 가지고 있다. ConvLSTM은 “Input to State”와 “State to State”에 컨볼류션 연산을 사용하는 특징을 가진다. 컨볼류션 연산을 사용함으로써 각 셀에 존재하는 웨이트 수가 FC-LSTM보다 확연히 적어지는 효과를 기대할 수 있다. 다음의 식(7.1)~(7.5)에서의 \* 은 컨볼류션 연산을 의미한다. 이러한 특성을 제외한 이외의 수식은 FC-LSTM과 동일하다.

$$i_t = \sigma(W_{\xi}^*X_t + W_{hi}^*H_{t-1} + W_{ci} \circ C_{t-1} + b_i) \quad (7.1)$$

$$f_t = \sigma(W_{xf}^*X_t + W_{hf}^*H_{t-1} + W_{cf} \circ C_{t-1} + b_f) \quad (7.2)$$

$$C_t = f_t \circ C_{t-1} + i_t \circ \tanh(W_{xc}^*X_t + W_{hc}^*H_{t-1} + b_c) \quad (7.3)$$

$$o_t = \sigma(W_{xo}^*X_t + W_{ho}^*H_{t-1} + W_{co} \circ C_t + b_o) \quad (7.4)$$

$$H_t = o_t \circ \tanh(C_t) \quad (7.5)$$

ConvLSTM 구조는 Figure 3과 같으며 LSTM 셀에서 공간적인 의미와 시간적인 의미를 동시에 반영할 수 있다(Shi et al., 2015).

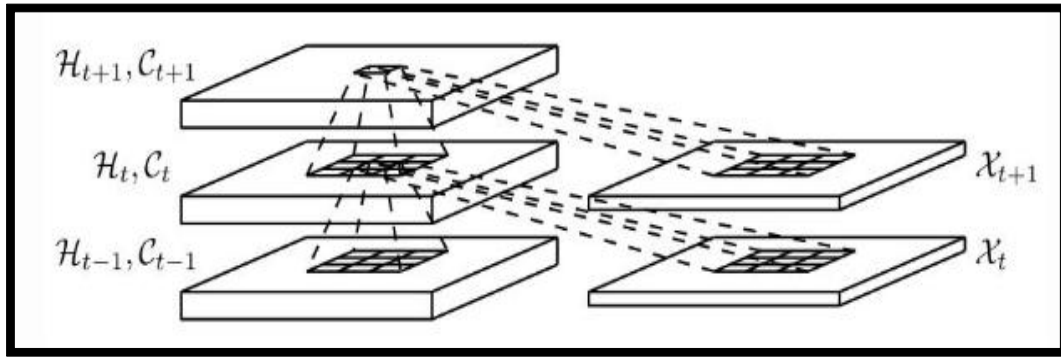


Figure 3. Convolutional LSTM framework (Shi *et al.*, 2015)

## 2. 본론

### 2.1 연구 방안

#### 1) 실험 환경

본 연구에서는 진행된 수치실험은 고객 수요패턴 예측을 위해 적용한 컨볼루션 LSTM의 성능비교를 위하여 동일 데이터를 이용하여 딥러닝 모델(GRU, LSTM)과 시계열 예측 모델(SARIMA, Holt-윈터스 기법)로 구분되었다. 구체적인 실험환경 설정사항으로 딥러닝 모델들의 경우는 Google-Colab의 TPU 런타임에서 Python 3.6.9의 Tensorflow 2.3.0, Keras2.4.3 등의 기계학습 라이브러리를 사용해서 진행되었다. 또한, SARIMA 와 Holt-윈터스 기법의 실험환경은 Python 3.6.9의 Pmdarima, Statsmodel, Pandas 라이브러리와 마이크로소프트 Excel 등을 사용하였다. 수치실험에 사용된 하드웨어 사양은 Intel® Core™ i5-7500, DDR4 RAM 16.0GB 이다.

#### 2) 실험 설계

##### ① 데이터 전처리

본 연구에서는 영국의 온라인 리테일 회사의 2010년 12월부터 2011년 12월까지의 1년 동안의 고객주문 데이터를 이용했으며, 확보된 고객주문건수는 모두 54만 여 건이다. 개략적으로는 공휴일과 주말을 제외하고 실제적으로는 총 305일에 해당하는 영업일 기준의 데이터로 구성되어있다. 데이터 항목체계는 '주문번호', '제품식별코드', '제품설명', '주문수량', '주문시간', '단위가격', '고객ID', 그리고 '주문국가'로 구성되어 있다. 단, 고객 주문시각은 초 단위로 설정되어 있다. 본 연구의 목적에 따라서, 품목별 시간대별 고객 주문 수량을 예측하기 위해 '제품식별코드', '주문시각', '주문수량'만을 선택적으로 발췌하여 실험을 진행하였으며, 데이터 전처리는 1) 이상치 및 결측치 제거, 2) 일자별 구간화, 3) 주문 시간대 별 구간화, 그리고 4) 주문데이터의 이미지화의 순서로 진행되었다. 이상치는 논리적 이상치와 통계적 이상치로 나누어서 처리하였다. 먼저, 논리적 이상치에 해당하는 '해외배송비용', '반품비용', 그리고 '반품처리'를 모두 제거하였다. 또한, 주문 데이터 중 주문 수량에서 상위 0.03%에 해당하는 1,000 이상의 주문수량을 가진 경우를 통계적 이상치로 판단하여 논리적 이상치와 함께 제거하였다. 이러한 품목별 시간구간별 수요정보를 본 연구에서는 각 행(row)은 시간, 열(column)은 품목을 의미하는 2차원 공간상의 이미지 체계를 구성하였다. 이러한 구조에 기반하여 2차원 공간상에 나타나는 각 픽셀의 색상(Color)은 주문수량을 나타낼 수 있도록 이미지를 구성함으로써 시간적 요소와 공간적 요소를 모두 반영할 수 있는 이미지 데이터로 구성하였다. 이를 위한 고객주문 데이터의 이미지화 과정에 적용한 절차는 다음과 같다. 아래 과정을 도식화 한 것은 Appendix a에 첨부하였다.

**[단계-1]** 고객주문 데이터를 시간대 별로 구간화 한다. 이상치와 결측치를 제거한 총 53만 여 건의 데이터에서 가장 주문이 일찍 들어온 시각(오전 7시 40분)과 가장 늦게 들어온 시각(오후 8시 20분) 사이의 데이터를 15분 간격으로 나누어 총 77개 구간으로 구간화한다.

[단계-2] 수요패턴을 이미지로 변환할 품목을 선정한다. 이미지의 row와 column 수를 일치시키기 위하여 총 4082개의 품목 중 주문 수량 상위 77개 품목의 주문 데이터만 활용하였다.

[단계-3] 15분 간격으로 설정된 각 시간구간별 품목들의 주문 수량 데이터를 RGB값으로 변환한다. 주문 수량을 16진법으로 변환시킨 후 1,2번째 자리의 수는 Green값으로, 3,4번째 자리의 수는 Red값으로 설정해주었다.

이와 같은 고객주문 데이터의 이미지 변환 과정을 통해서 시간구간별 고객주문 수량을 RGB값으로 변환함으로써 이미지(77x77x3)로 만들 수 있었다. 고객주문 정보와 이를 이미지 형태로 변환한 예시는 Appendix에 제시하였다. 최종 데이터는 주문일자, 시간대, 품목, 주문수량 순으로 (305x77x77x3) 구조의 4차원 텐서 형태로 구성하였다. 또한, 비교군의 모형과 ConvLSTM 모형에 대해 공통적으로 시계열 데이터의 마지막 28일 중 14일을 Validation 데이터로, 나머지 14일을 Test 데이터로 사용하였으며, 이를 제외한 부분을 Training 데이터로 사용하였다.

## ② 컨볼루션 LSTM 모형 수립

본 연구에서는 Tensorflow 라이브러리의 ConvLSTM2D 함수를 사용해서 ConvLSTM을 구현하였다. ConvLSTM2D의 필터 개수와 커널 크기, 레이어의 깊이를 변화시키면서 모형의 성능을 확인하고, 이를 바탕으로 모형을 수립하였다. 필터 개수는 10, 20, 40, 60, 80개 순으로 증가시켰으며, 커널 크기는 1x1, 2x2, 3x3 순으로 증가시켰고, 레이어의 깊이는 1, 2, 3단으로 증가시켰다. 최종 결과 값을 이미지의 형태로 만들기 위해서 모형의 마지막 레이어는 Conv3D 레이어를 사용하였으며, 필터는 RGB값을 맞춰주기 위해서 3개를 사용하였고, 커널의 크기는 3x3으로 고정하였다.

모형 학습에서는 두 이미지의 차이를 나타내는 지표로 알려진 Binary cross entropy를 Loss로 사용하였으며, 최적화 함수는 adadelat를 사용하였다(Creswel et al., 2017). 검증 데이터에서의 Loss값이 30회 동안 1퍼센트 이상 줄어들지 않으면 훈련을 종료하고 이미지 형태로 나온 최종 결과값을 위에서 설명한 주문 데이터에서 이미지를 만드는 순서의 역순으로 변환해주고, 이를 실제 값과 비교하면서 최적 파라미터를 찾았다. 이를 아래와 같이 도식화할 수 있다.

커널의 크기가 1x1, 2x2인 경우 필터의 개수와 레이어의 깊이와 상관없이 정상적으로 학습이 이루어지지 않았다. 따라서 커널의 크기가 3x3인 경우만 고려하면서 최적 파라미터를 찾았다.

위 실험을 통해서 레이어의 깊이가 깊어질수록 동일한 파라미터의 경우 좋은 성능을 내는 것을 알 수 있었으며, 커널의 크기가 일정 크기 이상인 경우에만 정상적인 학습이 이루어 지는 것을 알 수 있었다. 본 연구의 최종 컨볼루션 LSTM 모형은 레이어를 3단으로 쌓았으며, 각각의 레이어는 필터 개수 80개, 커널 크기 3x3의 파라미터를 가지고, 마지막 레이어는 Conv3D를 사용한다.

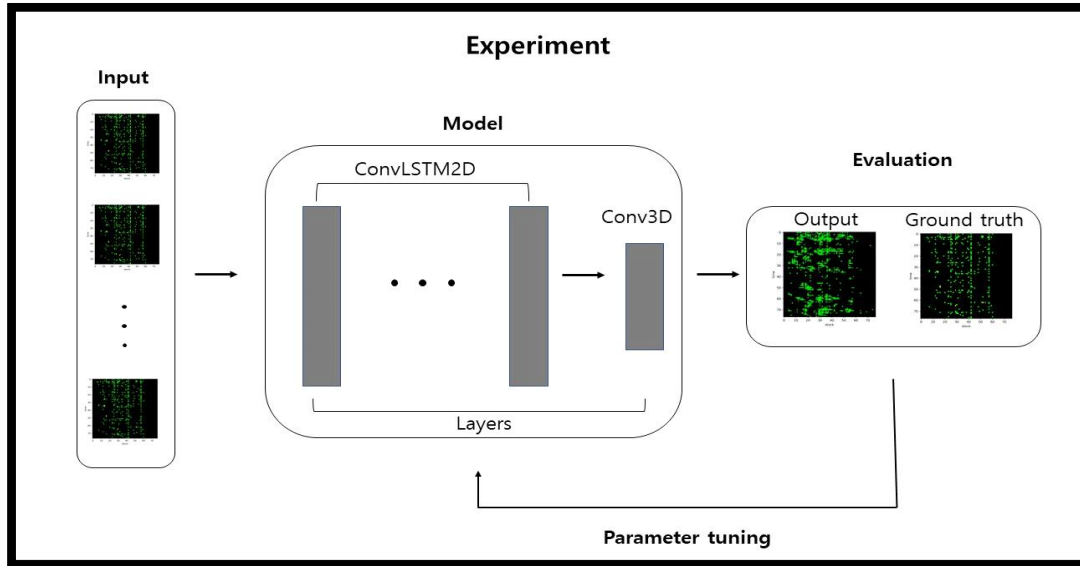


Figure 4. Framework of ConvLSTM for demand prediction

### ③ 비교군(Holt-Winters, SARIMA, LSTM, GRU) 모형 추정 및 수립

비교군 중 SARIMA의 실험은 다음과 같은 과정으로 진행되었다. 일별 예측에서는 Pmdarima의 auto\_arima를 이용하였고, 시간대별 예측은 SARIMAX 라이브러리를 이용하였다. auto\_arima를 이용한 실험에서는 먼저 시계열 데이터를 statsmodel의 seasonal\_decompose 함수로 추세와 계절성 정도를 확인한 후, 파라미터 d와 D의 값은 1로 고정한 뒤, p와 q에 대해서는 0~10의 범위 내에서, P와 Q는 0~3의 범위 내에서 최적의 값을 탐색하여 AIC 값을 최소화하는 최적의 파라미터들을 추정하였다. 이러한 과정을 거쳐 일별 주문량 예측에는 ARIMA(0, 1, 1)(2, 1, 1)<sub>11</sub> 모형을 적용하였다. 시간대별 예측 실험에서는 데이터 특성상 auto\_arima를 이용하여 좋은 결과를 얻기 어려웠기 때문에 일별 예측에서 사용한 파라미터들을 기준으로 변경해가며 실험하였고, 그 중 가장 낮은 RMSE 값을 보이는 값은 일별 예측에서와 동일한 모형인 ARIMA(0, 1, 1)(2, 1, 1)<sub>11</sub> 모형임을 확인할 수 있었다.

홀트-윈터스 기법을 적용하기 위해 계절성의 주기인  $s$ (=계절성분의 길이)를 6일, 7일, 12일, 14일, 30일, 42일로 바꾸어가며 최적의  $s$ 값을 탐색하였다. 또한 그 중 가장 작은 MSE, MAE, MAPE, RMSE 값을 보이는 12일을 계절 성분 주기  $s$ 로 선정하여 모형을 수립하였다. 그리고 홀트-윈터스 기법을 사용하기 위해서는 데이터가 시작되는 초기 구간의 계절성, 추세, 평균수준을 정해주어야 한다. 이때, 첫 번째 주기의  $S_t$ (계절성)은 식 (8.1)과 같다.

$$S_t = \frac{Y_t}{\text{구간 내 } Y_t \text{의 평균}} \quad (8.1)$$

그리고 초기  $L_t$ (평균수준)과  $b_t$ (추세)는 두 번째 주기의 첫 기간에 설정하게 되는데,  $L_1$ (두 번째 구간의 첫 번째 평균수준)은 식 [7.2]과 같이,

$$L_1 = \frac{Y_{1+s}(\text{두 번째 구간의 첫 번째 수요})}{S_1} \quad (8.2)$$

정해주고,  $b_1$ 은 다음 식 [7.3]과 같이 정한다.

$$b_1 = L_1 - \frac{Y_s(\text{첫 번째 구간의 마지막 수요})}{S_s(\text{첫 번째 구간의 마지막 계절성})} \quad (8.3)$$

그 이후 예측을 위해 1.2의 1)에 서술한 홀트-윈터스 승법모형을 이용하여 예측연산을 전개하였다. 그리고 홀트-윈터스 기법에서 사용되는 평활모수  $\alpha, \beta, \gamma$ 는 RMSE를 최소화하는 값으로 최적화하여 실험을 진행하였다.

## 2.2 실험 결과 및 분석

본 실험에서는 적용대상이 되는 상위 77개 품목의 수요합계를 일별 수요패턴 예측과 시간구간별 수요패턴 예측으로 구분하여 진행하였다. 즉, 일별 수요패턴은 77개 품목에 대해서 동일 일자에 발생한 수요량의 합계치를 산정하고, 해당하는 일별 수요량을 예측하는 실험으로 진행하였다. 본 실험에서 적용하는 딥러닝 기반의 예측방식과 시계열 기반의 예측방식에 대한 상대적인 성능 비교를 위하여 수요정보의 집계수준 (Aggregation level)에 따른 적용방법 등의 상대적인 수요예측 성능을 분석하고자 한다. 기존 예측기법과의 비교를 위해 77개 품목 각각에 대해 설정 구간별로 수요량을 예측하고, 이를 모든 품목에 대해 합산하여 구간별 총 예측 수요량으로 예측 정확도를 비교하였다. 또한, 본 연구에서 제안하는 컨볼루션 LSTM기반의 수요예측 방법의 장/단점을 분석하기 위하여 수치실험 조건을 차별화 하였다.

### 1) 일별 수요패턴 예측 (Daily demand pattern prediction)

컨볼루션 LSTM이 일자별 예측에서 MAE기준으로 가장 좋은 성능을 보여주었으며 MAE를 제외한 지표들에서는 SARIMA가 가장 좋은 성능을 보여주었다. Table 1에서 컨볼루션 LSTM과 SARIMA는 다른 방법론들과 비교했을 때, MAPE 지표에서 특히 큰 차이를 보인다. GRU와 LSTM은 지표에서 전반적으로 비슷한 값을 보였으며, 홀트-윈터스 기법의 경우 다른 방법론들과 비교하였을 때 전체적인 지표에서 모두 낮은 성능을 보여주고 있다.

일자별 예측의 경우 큰 오차를 보이고 있는 Holt-Winters 방법을 제외한 4가지 방법론 모두 준수한 오차를 보였으며, Figure 5와 figure 6a ~ 6d 에서 나타나는 것처럼 수요의 패턴을 어느 정도 예측하는 것을 확인할 수 있다. 하지만 ConvLSTM 의 경우 급격한 변화에 민감하게 반응하지만, GRU, LSTM, SARIMA의 경우 급격한 변화를 잘 예측하지 못하고 예측치가 일정한 수준에서 크게 변하지 않는 것을 볼 수 있다. 이는 Table 2 에서도 나타나는데 실제 값과 ConvLSTM은 유사한 값을 가지나, GRU, LSTM, SARIMA 등은 예측의 평균치에서 크게 달라지지 않는다.

Table 1. Performance comparison of daily prediction results

Model	Accuracy	MSE	RMSE	MAE	MAPE
ConvLSTM		3.228570e+06	1796.822	<u>1460.428</u>	37.128
GRU		3.530255e+06	1878.897	1585.552	44.720
LSTM		3.432641e+06	1852.739	1573.307	44.975
SARIMA		<u>3.084248e+06</u>	<u>1756.203</u>	1594.036	<u>35.612</u>
Holt-Winters		5.034231e+06	2243.710	1894.353	53.646

Table 2. Mean, Max, Min, and STD values of daily prediction results

Model	Statistics	Mean	Max	Min	STD
Actual(True Data)		5488.286	8041.000	1948.000	1881.077
ConvLSTM		5388.749	8192.064	1347.813	1802.190
GRU		4764.260	5674.180	3104.324	704.366
LSTM		4832.886	5896.279	3441.449	762.360
SARIMA		4826.330	5788.540	3635.976	662.229
Holt-Winters		5170.456	8685.643	2815.270	1774.032

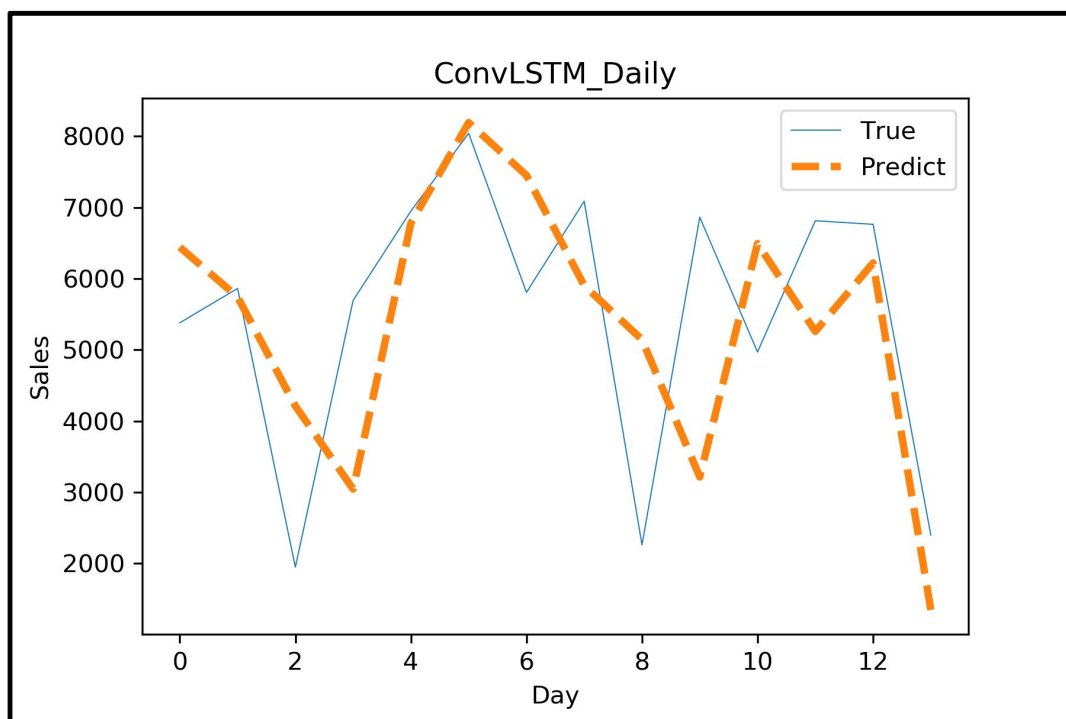


Figure 5. Results of daily prediction by ConvLSTM

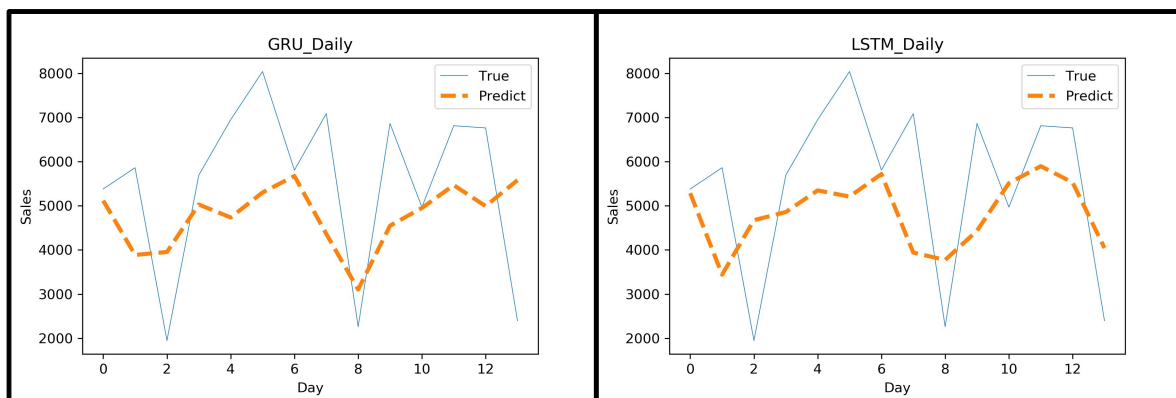


Figure 6a. Results of daily prediction by GRU

Figure 6b. Results of daily prediction by LSTM



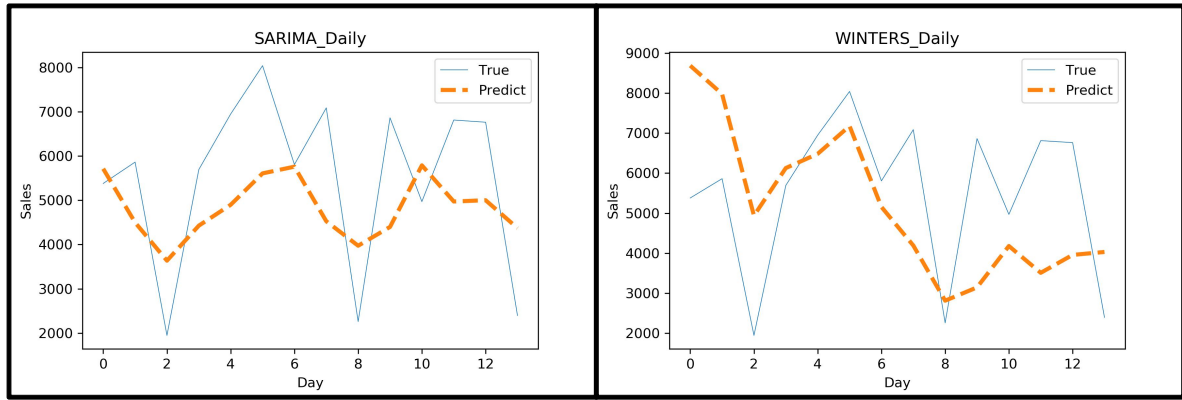


Figure 6c. Results of daily prediction by SARIMA      Figure 6d. Results of daily prediction by Holt-Winters

## 2) 시간구간별 수요패턴 예측 (Hourly demand pattern prediction)

앞서 진행한 일별 수요패턴 예측실험과는 달리 좀 더 세분화된 시간구간 단위별 수요패턴에 대한 예측 실험을 진행하였다. Table 3에서 확인할 수 있듯이 ConvLSTM이 MSE, RMSE, MAE에서 모두 가장 좋은 성능을 보여주었고, 일자별 예측에서 MAE지표를 제외하고 가장 좋은 성능을 보여주었던 SARIMA는 GRU와 LSTM과 비교하였을 때 상대적으로 큰 차이로 낮은 성능을 보여주고 있다. GRU와 LSTM의 성능이 시간대별 수요 예측에서 개선된 것으로 보이며, 이는 시간대별 예측에서는 하루를 13시간으로 보고 1시간 단위로 데이터를 나누었기 때문에 기존의 305일치 데이터에서 3965시간으로 데이터의 양이 많아졌고, 학습 데이터가 많아짐에 따라서 딥러닝 시계열 예측 모형인 GRU와 LSTM의 성능이 SARIMA보다 높게 나온 것으로 추정된다.

Figure 7과 8a~8c 에서 보이는 바와 같이 일자별 수요 데이터 예측과 동일하게 ConvLSTM은 수요의 급격한 변화를 어느 정도 예측하지만, GRU, LSTM, SARIMA에서는 급격한 변화를 예측하지 못하고, 이는 <Table 4>에서도 확인할 수 있다. GRU, SARIMA의 예측 값에는 음수 값이 존재하며, LSTM의 경우 위 두 가지 모델보다는 낮은 오차를 보여주었지만, 실제 데이터에서 주문량이 0인 시점들은 예측하지 못했다. ConvLSTM의 경우 이미지의 RGB값이 0~255 사이의 숫자로 표현되기 때문에 음수 값이 나오지 않았으며, 주문량이 0인 시점을 예측하는 모습을 볼 수 있다.

Table 3. Performance comparison of hourly prediction results

Model	Accuracy	MSE	RMSE	MAE
ConvLSTM		<u>146601.811</u>	<u>382.886</u>	<u>243.581</u>
GRU		159026.510	398.781	270.749
LSTM		150086.307	387.410	271.107
SARIMA		211561.362	459.958	344.083

Table 4. Mean, Max, Min, and STD values of hourly prediction results

model	statistics	Mean	Max	Min	STD
Actual(True Data)		422.176	2236.000	0	457.748
ConvLSTM		<b>414.519</b>	<b>1463.672</b>	<b>0</b>	<b>381.911</b>
GRU		286.941	929.322	-127.195	266.181
LSTM		366.271	800.940	45.019	234.055
SARIMA		342.325	749.677	-387.563	137.723

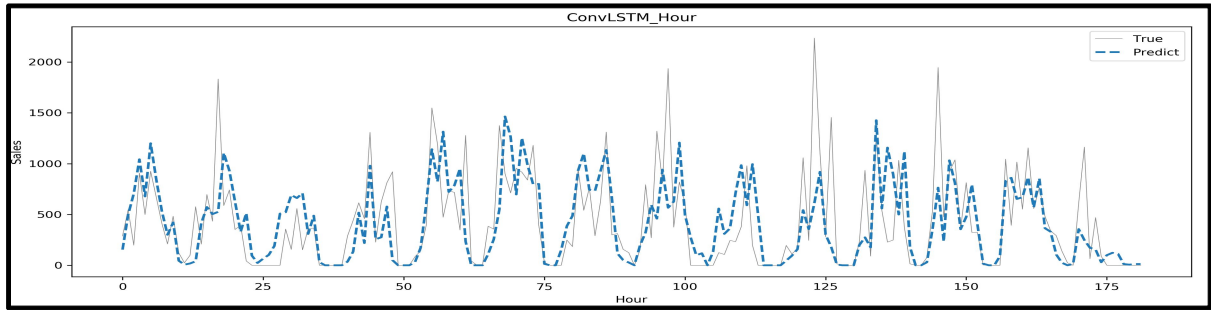


Figure 7. Results of hourly prediction by ConvLSTM

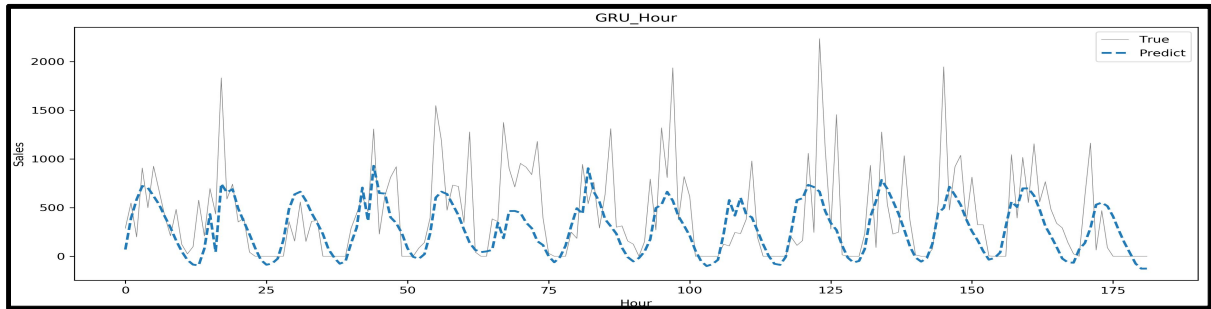


Figure 8a. Results of hourly prediction by GRU

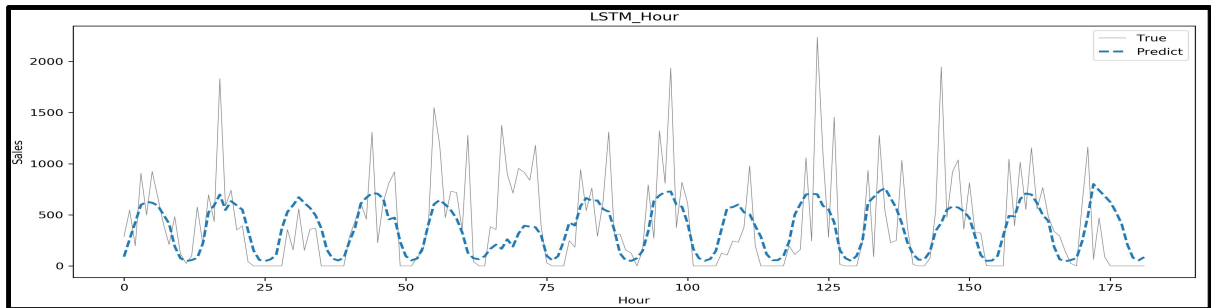


Figure 8b. Results of hourly prediction by LSTM

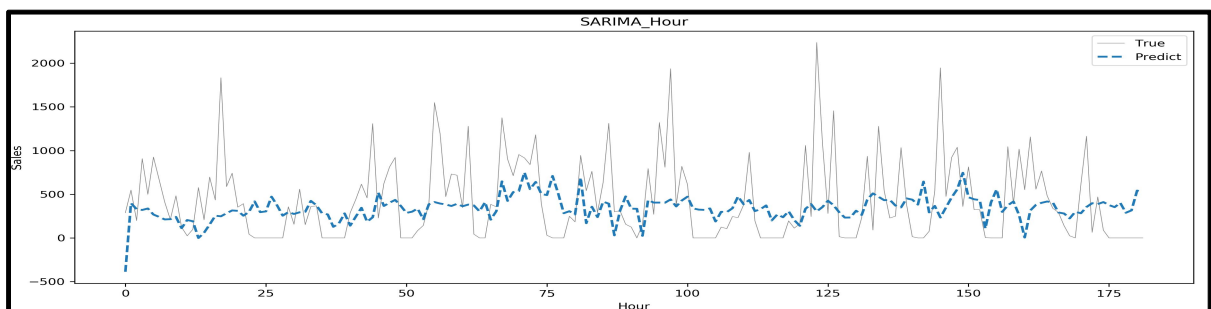


Figure 8c. Results of hourly prediction by SARIMA

### 3. 결론 및 향후 과제

본 연구에서는 온라인 리테일러에게 접수되는 수요 데이터 중 상위 77개 품목에 대한 시간대별 수요패턴의 예측력을 제고하기 위하여 이미지 시계열 분야에서 활용되고 있는 컨볼루션 LSTM 기법을 활용하였다. 즉, 시간적 특성과 품목별 특성으로 정의된 2차원 공간상에서 수요량을 이미지화하는 과정을 통한 해당 기법의 도입 방안을 제시하였다. 연구에서 진행한 구체적인 적용단계는 1) 고객 주문 데이터를 15분 단위로 품목별 합을 계산하고, 2) 구간별 수요를 RGB값으로 대응 시켜 3) 2차원 이미지 형태로 바꾸어 주었고, 4) 이미지화 된 데이터를 통한 수요예측을 진행하였다. 또한, 제안된 방식의 수요패턴 예측성능을 검증

하고자 Holt-윈터스 기법, SARIMA과 함께 딥러닝 시계열 모델인 GRU, LSTM 과의 예측 정확도를 비교하였다.

고객주문 데이터를 이용하여 진행한 일별 수요예측 및 시간대별 수요예측 등 2가지 수치실험 결과에 따르면, 본 연구에서 제안하는 방식의 ConvLSTM 모형의 일별 수요패턴 예측 성능은 기존의 시계열 분석 기법인 SARIMA와 비슷한 수준이었으며, 다른 딥러닝 모델들과도 비슷하거나 약간의 성능 향상이 있었다. 반면에, 시간대 별 고객수요에 대한 예측성능은 고려한 다른 방법/모델들과 비교하였을 때 모든 지표에서 우수한 성능을 보여주었다. 이는 ConvLSTM이 이미지의 시간적인 특성뿐만 아니라 공간적인 특성도 예측에 반영하기 때문인데, 수요 데이터를 이미지화 시키면 품목별 15분간의 주문 수량이 공간적인 특성으로 바뀌어서 상대적으로 우수한 예측성능을 낼 수 있었던 것으로 추정된다. 따라서, 본 연구에서 제안하는 ConvLSTM 기법을 통해서 단기 구간(예: 2시간 이내의 예측 수요패턴)에서의 우수한 수요예측력을 활용한다면 온라인 리테일 회사에서의 단기 구간 수요에 대한 사전준비를 통해서 물류시스템 실행력 제고가 가능할 것으로 예상된다. 구체적으로는 실시간 수요에 대한 예측력에 기반하여 고객수요 충족을 위한 대응력을 향상하고, 물류시스템 운영자원의 효율적인 운영에도 의미 있는 개선을 도출할 수 있을 것으로 기대된다. 또한 본 연구에서는 비교군과의 비교를 위하여 불가피하게 품목별로 예측한 결과를 전체 품목에 대해 합산하여 예측 정확도를 평가하였지만, 실험 중 확인한 결과 품목별 수요패턴 예측에서도 높은 성능을 보여주었다. 이를 활용하면 품목별 실시간 수요에 대한 예측도 가능할 것으로 예상된다.

본 연구에서는 온라인 리테일 회사의 1년 치 데이터를 사용하였는데, 딥러닝 모델의 특성상 많은 수의 파라미터를 추정하기 때문에 데이터가 적은 경우 파라미터를 적절하게 추정하지 못한다. 이는 일자별 수요예측과 시간대별 수요예측의 성능 비교에서도 확인할 수 있는데, 일자별 수요예측의 경우 305개의 데이터를 사용했고, 시간대별 수요예측은 3965개의 데이터를 사용하였기 때문에 일자별 수요예측에서 SARIMA에 비해서 낮은 성능을 보이던 GRU와 LSTM이 시간대별 수요예측에서 높은 성능을 보여주었다. ConvLSTM의 경우 하루를 하나의 이미지로 만든 데이터를 사용하였고, 이미지 데이터 안에 시간대별 수요가 포함되어 있기 때문에 시간대별 예측에서도 동일한 305개의 데이터만 사용할 수 있었다. 따라서 추후 더 많은 양의 데이터를 사용해서 학습한다면 보다 더 높은 성능을 보일 수 있을 것으로 예상된다. 또한 위의 <데이터 전처리> 부분에서 밝힌 바와 같이, 본 연구에서 사용된 원본 데이터에는 총 4082개의 품목이 있었지만 15분 단위로 품목별 주문 수량을 합해주면 77개의 구간이 나오기 때문에 행의 개수와 열의 개수가 일치하도록 하기 위하여 상위 77개의 품목만을 예측에 사용하였다. 구간을 더 작은 단위의 시간으로 나누고, 더 많은 품목을 사용하여 예측을 실시한다면 더 좋은 성능을 낼 수 있을 것으로 기대한다. 또한 본 연구에서는 실험 환경의 메모리 제약으로 커널의 크기를  $3 \times 3$ 보다 크게 하거나 레이어의 깊이를 4단 이상으로 할 경우 실험을 진행할 수 없었다. 따라서 파라미터를 다르게 설정함으로써 모델이 향상된 성능을 낼 가능성이 있음을 기대한다.

본 연구에서 사용된 이미지 시계열 예측 기법인 ConvLSTM은 Shi et al.(2015)의 연구에서도 지적인 예측 단위가 멀어질수록 이미지에 노이즈가 발생하는 한계점이 있다. 본 연구의 실험에서도 위와 같은 현상을 확인할 수 있었으며, 이는 예측일자가 멀어지면 Red와 Green만으로 표현된 데이터에서 Blue값이 조금씩 나오는 형태로 나타났다. 본 연구에서는 이를 해결하기 위해서 예측일자를 바로 다음 일자로 설정하였으나, 노이즈를 해결하고 예측일자를 늘리기 위해서는 추후 논의가 필요하다.

## 참고문헌

- Creswel, A. Arulkumaran, K. Bharath, A. A.(2017), On denoising autoencoders trained to minimise binary cross-entropy, arXiv
- Olah, C.(2015), Understanding LSTM Networks, <http://colah.github.io/posts/2015-08-Understanding-LSTMs/>
- Jung, D(2004), Prediction of the number of domestic breeding hog Using Winters Model, Kyung-sung University, Republic of Korea
- Lee, H. Kim, K. Jeong, H. Lee, H. Kim, H. Park, J.(2020), A Study on Wind Power Forecasting Using LSTM Method, The transactions of The Korean Institute of Electrical Engineers, 69(8), 1157-1164
- Kim, H.(2017), Prediction of Time Series Data Using LSTM/GRU Recurrent Neural Network, Korea National Open University, Republic of Korea
- Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. Neural Computation, 9(8), 1735-1780
- Jeong, H. , Lim, C.(2019), A review of artificial intelligence based demand forecasting techniques, The Korean Journal of Applied Statistics, 32(6), 795-835
- Song, I., Cho, N.(2015), Forecasting based Optimal Storage Policy Model of a Warehouse Management System, Journal of the Korean Society of Supply Chain Management, 15(1), 1-7
- Lee, J., Kwon, Y.(2011), A Study on Dynamic Change of Transportation Demand Using Seasonal ARIMA Model, Journal of Korean Society of Transportation, 29(5), 139-155
- Jong-Tae-Kim(2009), Forecasting number of student by Holt-Winters additive model, Journal of the Korean Data And Information Science Society, 20(4), 685-694
- Nenni, M. E., Giustiniano, L., Pirolo, L.(2013) Demand forecasting in the fashion industry: A review, International Journal of Engineering Business Management
- Lee, S., Sun, Y., Lee, J., Lee, D., Cho, E., Park, D., Kim, Y., Sim, I., Kim, J.(2019), Short-term Power Consumption Forecasting Based on IoT Power Meter with LSTM and GRU Deep Learning, The Journal of The Institute of Internet, Broadcasting and Communication, 19(5), 79-85
- Kim, S., Jung, H., Park, J., Baek, S., Kim, W., Chon, K., Song, K.(2014), Weekly Maximum Electric Load Forecasting for 104 Weeks by Seasonal ARIMA Model, Journal of the Korean Institute of Illuminating and Electrical Installation Engineers, 28(1), 50-56
- Chaudhuri, T. , Ghosh, I.(2016), Artificial Neural Network and Time Series Modeling Based Approach to Forecasting the Exchange Rate in a Multivariate Framework, Journal of Insurance and Financial Management, 1(5), 92-123
- Weng, T., Liu, W., Xiao, J.(2019), Supply chain sales forecasting based on lightGBM and LSTM combination model, Industrial Management & Data Systems, 120(2), 265-279
- Shi, X., Chen, Z., Wang, H., Yeung, D.(2015), Convolutional LSTM network: A machine learning approach for precipitation nowcasting, Advances in neural information processing systems 28(NIPS2015), 802-810
- Ai, Y., Li, Z., Gan, M., Zhang, Y., Yu, D., Chen, W., Ju, Y.(2018), A deep learning approach on short-term spatiotemporal distribution forecasting of dockless bike-sharing system, Neural Computing and Applications, 31, 1665-1677

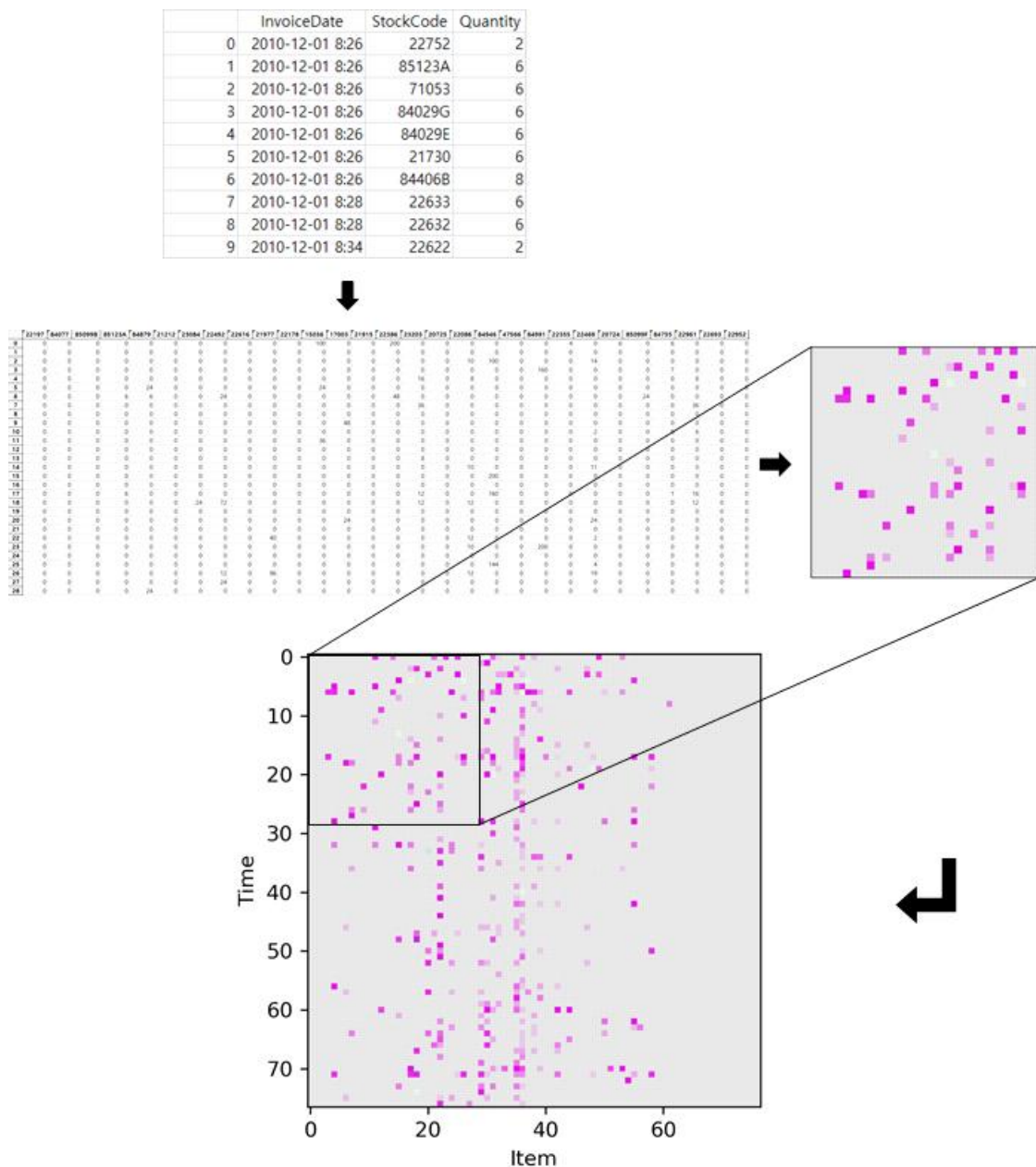
Chang, Y., Liao, M.(2010), *A Seasonal ARIMA Model of Tourism Forecasting: The Case of Taiwan*, *Asia Pacific Journal of Tourism Research*, 15(2), 215-221

Wu, Y. , Wu, Q. , Zhu, J.(2019), *Improved EEMD-based crude oil price forecasting using LSTM networks*, *Physica A: Statistical Mechanics and its Applications*, 516, 114-124

Coupang(2020), *Established a cutting-edge distribution center in Gimcheon-si*, <https://news.coupang.com/archives/5058>

UCI(2015), *Online Retail Data Set*, <https://archive.ics.uci.edu/ml/datasets/online+retail>

## Appendix



Appendix a. Process of generating the images for corresponding demand patterns

# 이열 배치설계를 위한 Cuckoo Search 기반의 최적화 알고리즘 개발

윤영민<sup>1</sup>, 채준재<sup>1,†</sup>

<sup>1</sup>한국항공대학교 항공교통물류학부

## Development of Cuckoo Search Based Optimization Approach for a Double Row Layout Problem

Yeongmin Yun<sup>1</sup>, junjae Chae<sup>1†</sup>

<sup>1</sup>School of Air Transport, Transportation and Logistics, Korea Aerospace University

Effective layout design can improve productivity and manufacturing efficiency. Given a number of departments with known lengths and the flow cost between any two departments, a double row layout problem(DRLP) determines an arrangement of departments on either side of a central corridor so that the total material moved flow is minimized when the material handling path is the straight line along the corridor. DRLP arises in many other real-world applications including manufacturing system. This paper presents a approach in cuckoo search algorithm(CS) for obtaining good solutions for the DRLP. Random-key encoding scheme is applied to CS for representing a solution of DRLP and enabling Lévy flights to perform on DRLP. To evaluate the approach, computational experiments are conducted from benchmark problems. the obtained results show the better performance of proposed approach then a previous study.

**Keywords:** Facility layout, Double row layout problem, Meta-heuristic, Cuckoo Search

### 1. 서론

배치설계는 물류센터 나 생산시설에서 개체들의 상대적인 위치를 최적으로 배치하는 것을 목적으로 하는 전형적인 조합 최적화 문제로서, 문제의 크기가 커질수록 최적 값을 구하는데 걸리는 시간이 기하급수적으로 증가하는 NP-hard 문제이다(Garey and Johnson, 1979). 여기서 개체는 물리적인 형태의 설비뿐만 아니라 물류시설의 보관공간이나 제조시설의 생산공간과 같은 무형의 서비스 공간을 포함한 개념이다(Chae and Regan, 2020). 많은 연구 들에서 개체의 기본적인 형태는 직사각형의 형태로 가정한다(Chae and Regan, 2016; Kang and Chae, 2017). 또한, 배치설계에서 개체들은 기본적으로 주어진 공간안에서 겹쳐지지 않게 배치되어야 한다(Ingole and Singh, 2017). 배치설계에서 최적에 대한 기준은 다양한데 각 개체들 간의 자원의 흐름, 즉 물동량을 고려한 개체간 물류 이동 거리의 총합을 최소화하는 것을 일반적으로 많이 사용한다. 본 연구에서는 지금까지의 연구 들에서 일반적으로 가정한 것처럼 개체를 직사각형의 형태로 가정하였으며, 배치에 대한 최적의 기준 역시 총 물류 이동 거리의 합으로 가정하였다.

일반적으로 시설계획(Facility planning)에 있어서 효율적인 배치설계는 매우 중요하다(Liu et al., 2020). 물류센터나 생산시설 내에서 설비나 공간에 대한 배치형태는 작업자나 AGV(Automated Guided Vehicle)의 작업 동선과 같은 운영 효율성과 연관된 요소들에 많은 영향을 준다(Tompkins et al., 2010). 또한 제조

과정에서 생산비용의 15~70%은 개체의 배치형태와 관련되어 있으며, 효율적인 배치설계는 시설내 생산작업 운영비용을 10~30%까지 줄일 수 있다(Braglia et al., 2003).

현대에 와서 변화되고 있는 시장환경을 보면, 제품의 종류가 다양해지고 있어 각 제품의 수요량을 예측하기가 점점 더 어려워지고 있으며 제품의 수명 주기 또한 짧아지고 있다. 이러한 흐름에 따라 생산제조시설에서의 한 가지 제품라인은 더 이상 효율성이 없고 설비나 장비의 교체 없이 설비나 제조공간의 재배치를 통한 한 제품라인에서 다른 제품 라인으로의 빠른 변화가 요구되고 있다(Benjaafar and Sheikhzadeh, 2000). 배치설계가 유동적으로 변경 가능해지면 다양해지는 제품의 종류와 짧아지는 제품의 수명에 따라 배치를 효율적으로 바꿀 수 있어 생산성과 제조 효율성을 증가시킬 수 있다.

배치설계에 관련된 연구는 예전부터 오래도록 연구가 진행되어 왔지만 최근 배치설계의 효율화가 중요해진 시점에서 더욱 필요한 연구라고 할 수 있으며, 오랜 시간 들여 한 번 좋은 배치를 찾으면 끝이던 과거와는 달리, 갈수록 제품의 수명주기가 짧아지고 있고 제품의 종류가 다양해지고 있어 빠른 시간안에 효율적인 배치를 찾을 수 있는 연구가 필요한 상황이다.

여러 배치설계 유형 중 ‘열’ 배치설계(row layout problems)는 개체를 줄지어 평행하게 배치하는 유형이다. 그리고 개체를 두 줄로 평행하게 배치할 때 해당 배치형태를 이열 배치라고 한다(Ahonen et al., 2014; Amaral, 2012a, 2013a, 2020; S. Wang et al., 2015; Zuo et al., 2014, 2016). 이열 배치설계는 배치해야 할 개체들의 길이(length)와 각 개체들 간의 물동량(flow)이 주어졌을 때, 개체들 간 물류 이동 거리의 총합을 최소화하도록 하는 개체들의 이열 배치를 찾는 문제이다. 이열 배치설계는 대표적으로 AGV와 같은, 개체들 사이에서 자원을 운반하는 운송체가 존재하는 제조 유연 생산 시스템내 개체들에 대한 배치설계에 주로 적용된다(Heragu and Kusiak, 1988; Tubaileh and Siam, 2017). 이때, <Figure 1>에서 볼 수 있듯이 개체들은 AGV가 작업을 위해 움직이는 경로 및 동선의 위쪽 또는 아래쪽에 배치되며, 설비내 자원의 입출력 지점은 각 설비의 AGV 동선 방향 쪽 중간 지점으로 설정한다. 중앙의 AGV의 경로는 일반적으로 작업 동선의 효율성을 높이기 위해 일직선의 형태로 설정된다. 그리고 이열 배치설계는 FMS 내에서 뿐만 아니라, LCD 생산라인(Chung and Tanchoco, 2010), 오피스나 병원과 같은 다양한 건물 내의 공간에 대한 배치(Ahonen et al., 2014), 반도체 제조공정내 설비배치(S.Wang et al., 2015; Zuo et al., 2014) 등 다양한 실제 업무 환경에서 많이 적용되고 있다.

이와 같이 이열 배치설계는 실제 업무 환경에서 많이 활용되고 있는 문제이며 기본적인 배치형태이기 때문에 다른 배치유형으로의 확장이 용이할 것으로 판단하여 본 연구에서는 이열 배치설계에 대해 다루고자 하였다.

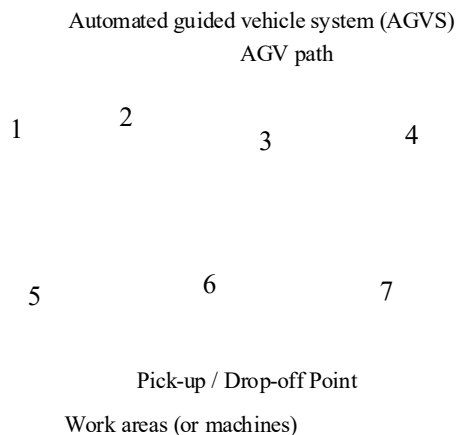


Figure 1. Double row layout problem

하지만 이열 배치설계 역시 배치설계의 한 종류이므로 NP-Hard 조합최적화 문제이다(Amaral, 2012a, 2013b; Secchin and Amaral, 2019). 그렇기 때문에 배치해야 할 개체의 수가 늘어날수록 문제를 풀기위한

계산시간은 기하급수적으로 증가한다(Chae and Regan, 2020; Secchin and Amaral, 2019). 따라서 개체 수가 많아질수록 분단탐색법과 같은 정확 기법을 통해 문제를 해결하기에는 시간적인 한계가 존재한다. 그렇기 때문에 많은 연구들에서 일반적으로 규모가 큰 문제들을 빠른 시간 내에 해결하기 위해 다양한 휴리스틱 알고리즘과 메타 휴리스틱 방법론이 개발되고 적용되어 왔다.

본 연구에서는 이열 배치설계에 대한 해결 방법론으로 메타 휴리스틱 알고리즘으로는 비교적 최근에 고안된 Cuckoo Search(CS) 알고리즘을 적용하였다. CS 알고리즘은 일부 빠꾸기 종의 특이한 번식행위를 모방하여 만들어졌다(X. S. Yang and Deb, 2009). 몇몇의 빠꾸기 종들은 다른 새의 둥지에 알을 낳아 새끼를 부화시키는 "타란"이라는 행위를 통해 번식하는데 만약 둥지의 주인이 이를 알아채면, 빠꾸기의 알을 버리거나 자신의 둥지를 버리고 다른 곳으로 이동해 새로 둥지를 만든다. 그렇게 되면 빠꾸기의 알은 버려져 생존율이 떨어지기 때문에 주인 새의 알과 색깔 및 모양을 비슷하게 만들어 알이 버려지는 확률을 낮춰 생존율을 높인다. CS 알고리즘은 그동안 많은 연구들에서 다양한 최적화 문제들에 적용되었으며 다른 최적화 알고리즘들과 비교 시에도 CS 알고리즘이 매우 효율적이라는 것을 보여주었다(Civicioglu and Besdok, 2013; Gandomi et al., 2012, 2013; Srivastava et al., 2012; Walton et al., 2011; X. S. Yang and Deb, 2010, 2014). 하지만 CS 알고리즘은 해를 탐색하는 영역 즉, 가능해(feasible solutions)의 영역이 연속적인 연속 최적화 문제에 맞춰 고안된 메타 휴리스틱 중 하나인데, 배치설계와 같은 조합 최적화 문제는 가능해의 영역이 이산 집합에 속하거나 이산적(discrete)인 이산 최적화 문제이기 때문에 CS 알고리즘과 같은 메타 휴리스틱을 조합 최적화 문제에 적용하게 되면 문제 복잡성이 증가하는 문제가 발생한다(Kang et al., 2018). 이와 같은 문제로 인해 CS 알고리즘을 배치설계를 포함한 조합 최적화 문제에 적용한 연구는 지금까지 많지 않았다.

CS 같은 메타 휴리스틱을 배치설계와 같은 조합 최적화 문제에 적용하기 위해서는 메타 휴리스틱의 좋은 성능은 잃지 않으면서 적절하게 조합 최적화 문제에 적용시키는 것이 중요하다(Ouaarab et al., 2015b). 특히, CS 알고리즘은 Lévy flight라는 랜덤 워크 모델을 통해 새로운 해를 탐색하는데, Lévy flight는 결과 값으로 특정 실수범위의 상수 값을 도출하기 때문에 Lévy flight 상수 값으로 이열 배치설계의 새로운 해를 이산적으로 탐색하기 위한 방법이 필요하다. 따라서 본 연구에서는 이열 배치설계의 해를 표현하고 Lévy flight 상수 값의 적용 방법으로 Random-key 방식을 사용했다.

본 논문의 다음 장 구성은 다음과 같다. 2장에서는 이열 배치설계와 Cuckoo search(CS)알고리즘에 대한 기존 연구에 대해 정리하고, 3장에서는 본 논문에서 다루고자 하는 이열 배치설계의 수리모형에 대해, 그리고 4장에서는 물류 비용이 최소화되도록 하는 좋은 이열 배치형태를 찾기 위해 본 연구에서 적용한 방법론에 대해 설명한다. 5장에서는 본 연구에서 제안한 방법론의 적용에 대한 실험 결과를 보이며, 이전 연구 결과와의 비교를 통해 분석한 내용을 담았다. 마지막 6장에서는 본 논문에 대한 결론과 함께 향후의 연구과제를 제시하였다.

## 2. 기존 문헌 연구

배치설계와 관련된 연구들은 지난 수십년 전부터 지금까지 많은 연구가 진행되었고, 배치설계를 해결하기 위한 모델과 방법론 또한 다양하게 제시되어 왔으며 이에 대해 다음과 같은 많은 연구 들에서 정리되어 졌다(Drira et al., 2007; Heragu and Kusiak, 1988; Levary and Kalchik, 1985; Meller and Gau, 1996; Singh and Sharma, 2006). 그리고 다양한 방법론들은 크게 분단탐색법(Branch and bound method)과 같은 정확 기법(Exact methods)과 휴리스틱 접근법이나 메타 휴리스틱과 같은 근사적 기법(Approximated approaches)으로 분류될 수 있다(Drira et al., 2007). 하지만 배치설계는 서론에서 설명한 것과 같이, NP-hard 문제라 문제의 규모가 커질수록 최적 값을 찾기 어렵기 때문에 정확 기법으로 해결이 가능한 작은 규모의 문제들에 대해서만 정확 기법을 적용한 연구들이 진행되어져 왔으며(Chae and Regan, 2020), 배치설계와 관련된 대부분의 연구들은 근사적 기법을 적용해왔다(Chae and Regan, 2020; Garcia-Hernandez et al., 2019; Kang and Chae, 2017; Kim and Chae, 2019; Scalia et al., 2019).



또한, 배치설계는 매우 다양한 기준으로 분류될 수 있는데 개체들이 배치되는 형태나 방식(Layout configuration)에 따라서도 다양한 유형들이 존재한다. Drira et al.,(2007)에 따르면, 배치설계는 배치 형태에 따라 일렬 배치(single-row), 다열 배치(multi-row), 루프 배치(loop), 오픈필드(open field), 다층 배치(multi-floor)로 분류된다. 본 연구에서 다루는 이열 배치형태(double-row)는 제시된 분류에 속하지는 않지만 복도의 한쪽에만 개체를 배치하는 일렬 배치 형태와 달리, 중앙 복도의 양쪽으로 배치하므로 복도의 한쪽에만 개체를 배치하는 형태에서 복도의 양쪽에 배치한다는 의미에서 이열 배치설계를 일렬 배치설계의 확장된 문제 형태라고 볼 수 있다(Chae and Regan, 2020).

이열 배치설계(DRLP)는 서론에서 밝힌 것과 같이 실제 업무 환경에서 많이 활용되고 있는 문제지만 이 전부터 연구가 많이 진행되어왔던 일렬 배치설계와 달리, 이열 배치설계는 비교적 최근부터 연구가 진행되기 시작했으며, 일렬 배치설계에 비해 연구에 대한 관심이 제한적이었다(Guan et al., 2020). 그렇기 때문에 이열 배치설계 관련 연구들에 대해서는 정확 기법과 근사 기법으로 구분하지 않고 각 연구들에 대해 조금 더 자세히 살펴보았다.

Chung and Tanchoco(2010)은 Heragu and Kusiak(1991)가 제시하고 Amaral(2006)의 연구에서 개선시킨 SRLP에 대한 MIP 수리 모형을 확장시켜 처음으로 DRLP에 대한 MIP 수리 모형을 제시하였고, 제시된 수리 모형을 통해 정의된 DRLP를 풀기 위해 5개의 휴리스틱 알고리즘을 고안하고 적용하였다. Zhang and Murray(2012)은 Chung and Tanchoco(2010)의 MIP 수리 모형이 최소 이격 간격에 대한 제한조건이 없어 최소 이격 간격이 없는 배치형태가 허용되는 문제점이 있다고 지적하면서 Chung and Tanchoco(2010)의 MIP 모형의 수리 식을 수정하였다. Murray et al.(2013)는 이동하는 방향에 따라 두 설비 간의 이동횟수가 서로 다른 설비 간의 비 대칭적인 자원의 흐름을 고려하도록 Zhang and Murray(2012)가 수정한 수리 모형을 확장시켰다. 그리고 Chung and Tanchoco(2010)가 제시한 휴리스틱 알고리즘에 간단한 주변 탐색과정(local search)을 추가하여 개선된 휴리스틱 방법론을 제시하였다. Zhang and Cheng(2014)은 DRLP를 조합 최적화 문제와 선형계획법(LP)문제로 구분해 조합 최적화 문제를 풀기위한 3가지의 휴리스틱 방법론과 선형계획법을 위한 프로그램으로 CPLEX를 사용하였다. Anjos et al.(2016)은 모든 설비들에 대해 인접한 설비 간의 거리가 모두 같은 DRLP를 풀기 위해 Zhang and Cheng(2014)처럼 DRLP를 2단계로 구분해 정수계획법(Integer linear programming)과 semidefinite programming(SDP)을 사용하는 방법론을 제시하였다. Amaral(2013b), Secchin and Amaral(2019), Amaral(2019), 그리고 Chae and Regan(2020)은 개체 간의 수평, 수직적 위치에 관한 상대적인 위치관계를 나타낼 수 있는 하나의 이진변수를 통해 DRLP 문제를 새로운 MIP로 표현하고 이후 순차적으로 발전시킨 논문들이다. 따라서 최근의 MIP 모델(Chae and Regan, 2020)은 Amaral(2013b)의 모델보다 확연히 효율적으로 DRLP 문제를 풀수 있다.

최근 DRLP에 대해 휴리스틱 방법론을 적용하는 연구들이 진행되었다. 방법론은 이전의 연구들처럼 DRLP를 개체들의 상대적인 위치를 찾아내는 조합 최적화 문제에 대한 부분과 개체 간의 상대적인 위치에서의 절대적인 위치를 찾는 해의 범위가 연속적인 문제로 구분하여 제시되었다. Amaral(2020)은 개체들의 상대적인 위치와 절대적 위치를 모두 휴리스틱으로 결정하는 방법론을 먼저 제시하고 더 좋은 배치를 찾기 위해 상대적인 위치만 휴리스틱으로 결정하고 절대적 위치를 LP로 해결하는 방법론을 제시하여 LP를 적용했을 때의 개선효과를 확인하였다. Guan et al.(2020) 역시 개체들의 상대적인 위치를 찾는 방법론과 절대적 위치를 찾는 방법론을 구분하여 제시했는데 상대적인 위치를 결정하는 방법론은 이전의 연구들처럼 휴리스틱 알고리즘을 사용하였지만 절대적인 위치를 결정하는 방법론은 이전 연구들과 달리, LP 사용 시 큰 규모의 문제에 대해 값을 도출하는데 너무 많이 걸린다는 단점을 해결하기 위해 LP와 같은 정확기법을 사용하지 않고 메타 휴리스틱인 PSO(Particle Swarm Optimization)를 사용하였다. 해당 연구는 처음으로 개체 수가 80개인 문제까지 제시하는 방법론을 적용하여 실험하였고, 실험 결과에서 이전 연구들보다 7개 문제들에 대해 더 좋은 값을 찾아냈다.

배치 설계문제 유형 중에서 DRLP와 연관되어 DRLP와 유사한 문제가 있는데 복도 할당 문제(corridor allocation problem; CAP)와 평행 열 배치 문제(parallel row ordering problem; PROP)이다. CAP(Ahonen et al., 2014; Amaral, 2012b)와 PROP(Amaral, 2013a; X. Yang et al., 2020)는 DRLP와 같이 이열(double

row)을 따라 개체들을 배치하지만 각 열의 가장 왼쪽에 위치하는 첫 개체가 미리 할당된다는 점에서 DRLP와 다르다고 할 수 있다. 그리고 CAP와 PROP는 두 인접한 개체 사이에 공간이 없다고 가정한다. 심지어 PROP는 각 개체들이 특정 열에 미리 할당되는 문제로 DRLP에 비해 더 제한적인 문제라고 할 수 있다. 이 두 유형은 주로 사무실로 사용되는 건물이나 학교, 병원 등의 공간에 대한 배치에 적용된다.

앞서 서론에서 언급한 것처럼, CS 알고리즘은 해의 범위가 연속적인 최적화 문제들에 쉽고 간단하게 적용이 가능하며 알고리즘이 최종 결과 값을 도출하는데 걸리는 시간 역시 다른 메타 휴리스틱 방법론들과 비교했을 때 상대적으로 빨랐다. CS 알고리즘의 적용과 성능에 대한 전반적인 정리 및 요약은 X.S. Yang and Deb(2014)가 다루었다.

CS 알고리즘은 해를 탐색하는 범위가 연속적인 최적화 문제에 맞춰 고안된 알고리즘으로 새로운 해를 탐색하는 방법으로 Lévy flight 라는 랜덤 워크 모델을 사용하기 때문에 CS 알고리즘을 조합 최적화 문제에 적용하기가 어렵다고 앞에서 설명하였다. 실수 값의 형태로 도출되는 Lévy flight 값을 정수 값이나 이진 값(1또는 0)으로 이루어진 배열의 형태로 표현되는 이산적인 해(discrete solution)에 적용하기 위해서는 일종의 변형이 필요하다. Burnwal and Deb(2013)은 FMS의 일정 최적화(scheduling optimization)를 위해 CS 알고리즘 기반의 접근법을 제시하였는데 Lévy flight 값에 의한 해 탐색의 방법으로 Lévy flight 값을 발생시켜 그 값에 가장 가까운 정수를 찾고 해당 정수 값을 정수로 이루어진 배열로 표현된 해에 더해줌으로써 해에 변형이 일어나게 하였다. 해당 연구에서 다루는 flow shop scheduling 문제에서는 작업(job)에 대한 순서정보를 담은 배열이 해(solution)가 되는데 Lévy flight 값에 의해 순서에 해당되는 정수 값을 임의로 바꿔 줌으로써 변형이 일어난다. Li and Yin(2013)와 Ouaarab et al.(2015a)은 앞에서 설명한 flow show scheduling 문제에 CS 알고리즘을 적용하였는데 Li and Yin(2013)은 Lévy flight 를 사용하는 방법으로 본 연구에서 적용한 Random-key 방법을 적용하였다. 하지만 본 연구에서 적용한 Random-key 방법과는 다르게, 1이 넘어가는 값에 대해 제한을 두지 않고 사용하였다. Ouaarab et al.(2015a)은 Lévy flight 값에 따라 구간을 설정하여 해당 구간에 따라 다른 주변 탐색 기법(local search)을 적용하여 해를 탐색하였다. Ain and Bey(2012)은 해가 0과 1의 이진 값으로 이루어진 배열로 표현된 배낭문제(knapsack problem)를 풀기 위해 기존의 CS 알고리즘을 변형시켜 binary cuckoo search(BCS)알고리즘을 고안하여 적용하였다. Ouaarab et al.(2014a, 2014b)은 외판원 문제 (traveling salesman problem; TSP)를 해결하기 위해 CS 알고리즘을 적용하였다. 그리고 Quaarab et al.(2015b)는 Bean(1994)이 처음 소개한 Random-key 방법을 TSP에 적용하여 소개하였다. Random-key 방식은 Lévy flight 값을 적용하기 위해 앞서 다른 연구 들에서 사용한 것처럼 Lévy flight 값을 발생시키고 나서 다시 2-opt와 같은 방식을 통해 적용해줄 필요없이 곧바로 조합 최적화 문제의 해에 적용 가능하다. 이러한 이점 때문에 이후 연구들에서는 CS를 조합 최적화 문제에 적용할 때 Random-key 방법이 주로 적용되어졌다. Kang et al.(2018)은 본 연구에서 다루는 배치설계에 Random-key 방법을 적용한 CS 알고리즘을 사용하였다. 해당 연구에서는 Drira et al.(2007)가 배치형태에 따라 분류한 유형 중 하나인 루프(loop) 형태로 개체를 배치하는 문제를 다루었으며 이열 배치설계는 개체의 길이가 정해져 있는 것과 달리 개체의 길이가 정해져 있지 않고 배치공간에 따라 유동적으로 바뀌는 개체에 대해 다루었다.

지금까지 살펴본 본 연구에 대한 기존 문헌들을 통해 알 수 있었던 것은 다음과 같다. CS 알고리즘은 연속 최적화 문제에서의 성능이 검증되어 최근 조합 최적화 문제에 적용해보는 연구가 늘어나고 있지만 아직까지 본 연구에서 다루는 이열 배치설계에 CS 알고리즘을 적용하여 진행된 연구는 없었다. 따라서 본 연구에서는 이열 배치설계(DRLP)에 대한 해결 방법론으로 메타 휴리스틱 방법인 CS 알고리즘을 적용하였으며, 여러 연구들에서 성능이 검증된 Random-key 방법으로 Lévy flight 값을 적용해주었다.

### 3. 이열 배치설계 문제 정의(Problem description)

이 장에서는 이열 배치설계 문제(DRLP)에 대해 자세히 설명하고 이열 배치설계의 수리 식 모형을 제시하고 설명해 본다. 본 연구에서는 서로 다른 길이의 개체들을 이열(Double row)로 배치하는 문제를 다루고

있다. 이열 배치설계의 기본적인 형태는 최적 값을 찾기 위한 혼합정수계획법(Mixed Integer Programming; MIP)모델로 수리 식을 표현할 수 있다(Chung and Tanchoco, 2010). 그리고 앞 장에서 설명한 것처럼 이열 배치설계에 대해 여러 연구들에서 문제 해결 시간을 단축시키고 효율성을 높이기 위해 MIP 모델에 제한조건을 추가하는 등의 변형을 주었다(Amaral, 2013b, 2019; Chae and Regan, 2020; Secchin and Amaral, 2019; Zhang and Murray, 2012). 하지만 본 연구에서는 이해를 위해 기본적인 개념적 모델을 소개한다.

이열 배치설계는 <Figure 2>에서 보여 지는 것처럼 시설 내에 중앙 복도가 있다고 가정하며 중앙 복도의 너비는 문제에서 고려하지 않는다. 각 개체들은 중앙 복도의 위쪽과 아래쪽에 배치된다. 각 개체들의 형태는 직사각형으로 가정되며, 개체의 길이와 개체 간의 자원에 대한 흐름(물동량)이 주어진다. 자원들이 나가거나 들어오는 개체의 입출력 지점은 중앙 복도 방향 중간 지점이라고 가정한다. 이 배치설계의 목적은 물류비용이 최소화되는 배치를 찾아내는 것으로, 이때의 물류비용은 각 개체 간의 입출력 지점 사이의 거리와 물동량의 곱의 총합으로 결정된다. 여기서의 거리는 중앙 복도에 다니는 작업자나 AGV의 작업동선을 최소화하기 위해 최단거리를 가정하는데 복도의 너비가 고려되지 않기 때문에 직선거리가 사용된다.

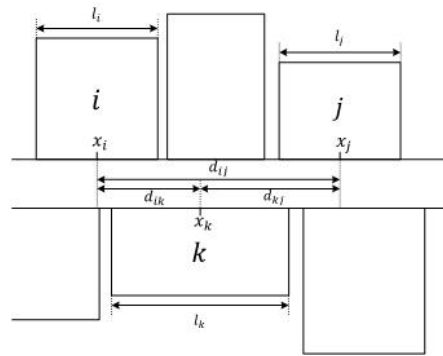


Figure 2. Representation of DRLP

또한, <Figure 2>에서 볼 수 있는 것처럼  $x_i$ 는 개체  $i$ 의 중간 지점의 좌표를 의미하며,  $l_i$ 는 개체  $i$ 의 길이를 나타낸다. 그리고 개체  $i$ 와  $j$ 개체 간의 거리를  $d_{ij}$ 로 표현한 것을 확인할 수 있다. 모든 개체들의 길이는  $L$ 로 나타냈으며, 개체와 개체 사이의 물류 흐름량인 물동량은  $f_{ij}$ 로 표현하였다. 본 연구에서 사용한 이열 배치설계 시에 주어지는 상수 값인 파라미터(Parameters)와 개체의 배치에 따라 변하는 값인 변수(Variables)는 각각 <Table 1>, <Table 2>에 요약하여 정리하였다.

Table 1. Parameters of the double row layout problems

$n$	The number of departments
$f_{ij}$	Amount of flow between department $i$ and $j$
$l_i$	Length of department
$L$	Sum of total departments' length( $L = l_1 + l_2 + l_3 + \dots + l_n$ )

Table 2. Variables of the double row layout problems

$x_i$	Abscissa of the center or P/D point of department
$d_{ij}$	Distance between department $i$ and $j$
$\alpha_{ij}$	Horizontal and vertical relative location with a combination of variables (1, if department $i$ is to the left of department $j$ and 0, otherwise)

원래 배치설계 시에는 <Figure 2>에서처럼 모든 개체들에 대해서 근접한 두 개체 사이에 서로 다른 최소한의 이격 간격(clearance)가 존재한다. 이러한 경우, 이열 배치설계를 위해서 주어지는 파라미터는 <Table 1>에서 설명한 파라미터를 포함해서 개체  $i$ 와 개체  $j$ 사이에 필요한 이격 간격(clearance)이 추가되는데 (Heragu and Kusiak, 1988)실제 문제에서는 최소 이격 간격은 주로 유지보수(maintenance)를 위해 사용되기 때문에 개체의 배치순서가 아닌, 개체에 따라 달라지며 이런 경우에는 이격 간격을 개체의 길이에 포함하여 생각할 수 있다(Chung and Tanchoco, 2010). 따라서 본 연구에서는 가장 간단한 이열 배치설계 모형에 대해 설명하기 위해 이격간격이 없는 유형의 모형에 대해 설명하고 수리 모형의 목적함수와 제한조건들은 다음과 같다.

$$\text{Minimize} \quad \sum_{(i,j); i < j} c_{ij} d_{ij} \quad (1)$$

$$\text{Subject to} \quad d_{ij} \geq x_i - x_j, \quad \forall i, j (i < j) \quad (2)$$

$$d_{ij} \geq x_j - x_i, \quad \forall i, j (i < j) \quad (3)$$

$$d_{ij} - \alpha_{ij} \frac{(l_i + l_j)}{2} - \alpha_{ji} \frac{(l_i + l_j)}{2} \geq 0, \quad \forall i, j (i < j) \quad (4)$$

$$x_i + \frac{(l_i + l_j)}{2} \leq x_j + L(1 - \alpha_{ij}), \quad \forall i, j (i < j) \quad (5)$$

$$\alpha_{ij} \in \{0, 1\}, \quad \forall i, j (i \neq j) \quad (6)$$

$$\frac{l_i}{2} \leq x_i \leq L - \frac{l_i}{2}, \quad 1 \leq i \leq n \quad (7)$$

앞서 설명한 것처럼 이열 배치설계의 목적함수(The objective function)는 각 개체 간의 입출력 지점 사이의 거리와 물동량의 곱의 총합으로 계산되는 물류비용을 최소화하는 것으로 (1) 식으로 표현될 수 있다. 제한조건(Constraints) (2)와 (3)은 각 개체 간의 거리를 절대값으로 설정해주고 (4)는 만약 개체  $i$ 와 개체  $j$ 가 같은 열에 있으면  $i$ 와  $j$ 사이 거리인  $n$ 는 적어도  $n$  되도록 해주는 조건이다. 제한조건(5)는 개체 간의 겹침 방지 조건으로 어떤 개체들도 서로 겹쳐지지 않게 만드는 조건이다. 제한조건 (6)은 이열 배치설계의 변수에서 설명한 개체  $i$ 와 개체  $j$ 의 상대적인 위치에 대한 이진변수를 정의해주는 조건이다. 마지막 제한조건(7)은 변수의 범위에 대한 조건이다.

본 연구에서 사용한 방법론에서는 위에서 설명한 제한조건들을 CS 알고리즘과정내에서 적용하였다. 따라서 CS 알고리즘에서는 제한조건들을 모두 만족하는 가능 해(feasible solution)만 탐색하게 되어 해를 탐색하는 과정을 더욱 효율적으로 만들어 준다.

#### 4. Cuckoo Search 기반의 이열 배치설계 방법론

이 장에서는 이열 배치설계를 해결하기 위한 방법론에 대해서 자세히 다룬다. 본 연구에서는 메타 휴리스틱인 Cuckoo Search 알고리즘을 적용하였고 이열 배치에서의 개체 간 빈 공간(clearance)문제를 해결하면서 상대적인 위치에서의 가장 좋은 배치를 도출하기 위해 flow값이 존재하지 않는 가상의 Dummy 개체를 추가하였다. 하지만 메타 휴리스틱을 통한 새로운 이열 배치 탐색과 탐색한 이열 배치에 대한 평가를 위해서는 이열 배치형태를 순열(Permutation)과 같은 이산적인 해(Discrete solution)의 형태로 표현하는 과정이 먼저 필요하며 개체들을 어떻게 이열(double row)로 배치하는지에 따라 적용하는 메타 휴리스틱의 성능도 달라지기 때문에 적용한 CS 알고리즘을 설명하기에 앞서 먼저 이열 배치형태를 메타 휴리스틱에 적용할 수 있도록 이열 배치형태를 어떻게 표현했는지(표현방법)와 어떤 방식으로 각 개체들을 이열로 배치했는지(이열 배치 방식)에 대해 설명한다. 그 다음, 본 연구에서 사용하는 메타 휴리스틱 방법인 Cuckoo

Search(CS)알고리즘에 대해 설명하고, 조합 최적화 문제인 이열 배치설계에 CS(의 Lévy flight)를 적용하기 위한 방법으로 본 연구에서 사용한 Random-key 방식에 대해서 설명한다. 그리고 마지막으로 이열 배치에서 개체 간의 빈 공간(clearance)에 대한 문제와 함께 이를 해결하기 위해 적용한 Dummy 개체에 대해 설명한다.

#### 4.1 이열 배치 방식(Layout representation)

본 연구에서는 개체들에 대한 이열 배치 형태를 기본적으로 개체들을 나열하는 순서(sequence)에 대한 정보를 담은 순열(permutation)  $\pi = (\pi[1], \pi[2], \dots, \pi[i], \dots, \pi[n])$ 로 나타낸다. 하지만 어떤 순서로 개체들을 구성하여 어떻게 개체들을 윗 열(upper row)과 아랫 열(lower row)로 구분하는 지에 대한 방법을 본 연구에서는 두 가지로 사용하였다. 첫 번째는 upper row의 개체들 길이 합 와 lower row의 길이 합을 비슷하게 맞추는 방법이고 두 번째는 개체의 길이를 고려하지 않고 개체의 전체 개수를 고려하여 개체 개수의 반으로 upper row와 lower row를 구분하는 방법이다.

1) 개체의 길이를 고려해 이열(double row)간 길이 밸런스를 맞추는 방법: LBM

본 연구에서 개체들에 대해 윗 열과 아랫 열중 어느 열로 배치할지에 대한 첫 번째 방법으로 윗 열과 아랫 열에 배치되는 개체들의 길이를 고려하여 열 간의 길이 밸런스를 맞추는 방법(Length Balance based Method; LBM)을 사용하였다. 해당 방법에 대한 pseudo-code는 <Algorithm 1>에 제시하였다.

Algorithm 1. Length Balance based Method(LBM)

---

Input: A permutation  $\pi$  (sequence of departments)  
Output: A upper row sequence  $\pi_U$ ; A lower row sequence  $\pi_L$ ;  
 $L_u \leftarrow 0$ ;  $L_l \leftarrow 0$ ;  $\pi_U = \emptyset$ ;  $\pi_L = \emptyset$ ;  
**for**  $i \leftarrow$  from 1 to  $n$  **do**  
    **if** ( $L_u < L_l$ ) **then**  
         $L_u \leftarrow L_u + l_{\pi[i]}$ ;  $\pi_U \leftarrow \pi_U \cup \pi[i]$ ;  
    **else if** ( $L_l < L_u$ ) **then**  
         $L_l \leftarrow L_l + l_{\pi[i]}$ ;  $\pi_L \leftarrow \pi_L \cup \pi[i]$ ;  
    **else**  
         $r \leftarrow$  generate a uniform random number in (0, 1);  
        **if** ( $r < 0.5$ ) **then**  
             $L_u \leftarrow L_u + l_{\pi[i]}$ ;  $\pi_U \leftarrow \pi_U \cup \pi[i]$ ;  
        **else**  
             $L_l \leftarrow L_l + l_{\pi[i]}$ ;  $\pi_L \leftarrow \pi_L \cup \pi[i]$ ;  
        **end if**  
    **end if**  
**end for**

---

upper row의 모든 개체에 대한 길이 합을  $L_u$ , lower row의 모든 개체에 대한 길이 합을  $L_l$ 로 나타냈고 upper row의 순서에 대한 배열(sequence)과 lower row의 순서에 대한 배열을 각각  $\pi_U$ 와  $\pi_L$ 로 표현했다. LBM은 처음에  $L_u$ 과  $L_l$ 을 0으로 설정하고  $\pi_U$ 와  $\pi_L$ 을 비어있는 배열로 설정하여 초기화한다. 그 다음 각 개체가 upper row 또는 lower row에 할당되는 배치방식은 다음과 같다. 만약  $L_u$ 보다  $L_l$ 이 더 길면 모든 개체의 순서에 대한 순열  $\pi$ 의  $i$ 번째 개체의 길이  $l_{\pi[i]}$ 를  $L_u$ 에 더해주고  $\pi_U$ 에 해당 개체  $\pi[i]$ 를 추가한다. 반대로 만약  $L_l$ 보다  $L_u$ 가 더 길면  $l_{\pi[i]}$ 를  $L_l$ 에 더해주고  $\pi_L$ 에  $\pi[i]$ 를 추가해준다. 그리고 만약  $L_u$ 과  $L_l$ 이 같으면 위의 배치방식 중 하나를 랜덤하게 수행한다. 이 과정은 의 모든 개체가 upper 또는 lower row에 할당될 때까지 반복된다. 모든 과정이 끝나면 모든 개체가 또는 에 할당되어 upper row인지 lower row인지 결정된다. 위의 과정에 따라서 초기과정에  $L_u$ 과  $L_l$ 이 모두 0이므로  $L_u$ 과  $L_l$ 이 같기 때문에  $\pi$ 의 첫 번째 순서에 해당되는 개체  $\pi[1]$ 는 upper 또는 lower중 랜덤하게 배치된다.

## 2) 개체의 개수를 고려해 개체 개수를 고려해 cut point를 설정하는 방법: CSM

개체들에 대해 upper row와 lower row중 어느 열로 배치할지에 대한 두 번째 방법으로 모든 개체의 순서에 대한 배열에 대해 cut point를 설정하는 방법(Cut point selecting method; CSM)을 사용하였다. CSM은 배열 내의 첫 번째부터 설정한 cut point에 해당되는 순서까지의 개체들을 순서대로 upper row에 배치한다. 그리고 나머지 cut point+1번째부터 마지막 순서까지의 개체들을 순서대로 lower row에 배치한다. 본 연구에서는 cut point를 전체 개체 수의 반을 cut point로 설정하였다. 만약 개체 수가 홀수인 경우에는 개체 수의 반에 대한 값에 반올림을 적용하여 cut point를 설정하였다. <Figure 3>과 같이, 개체가 11개라고 하면 11의 반인 5.5에서 반올림하여 cut point를 6으로 설정하고 이렇게 되면 1~6번 개체가 upper row에 배치되며 7번부터 11번까지의 개체가 lower row에 배치되게 된다.

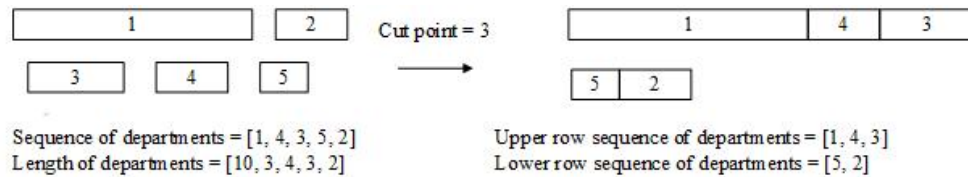


Figure 3. Cut point selecting method

## 4.2 Cuckoo Search 알고리즘

Cuckoo Search(CS) 알고리즘은 뺨꾸기들의 생존율을 높이기 위한 행동패턴에 영감을 받아 Yang과 Deb에 의해 고안되어진 발견적 기법이다(X. S. Yang and Deb, 2009). 일부 뺨꾸기 종은 매우 위험성이 큰 번식 전략으로써 탁란(brood parasitism)이라고 불리는 행위를 한다. 이 행위의 가장 큰 특징은 다른 새의 둥지에 자신의 알을 낳아 그 새가 자신의 알을 품어 알에서 새끼가 부화되도록 한다는 것이다. 이를 진화의 관점에서 보게 되면, 뺨꾸기들은 자신이 알을 낳은 둥지의 새가 자신의 알을 버릴 확률을 줄여 종의 생존율을 높여야 한다(Payne and Sorensen, 2005). 이러한 CS 알고리즘은 탐색이 효과적이면서 최적의 생존 전략인 Lévy flights를 새로운 해를 탐색하는데 사용함으로써 성공적으로 뺨꾸기들의 생존 행동패턴을 모방하였다.

프랑스 수학자 Paul Lévy의 이름을 딴 Lévy flights는 변화량(step length)이 확률밀도함수의 양쪽 꼬리 부분이 두꺼운 heavy-tailed 확률분포를 따르는 랜덤 워크의 한 종류이다(Brown et al., 2007). Lévy flights는 평균과 분산이 무한대라는 특성이 있어 CS 알고리즘이 해 공간을 더 효과적으로 탐색하게 해주며 정규 분포를 따르는 랜덤 워크(Gaussian random walk)보다 효과적이다(X. S. Yang and Deb, 2014). 또한, 여러 연구에서 동물들이 먹이를 탐색하거나 사냥꾼이 사냥감을 탐색하기 위한 행동 패턴이 Lévy flights와 일반적으로 같은 특성이 가진다고 알려져 있다(Ouaarab et al., 2015b). Lévy flights 모형은 일반적으로 변화량의 범위가 작게 나타나고 가끔씩 변화량의 범위가 크게 나타나는 특징이 있다(Brown et al., 2007; Shlesinger et al., 1995).

CS 알고리즘에서 둥지안의 알(host egg)은 한 개의 해(solution)을 나타내고 뺨꾸기의 알(cuckoo egg)은 새로운 해를 의미한다. CS 알고리즘의 목표는 둥지 내에서 안 좋은 해(host egg)와 교체할 새롭고 더 좋을 가능성이 있는 해(cuckoo egg)를 만드는 것이다. CS 알고리즘에서 뺨꾸기의 행동패턴에 대해 모방한 3가지 규칙은 다음과 같다(X. S. Yang and Deb, 2009, 2013, 2014). (1) 각각의 뺨꾸기들은 한 번에 한개의 알(solution)을 낳고 랜덤하게 선택된 둥지에 알을 넣는다; (2) 가장 좋은 품질의 알을 가진 가장 좋은 둥지는 다음 세대에 유지된다; (3) 알을 낳을 수 있는 둥지의 수는 고정되어 있다. 뺨꾸기의 알은 둥지의 주인 새에게  $P_a \in [0,1]$ 의 확률로 발견되며, 그럴 경우, 둥지 주인 새는 뺨꾸기의 알을 버리거나 둥지를 버리고 새로운 곳에서 새 둥지를 짓는다. 이러한 규칙들을 기반으로 한 기본적인 CS 알고리즘의 과정은 <Algorithm 2>에서 제시하였다.

---

Algorithm 2. Pseudo Code of the Cuckoo Search via Lévy flights

---

Objective function  $f(x)$ ,  $x = (x_1, \dots, x_m)^T$ ;  
 Generate initial population of  $n$  host nests  $x_i$  ( $i = 1, \dots, n$ );  
**While** ( $t < \text{the maximum number of iterations}$ ) or (stop criterion) **do**  
     Get a cuckoo (say,  $x_i$ ) randomly by Lévy flights;  
     Evaluate quality/fitness of new solution  $x_i$  or  $F(x_i)$ ;  
     Randomly select a nest within the population (say,  $x_j$ ) and evaluate its fitness  $F(x_j)$ ;  
     **if** ( $F(x_i) > F(x_j)$ ) **then**  
         replace  $x_j$  by the new solution  $x_i$ ;  
     **end if**  
     A fraction ( $p_a$ ) of worse nests are abandoned and replaced by a new ones;  
     Keep the best solutions (or nests with quality solutions);  
     Rank the solutions and find the current best  
**end while**

---

<Algorithm 2>를 보면, 초기화 단계에서 목적함수  $f(x)$ 를 정의하고, 각각의 목적함수에 대한 해  $x_i$  ( $i = 1, \dots, n$ )를 랜덤하게 생성하여 해 집단(population)을 형성한 뒤, 그 중 가장 좋은 해(current best solution)를 찾기 위해 각각의 해에 대한 목적함수 값(objective function value; OFV)을 계산한다. 그리고 CS 알고리즘은 식(8)과 같이, Lévy flights를 통해 새로운 해  $x_i^{(t+1)}$ 를 생성한다(X. S. Yang and Deb, 2014).

$$x_i^{(t+1)} = x_i^{(t)} + \alpha \text{Lévy}(s, \lambda) \quad (8)$$

where

$$\text{Lévy}(s, \lambda) = \frac{\lambda \Gamma(\lambda) \sin(\pi\lambda/2)}{\pi} \frac{1}{s^{1+\lambda}}, \quad (0 < \lambda < 2) \quad (9)$$

여기에서  $x_i^{(t)}$ 는  $t$  번째 iteration에서의 해이다. 그리고  $\alpha > 0$ 은 CS 알고리즘에 적용하고 있는 최적화 문제의 크기와 관련해 해가 변동되는 정도(step size)를 조절해주는 요소이다.  $\text{Lévy}(s, \lambda)$ 는 앞에서 설명한 Lévy flights 랜덤 워크로 발생하는 값으로 식(9)를 통해 도출된다. 여기에서  $\Gamma$ 는 감마함수(Gamma function)를 나타내고,  $\text{Lévy}(s, \lambda)$ 의 랜덤하게 변동되는 정도(step length)를 나타내는  $s$ 는 Lévy 분포를 따르는 값으로 Lévy 분포로부터 도출되는 랜덤 값이며 식(10)을 통해 값이 계산된다.

$$s = \frac{u}{|v|^{1/\lambda}} \quad (10)$$

식(10)에서의  $u$ 와  $v$ 는 식(11)같이 평균이 0이고, 표준편차가 각각  $\sigma_u$ 와  $\sigma_v$ 인 정규분포로부터 도출되는 랜덤 값들이고,  $\sigma_u$ 과  $\sigma_v$ 은 식(12)를 통해 계산된다. (X. S. Yang and Deb, 2014).

$$u \sim N(0, \sigma_u^2), \quad v \sim N(0, \sigma_v^2) \quad (11)$$

where

$$\sigma_u = \left\{ \frac{\Gamma(1+\lambda) \sin(\pi\lambda/2)}{\Gamma[(1+\lambda)/2] \lambda 2^{(\lambda-1)/2}} \right\}^{1/\lambda}, \quad \sigma_v = 1 \quad (12)$$

이렇게 Lévy flights를 통해 새로운 해  $x_i$ 를 생성한 후, 이 해에 대한 목적함수 값(OFV)  $F(x_i)$ 를 구한다. 그 다음,  $x_i$ 와 OFV를 비교하기 위해 현재의 해 집단(population)에서 랜덤하게 하나의 해( $x_j$ )를 선택한다. 새로운 해  $x_i$ 는 만약 목적함수 값  $F(x_i)$ 이  $x_j$ 의 목적함수 값  $F(x_j)$ 보다 좋다면  $x_j$ 를  $x_i$ 로 바꾸게 된다. 그 다음 단계로,  $P_a \in [0, 1]$ 의 비율만큼 안 좋은 해를 버리고 완전히 랜덤하게 해를 발생시키거나 Lévy flights가 크게 튀는 경우를 활용함으로써 새로운 곳에 새 둥지가 지어진다. 여기에서  $P_a$ 를 높게 설정하면 해를 넓게 탐색하는 과정(diversification)의 비율이 증가하고 반대로  $P_a$ 를 낮게 설정하면 그 과정의 비율

이 낮아진다. 이 과정이 끝나고 나면 현재 iteration까지 중의 해 집단에서 가장 좋은 해를 찾기 위해 OFV가 좋은 순서대로 정렬시킨다. 지금까지 설명한 과정들은 iteration의 최대 숫자에 도달할 때까지 반복된다.

본 연구에서는 CS의 개선된 알고리즘을 기반으로 적용하였다(Ouaarab et al., 2014b). 여기에서 개선된 점은 똑똑한 빠꾸기(smart cuckoo)라는 새로운 유형의 빠꾸기를 추가하였다. 이 새로운 유형의 빠꾸기는 자신이 알을 낳기로 선택한 둥지가 좋은 선택이었는지 확인하기 위해 알을 낳기로 결정한 둥지를 알을 낳기 전과 후에 관찰하여 알이 버려 지기 전에 더 좋은 둥지에 알을 낳기 위해 둥지를 바꾸거나 다른 곳으로 날아간다. 그래서 현재의 해로부터  $P_c \in [0,1]$ 만큼의 비율의 빠꾸기들은 Lévy flights를 통해 새로운 더 좋은 해를 탐색한다. 이러한 빠꾸기를 추가함으로써 CS가 개선되는 점은 현재의 iteration에서의 해들의 주변 탐색을 더 강화할 수 있으며, <Algorithm 3>을 통해 개선된 CS의 과정을 제시하였다.

Algorithm 3. Pseudo Code of the Improved Cuckoo Search Algorithm

---

Objective function  $f(x)$ ,  $x = (x_1, \dots, x_m)^T$ ;  
Generate initial population of  $n$  host nests  $x_i (i = 1, \dots, n)$ ;  
**While** ( $t < \text{the maximum number of iterations}$ ) or (stop criterion) **do**  
    **Start searching with a fraction ( $p_c$ ) of smart cuckoos**  
    Get a cuckoo (say,  $x_i$ ) randomly by Lévy flights;  
    Evaluate quality/fitness of new solution  $x_i$  or  $F(x_i)$ ;  
    Randomly select a nest within the population (say,  $x_j$ ) and evaluate its fitness  $F(x_j)$ ;  
    **if** ( $F(x_i) > F(x_j)$ ) **then**  
        replace  $x_j$  by the new solution  $x_i$ ;  
    **end if**  
    A fraction ( $p_a$ ) of worse nests are abandoned and replaced by a new ones;  
    Keep the best solutions (or nests with quality solutions);  
    Rank the solutions and find the current best  
**end while**

---

### 4.3 Random-key encoding scheme

CS 알고리즘이 연속 최적화 문제에 맞춰 고안되었기 때문에 Lévy flights의 연속적인 해 탐색범위를 조합 최적화 문제의 해에 적용할 수 있도록 일종의 변형(modification)과정이 필요하다. 그래서 본 연구에서는 최근 많은 연구들에서 메타 휴리스틱을 조합 최적화 문제에 적용할 때 쉽고 성능의 손실 없이 적용 가능하다고 많이 사용되고 있는 Random-key 방법을 사용하였다. Random-key 방법은 연속적인 공간(continuous space)을 이산적인 조합 가능한 공간(combinatorial space)으로 바꿔 주는데 사용하는 기법이다(Beaun, 1994).

본 연구에서는 Random-key를 통해 초기해 생성 및 새로운 해를 탐색하는 과정에도 적용하였다. Random-key로 이열 배치설계의 해를 표현하는 방법은 <Figure 4>와 같이 설명될 수 있다.

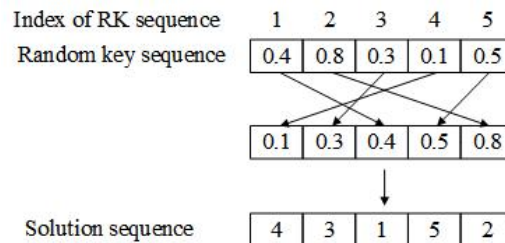


Figure 3. Random-key encoding scheme

만약 이열 배치설계에서 배치해야 할 개체의 수가 5개라면, 그림 5와 같이,  $[0, 1]$ 사이의 범위에 있는 실수를 균일한 분포(uniform)로 랜덤하게 5개를 발생시켜 배열에 저장한다. 여기서 Random-key 배열(sequence)의 순서(index)를 이열 배치설계에서 주어지는 개체들의 번호로 인식한다. 그 다음, Random-key



배열을 섞어서 개체들의 순서도 섞게 되는데 여기서 Random-key 배열을 섞는 방법은 Random-key 배열의 Random-key값을 작은 순서대로 정렬시키는 것이다. 그렇게 되면 가장 작은 0.1의 index인 4가 가장 앞으로 오고 가장 큰 0.8의 index인 2가 가장 뒤로 가면서 원래 개체의 순서가 [1,2,3,4,5]였던 것이 [4,3,1,5,2]로 바뀌게 된다.

#### 4.4 Dummy 개체 적용

CS 알고리즘을 통해 얻게 되는 배치는 개체들의 상대적인 배치를 찾을 뿐 해당 위치에서 정확히 개체들이 어디에 배치되어야 하는지는 CS 알고리즘을 통해 찾을 수 없다. 배치하는 개체들의 각각의 길이와 물동량에 따라서 달라지지만 경우에 따라서 개체들에 대한 배치가 특정 개체들 간의 공간이 있는 경우에 총 물류이동 거리가 가장 낮을 수 있다. 이에 대한 대표적인 예시가 바로 DRLP의 여러 배치문제 중에서 'H7' 문제 형태이다. 'H7'은 Hungerländer and Anjos(2012)의 연구에서 처음 제안된 문제 형태이며 문제의 구성은 <Figure 4>에서 보는 것과 같이, 개체의 수가 7개이고, 각각의 개체의 길이는 10, 3, 4, 3, 2, 2, 2이다. 해당 문제 형태의 최적 배치형태는 <Figure 5>와 같이 나타나는데, (a)는 공간을 넣지 않고 배치했을 때의 최적 배치 형태이며, (b)는 개체간의 공간을 넣어주었을 때의 최적 배치 형태이다. 두 배치 형태 모두 개체들의 상대적인 위치 즉, 배치 순서는 앞 열이 [3,2,5,7], 뒷 열이 [6,4,1]로 동일하지만 공간을 넣지 않았을 경우에는 목적함수 값(OFV)이 166이고 공간을 넣었을 때는 목적함수 값이 159로 공간을 넣지 않았을 때보다 넣었을 때 목적함수 값이 더 낮아지는 것을 확인할 수 있다. 따라서 CS 알고리즘을 통해 탐색한 개체들의 상대적인 위치에 대해서 정확한 배치 위치를 찾는 방법이 필요하다. 본 연구에서는 물동량 값(flow)이 없는 가상의 Dummy 개체를 설정하여 개체 간에 공간을 넣어주었다.



Figure 4. 'H7' Instance

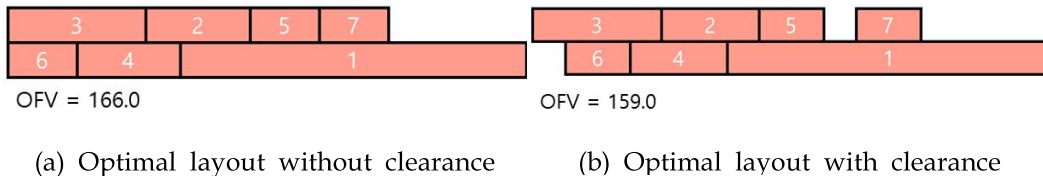


Figure 5. Optimal layout of 'H7' Instance

Dummy 개체를 추가할 때는 2가지를 설정해줘야 한다. 먼저, Dummy 개체의 수를 몇 개로 설정할 것인지 결정해줘야 하고 두 번째로 Dummy 개체의 길이를 몇으로 설정할 것인지를 결정해야 한다. Dummy 개체의 길이를 짧게 설정하고 개수를 많이 넣어줄수록 더 좋은 배치를 찾을 확률이 높아진다. 하지만 개체의 길이를 짧게 설정할수록 배치해야 할 개체의 수가 증가해 문제의 복잡도가 높아져 문제를 해결하는데 소요되는 시간이 기하급수적으로 증가한다. 또한, 개체의 수를 필요 이상으로 많이 설정하게 되면, 불필요하게 설정된 Dummy 개체들로 인하여 문제를 해결하는데 있어서 불필요한 시간소요가 발생하게 된다. 따라서 Dummy 개체의 길이를 너무 짧지도 길지도 않게 설정해주며 개수 역시 너무 많지도 적지도 않게 적절하게 설정해주는 것이 중요하다. 본 연구에서는 많은 시행착오를 거쳐 Dummy 개체의 길이를 0.5로 설정하고 해결해야 할 DRLP 유형의 개체 수에 따라 비례하게 설정하였다.

#### 5. 실험 결과

본 장에서는 4장에서 설명한 본 연구의 방법론의 성능을 평가하기 위해 진행한 실험에 대한 결과를 살펴보고 이전 연구들과 비교해본다. 본 연구에서 설명한 CS알고리즘을 포함한 방법론은 JAVA 프로그래밍 언

어를 통해 구현되었다. 실험은 Microsoft Windows 10 운영체제, Intel Core i9 CPU (3.3 GHz), 32GB RAM의 컴퓨터에서 진행되었다. 실험한 이열 배치설계 문제 데이터는 잘 알려진 문제들로 이전 연구들에서 실험이 진행되었던 개체 수가 5개부터 16개까지 18개의 문제들을 사용하였다(Amaral, 2006, 2013a; Fischer et al., 2019; Heragu and Kusiak, 1991; Hungerländer and Anjos, 2012; Secchin and Amaral, 2019).

## 5.1 매개변수 설정 (Parameters setting)

CS 알고리즘의 매개변수는 해 집단의 크기(Population Size;  $n$ ), 해 집단에서 해를 버리는 비율( $p_a$ ), 해 집단에서 smart cuckoo 역할을 하는 비율( $p_c$ ), Lévy flight의 step size 값( $\alpha$ )가 있다. 본 연구에서는 CS 알고리즘의 적절한 매개변수 값을 적용하기 위해 CS의 각 매개변수들에 대해 일정 간격을 두고 모든 매개변수 조합들에 대해 실험을 진행해 전반적인 문제들에 대해 가장 결과 값이 좋았던 조합으로 설정하였다. 그 결과  $n = 10$ ,  $p_a = 0.25$ ,  $p_c = 0.6$ ,  $\alpha = 0.3$ 으로 설정되었다. 해당 매개변수 조합으로 실험이 진행된 문제당 10번씩 실험을 진행하였다.

## 5.2. 이전 연구들의 MIP 모형과 실험결과 비교

비교를 진행한 연구에 대해서는 가장 최근에 이열 배치설계에 대한 연구가 진행이 되었던 MIP모형들의 결과로 비교하였다. 본 연구에서는 (Secchin and Amaral, 2019)에서 제안된 MIP 모형을 M1으로 표기하였으며, (Chae and Regan, 2020)에서 제안된 MIP 모형을 M2로 표기하였다. 그리고 본 연구에서 제시하는 방법론을 Improved CS로 표기하였다. 이전 연구들의 MIP 모형과 같이 실험이 진행된 문제들에 대한 연구 결과는 <Table 3>와 같이 나타났다. 표의 행에 대한 표기를 설명하면, N은 개체의 수를 나타내며 Optimal은 해당 문제의 최적 값을 의미한다. Best는 가장 잘 찾은 결과 값이며 M1과 M2는 MIP에 대한 풀이 값으로 최적값을 찾기 때문에 Best는 CS 만 해당이 된다. 단 시간제한으로 최적 해를 찾지 못한 경우에 대해서는 가장 좋은 값을 표기 하였다. Time(sec)은 각 방법론이 해당 문제에 대한 결과 값을 찾아내는데 걸린 시간을 초단위로 표현한 것을 의미한다. STD는 본 연구에서 10번 진행한 결과 값에 대한 표준 편차 값이고 Avg Time은 10번 진행한 실험의 평균 소요시간을 의미한다. 문제의 <Table 3>의 결과를 살펴보면, 본 연구 방법론의 STD 값이 모든 문제들에 대해 0의 값을 보였다. 이는 20개 이하의 <Table 3>의 문제들에 대해서는 방법론의 성능이 변동이 적고 결과 값을 잘 찾아내는 것을 의미한다. 각 연구들의 소요시간에 대해 살펴보면, 12개 이하의 문제들에 대해서는 MIP 모형의 소요시간이 본 연구의 방법론보다 더 빠른 결과값을 보였다. 하지만 13개 이상의 문제들에 대해서는 문제의 크기가 커질수록 MIP 모형의 소요시간이 크게 증가 하였다. 또한, 16개의 문제에 대해 이전 연구들의 MIP 모형은 최적 값을 찾지 못했다. 본 연구에서 제안한 방법론은 모든 문제에 대해 이전에 MIP로 찾은 최적 값을 매우 빠르게 찾았고 또한 지속적이고 안정적으로 값을 도출하였다. 이는 이 연구에서 테스트 되지 않았지만 큰 크기의 문제에서도 빠르게 좋은 해를 찾아 줄 것을 기대하게 한다.

Table 3. Result comparison

Instance			M1	M2	CS (in this study)		
Name	N	Optimal	Time (sec)	Time (sec)	Best	STD	Avg Time (sec)
HA5	5	52.5	0.05	0.02	52.5	0.0	1.291
HA6	6	190.5	0.06	0.05	190.5	0.0	1.324
HA7	7	159.0	0.11	0.08	159.0	0.0	3.76
HA8	8	189.5	0.39	0.19	189.5	0.0	3.769
HA9	9	486.5	1.02	0.75	486.5	0.0	2.957
HA10	10	821.0	8.44	2.36	821.0	0.0	10.655
HA11	11	773.5	15.39	2.98	773.5	0.0	12.354
HA12	12	1021.0	76.39	28.59	1021.0	0.0	15.411
HA13	13	1520.5	327.97	100.80	1520.5	0.0	17.824
HA14	14	1833.5	3,358.94	882.23	1833.5	0.0	19.751
HA15	15	2624.5	3,214.06	1918.53	2624.5	0.0	23.932
SA14a	14	2904.0	3,254.69	1207.88	2904.0	0.0	20.102
SA14b	14	2736.0	5,469.83	1532.05	2736.0	0.0	21.023
Am14_1	14	2738.5	4,986.4	1533.27	2738.5	0.0	21.893
HK15	15	16570.0	12297.41	2148.73	16570.0	0.0	25.297
Am15	15	3195.0	11221.34	5002.63	3195.0	0.0	27.465
Am16a	16	7365.5*	86400** (27.09%)***	86400** (13.69%)***	7365.5	0.0	30.127
Am16b	16	5870.5*	86400** (17.80%)***	86400** (8.29%)***	5870.5	0.0	30.372

\* best solution found before the process stops

\*\* The search process stops due to time limit (24 hrs) and the optimality gap is shown in parenthesis

\*\*\* Optimality gap

## 6. 결론

예전부터 오래도록 연구가 진행되었던 배치설계에 관련된 연구는 최근 배치설계의 효율화가 중요해진 시점에서 더욱 필요한 연구 분야이다. 본 연구에서는 배치설계의 다양한 조합 최적화 문제 중 제조환경에서 뿐만 아니라, 다양한 실제 업무 환경에서 가장 많이 활용되고 있는 이열 배치설계(Double row layout problem)에 대해 다루었고, 최근 규모가 커지고 있는 제조시설이나 물류센터에서 유동적인 배치설계를 고려해 규모가 큰 문제유형 해결에 집중하였다. 그리고 이열 배치설계를 해결하기 위한 방법론으로 메타 휴리스틱 중 하나인 Cuckoo Search 알고리즘을 사용하였고, 이열 배치설계 문제에 CS 알고리즘을 적용하기 위해 Levy flight 값에 Random-key 방식을 사용하여 해 탐색 공간을 이산적으로 바꾸었다. 그리고 이열 배치 방식으로 upper row와 lower row의 길이 밸런스를 맞추는 방법(Length balance based method)과 cut point를 설정하는 방법(Cut point selecting method)을 사용하여 적용한 CS 알고리즘의 해 집단(population)을 2개로 가져갔다. 또한, 배치의 빈 공간에 대한 문제를 해결하기 하고 물류비용을 더 줄일 수 있는 배치를 찾기 위해 Dummy 개체들을 추가하여 CS 알고리즘을 적용하였다.

이전 연구들에서 사용되었던 여러 가지 이열 배치설계 문제들에 대해 실험을 진행해 결과를 도출하였다. 실험결과, 이미 정확 기법(Exact method)을 통해 최적 값이 도출된 문제의 크기가 작은 문제들의 경우, 빠른 시간 내에 최적 값을 찾아냈다. 그리고 정확 기법을 통해 찾아낼 수 없는 규모의 문제에 대해서는 본 연구의 방법론이 더 빠른 시간 내에서 결과 값을 찾아내었다.

본 연구는 물류 및 유통 산업뿐만 아니라 현실에서 활용 범위가 넓은 이열 배치설계에 대한 해법을 제시함으로써 일열 배치와 관련된 다양한 분야에서 관련 비용을 절감시킬 수 있을 것으로 예상된다. 또한, 배치설계의 다른 유형의 문제에도 해당 방법론을 적용해 빠른 시간 안에 좋은 배치를 찾을 수 있을 것이라고 판단된다. 하지만 본 연구의 이열 배치설계는 배치되는 공간에 대해 제한을 두지 않아 현실적인 배치설계와는 조금 거리가 있다고 할 수 있다. 그렇기 때문에 배치공간에 대한 제한이 추가된 이열 배치설계에 대한 연구도 진행되면 실제 산업에 실용적으로 적용될 수 있을 것으로 예상된다.

## 참고문헌

- Ahonen, H., De Alvarenga, A. G., and Amaral, A. R. S. (2014). Simulated annealing and tabu search approaches for the Corridor Allocation Problem. *European Journal of Operational Research*, 232(1), 221-233.
- Ain, R., and Bey, E. (2012). Solving 0-1 knapsack problems by a discrete binary version of cuckoo search algorithm Amira Gherboudj \*, Abdesslem Layeb and Salim Chikhi. *Science*, 4(4), 2012.
- Amaral, A. R. S. (2006). On the exact solution of a facility layout problem. *European Journal of Operational Research*, 173(2), 508-518.
- Amaral, A. R. S. (2012a). The corridor allocation problem. *Computers and Operations Research*, 39(12), 3325-3330.
- Amaral, A. R. S. (2012b). The corridor allocation problem. *Computers and Operations Research*, 39(12), 3325-3330.
- Amaral, A. R. S. (2013a). A parallel ordering problem in facilities layout. *Computers and Operations Research*, 40(12), 2930-2939.
- Amaral, A. R. S. (2013b). Optimal solutions for the double row layout problem. *Optimization Letters*, 7(2), 407-413.
- Amaral, A. R. S. (2019). A mixed-integer programming formulation for the double row layout of machines in manufacturing systems. *International Journal of Production Research*, 57(1), 34-47.
- Amaral, A. R. S. (2020). A heuristic approach for the double row layout problem. *Annals of Operations Research*.
- Anjos, M. F., Fischer, A., and Hungerländer, P. (2016). Solution approaches for the double-row equidistant facility layout problem. In *Operations Research Proceedings 2014* (pp. 17-23). Springer.
- Bean, J. C. (1994). Genetic algorithms and random keys for sequencing and optimization. *ORSA Journal on Computing*, 6(2), 154-160.
- Benjaafar, S., and Sheikhzadeh, M. (2000). Design of flexible plant layouts. In *IIE Transactions (Institute of Industrial Engineers)* (Vol. 32, Issue 4).
- Braglia, M., Zaroni, S., and Zavanella, L. (2003). Layout design in dynamic environments: Strategies and quantitative indices. *International Journal of Production Research*, 41(5), 995-1016.
- Brown, C. T., Liebovitch, L. S., and Glendon, R. (2007). Lévy flights in Dobe Ju/hoansi foraging patterns. *Human Ecology*, 35(1), 129-138.
- Burnwal, S., and Deb, S. (2013). Scheduling optimization of flexible manufacturing system using cuckoo search-based approach. *International Journal of Advanced Manufacturing Technology*, 64(5-8), 951-959.
- Chae, J., and Regan, A. C. (2016). Layout design problems with heterogeneous area constraints. *Computers and Industrial Engineering*, 102(October), 198-207.
- Chae, J., and Regan, A. C. (2020). A mixed integer programming model for a double row layout problem. *Computers and Industrial Engineering*, 140(June 2019), 106244.
- Chung, J., and Tanchoco, J. M. A. (2010). The double row layout problem. *International Journal of Production Research*, 48(3), 709-727.
- Civicioglu, P., and Besdok, E. (2013). A conceptual comparison of the Cuckoo-search, particle swarm optimization,

- differential evolution and artificial bee colony algorithms. Artificial Intelligence Review, 39(4), 315–346.*
- Drira, A., Pierreval, H., and Hajri-Gabouj, S. (2007). Facility layout problems: A survey. *Annual Reviews in Control, 31(2), 255–267.*
- Fischer, A., Fischer, F., and Hungerländer, P. (2019). New exact approaches to row layout problems. *Mathematical Programming Computation, 11(4), 703–754.*
- Gandomi, A. H., Yang, X.-S., and Alavi, A. H. (2013). Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems. *Engineering with Computers, 29(1), 17–35.*
- Gandomi, A. H., Yang, X.-S., Talatahari, S., and Deb, S. (2012). Coupled eagle strategy and differential evolution for unconstrained and constrained global optimization. *Computers and Mathematics with Applications, 63(1), 191–200.*
- Garcia-Hernandez, L., Salas-Morera, L., Garcia-Hernandez, J. A., Salcedo-Sanz, S., and Valente de Oliveira, J. (2019). Applying the coral reefs optimization algorithm for solving unequal area facility layout problems. *Expert Systems with Applications, 138, 112819.*
- Garey, M. R., and Johnson, D. S. (1979). *Computers and intractability (Vol. 174).* freeman San Francisco.
- Guan, J., Lin, G., Feng, H. Bin, and Ruan, Z. Q. (2020). A decomposition-based algorithm for the double row layout problem. *Applied Mathematical Modelling, 77, 963–979.*
- Heragu, S. S., and Kusiak, A. (1988). Machine Layout Problem in Flexible Manufacturing Systems. In *Operations Research (Vol. 36, Issue 2, pp. 258–268).*
- Heragu, S. S., and Kusiak, A. (1991). Efficient models for the facility layout problem. *European Journal of Operational Research, 53(1), 1–13.*
- Hungerländer, P., and Anjos, M. F. (2012). A semidefinite optimization approach to space-free multi-row facility layout. *Groupe d'études et de recherche en analyse des décisions.*
- Ingole, S., and Singh, D. (2017). Unequal-area, fixed-shape facility layout problems using the firefly algorithm. *Engineering Optimization, 49(7), 1097–1115.*
- Kang, S., and Chae, J. (2017). Harmony search for the layout design of an unequal area facility. *Expert Systems with Applications, 79, 269–281.*
- Kang, S., Kim, M., and Chae, J. (2018). A closed loop based facility layout design using a cuckoo search algorithm. *Expert Systems with Applications, 93, 322–335.*
- Kim, M., and Chae, J. (2019). Monarch butterfly optimization for facility layout design based on a single loop material handling path. *Mathematics, 7(2).*
- Levary, R. R., and Kalchik, S. (1985). Facilities layout—a survey of solution procedures. *Computers and Industrial Engineering, 9(2), 141–148.*
- Li, X., and Yin, M. (2013). A hybrid cuckoo search via Lévy flights for the permutation flow shop scheduling problem. *International Journal of Production Research, 51(16), 4732–4754.*
- Liu, S., Zhang, Z., Guan, C., Zhu, L., Zhang, M., and Guo, P. (2020). An improved fireworks algorithm for the constrained single-row facility layout problem. *International Journal of Production Research, 0(0), 1–19.*
- Meller, R. D., and Gau, K.-Y. (1996). The facility layout problem: recent and emerging trends and perspectives. *Journal of Manufacturing Systems, 15(5), 351–366.*

- Murray, C. C., Smith, A. E., and Zhang, Z. (2013). An efficient local search heuristic for the double row layout problem with asymmetric material flow. *International Journal of Production Research*, 51(20), 6129-6139.
- Ouaarab, A., Ahiod, B., and Yang, X. S. (2014a). Discrete cuckoo search algorithm for the travelling salesman problem. *Neural Computing and Applications*, 24(7-8), 1659-1669.
- Ouaarab, A., Ahiod, B., and Yang, X. S. (2014b). Improved and discrete cuckoo search for solving the travelling salesman problem. In *Cuckoo search and firefly algorithm* (pp. 63-84). Springer.
- Ouaarab, A., Ahiod, B., and Yang, X. S. (2015a). Discrete cuckoo search applied to job shop scheduling problem. In *Recent Advances in Swarm Intelligence and Evolutionary Computation* (pp. 121-137). Springer.
- Ouaarab, A., Ahiod, B., and Yang, X. S. (2015b). Random-key cuckoo search for the travelling salesman problem. *Soft Computing*, 19(4), 1099-1106.
- Payne, R. B., and Sorensen, M. D. (2005). *The cuckoos* (Vol. 15). Oxford University Press.
- Scalia, G., Micale, R., Giallanza, A., and Marannano, G. (2019). Firefly algorithm based upon slicing structure encoding for unequal facility layout problem. *International Journal of Industrial Engineering Computations*, 10(3), 349-360.
- Secchin, L. D., and Amaral, A. R. S. (2019). An improved mixed-integer programming model for the double row layout of facilities. *Optimization Letters*, 13(1), 193-199.
- Shlesinger, M. F., Zaslavsky, G. M., and Frisch, U. (1995). *Lévy flights and related topics in physics*.
- Singh, S. P., and Sharma, R. R. K. (2006). A review of different approaches to the facility layout problems. *The International Journal of Advanced Manufacturing Technology*, 30(5-6), 425-433.
- Srivastava, P. R., Chis, M., Yang, X. S., and Deb, S. (2012). An efficient optimization algorithm for structural software testing. *International Journal of Artificial Intelligence*, 8(S12), 68-77.
- Tompkins, J. A., White, J. A., Bozer, Y. A., and Tanchoco, J. M. A. (2010). *Facilities planning*. John Wiley & Sons.
- Tubaileh, A., and Siam, J. (2017). Single and multi-row layout design for flexible manufacturing systems. *International Journal of Computer Integrated Manufacturing*, 30(12), 1316-1330.
- Walton, S., Hassan, O., Morgan, K., and Brown, M. R. (2011). Modified cuckoo search: a new gradient free optimisation algorithm. *Chaos, Solitons & Fractals*, 44(9), 710-718.
- Wang, S., Zuo, X., Liu, X., Zhao, X., and Li, J. (2015). Solving dynamic double row layout problem via combining simulated annealing and mathematical programming. *Applied Soft Computing Journal*, 37, 303-310.
- Yang, X., Cheng, W., Smith, A. E., and Amaral, A. R. S. (2020). An improved model for the parallel row ordering problem. *Journal of the Operational Research Society*, 71(3), 475-490.
- Yang, X. S., and Deb, S. (2009). Cuckoo search via Lévy flights. *2009 World Congress on Nature and Biologically Inspired Computing, NABIC 2009 - Proceedings*, 210-214.
- Yang, X. S., and Deb, S. (2010). Engineering optimisation by cuckoo search. *International Journal of Mathematical Modelling and Numerical Optimisation*, 1(4), 330-343.
- Yang, X. S., and Deb, S. (2013). Multiobjective cuckoo search for design optimization. *Computers & Operations Research*, 40(6), 1616-1624.
- Yang, X. S., and Deb, S. (2014). *Cuckoo search: Recent advances and applications*. Neural Computing and

*Applications*, 24(1), 169-174.

Zhang, Z., and Cheng, W. (2014). *Decomposition strategies and heuristic for double row layout problem*. *Computer Integrated Manufacturing Systems*, 20(3), 559-568.

Zhang, Z., and Murray, C. C. (2012). *A corrected formulation for the double row layout problem*. *International Journal of Production Research*, 50(15), 4220-4223.

Zuo, X., Murray, C. C., and Smith, A. E. (2014). *Solving an extended double row layout problem using multiobjective tabu search and linear programming*. *IEEE Transactions on Automation Science and Engineering*, 11(4), 1122-1132.

Zuo, X., Murray, C. C., and Smith, A. E. (2016). *Sharing clearances to improve machine layout*. *International Journal of Production Research*, 54(14), 4272-4285.

# Bass Model 기반의 접이식 컨테이너 수요 예측

김학성<sup>1</sup>, 하헌구<sup>2</sup>

<sup>1</sup>한국철도기술연구원,

<sup>2</sup>인하대학교 물류전문대학원

## Demand Forecasting of Foldable Container with Bass Model

Hag-Seoung Kim<sup>1</sup>, Hun-Koo Ha<sup>2</sup>

<sup>1</sup>Logistics System Research Division, Korea Railroad Research Institute

<sup>2</sup>Graduate School of Logistics, INHA University

Trade imbalance among the nations or regions has been deepening as a result of economic globalization. This trade imbalance causes another imbalance, in the area of container supply and demand. That is, Asian countries including China suffer a shortage of containers for shipping their products, while the United States or EU countries have a surplus of empty containers after unloading the freight. To deal with this problem, over-supplied empty containers are moved to the locations where containers are needed, yet this is an inefficient solution to the problem. In an effort to resolve this inefficiency, many countries are preparing to develop and introduce foldable containers, which can be transported and stored in a bundle by folding 3 or 4 empty containers together, significantly enhancing the transport and storage efficiency of empty containers. However, the foldable container is still in its early stage, with test products produced or introduced to the market in Europe and the United States. In this paper, we studied demand forecasting for the foldable containers with Bass model which is based on the diffusion theory. For this purpose, the parameters of the Bass model were estimated, and the forecasting model of the foldable container demand was constructed and evaluated.

**Keywords** : Foldable container, Bass Model, Empty container repositioning

### 1. 서론

오늘날 세계 경제는 국가 간 국경 없는 교역을 통하여 성장하여 왔다. 이러한 세계 경제 성장과 더불어 늘어나는 교역량 증가에 따라 각 국가별, 대륙별 무역 불균형 또한 심화되어 가고 있으며 현재와 같이 중국 등 아시아 지역이 노동 집약적 생산 중심지로서의 역할이 지속 될 경우 더욱 늘어날 것으로 보인다. 이러한 무역 불균형에 따라 컨테이너 수급 역시 불균형을 이루고 있는데, 아시아 지역에서는 상품 수출을 위한 공 컨테이너가 부족하고 이와 반대로 미국과 유럽에서는 상품을 하역하고 난 뒤 공 컨테이너가 남아 공급이 과잉되고 있는 실정이다. 이로 인해 매년 수많은 공 컨테이너가 재배치(repositioning)되고 있는 상황으로 해운회사와 같은 컨테이너 운송 기관에서는 매년 막대한 비용을 들여 이를 해결하고 있다. 이에 공 컨테이너의 보관 및 운송 효율성을 획기적으로 개선할 수 있는 접이식 컨테이너(foldable container)가 개발 중에



있다. 접이식 컨테이너는 화물을 하역하고 난 뒤 빈 컨테이너를 접철하여 3개 혹은 4개의 접철된 컨테이너를 하나로 묶어 운송 및 보관할 수 있는 컨테이너를 말한다. 그러나 접이식 컨테이너는 Bandara(2015)가 이야기한 것과 같이 아직 유럽과 미국 회사의 몇몇 제품이 시운전(pilot testing) 또는 시장 진입(market introduction) 단계에 있다. 이에 본 연구에서는 물류산업에 효율성을 제고해 줄 수 있는 신제품인 접이식 컨테이너가 시장에서 본격적으로 상품화되어 확산하는 것에 대한 수요예측방법에 대하여 연구하고자 한다.

## 2. 수요예측 모형 및 추정 방법

많은 연구자들이 신기술을 적용한 신제품 개발과 새로운 서비스의 도입 및 확산에 대하여 혁신확산이론 기반의 수요예측 모형을 개발하고 이를 설명하고자 노력하여 왔다. 이러한 확산 모형에서 가장 널리 활용되고 있는 모형은 Bass 모형으로 이를 토대로 한 논문이 750편에 이를 정도이다(Ozkaya, 2008).

Bass 모형에서 혁신 채택자 즉, 신제품을 구매하는 그룹은 두 개의 집단으로 나누어진다고 가정한다. 이 중 하나의 집단은 외부영향이라 할 수 있는 대중매체 등의 영향을 받아 신제품을 구매하는 집단이고, 다른 집단은 내부영향이라 할 수 있는 기 구매자의 구전에 의해서만 영향을 받아 구매하는 집단이다. Bass 모형에서는 이러한 첫 번째 집단을 혁신자(innovators)로, 그리고 그 다음의 집단을 모방자(imitators)로 정의한다. 이를 근거로 Bass 모형에서 사용하는 모수는 세 개로 구성된다. 첫째, 신제품을 궁극적으로 구입할 고객의 크기를 나타내는 잠재수요 크기  $M$ , 그리고 임의의 고객인 혁신자가 제품을 구입하는데 영향을 미치는 외부적 영향을 나타내는 모수  $p$ 와 제품 구매자들로부터의 내부적 영향을 받아 구입하는 모방자가 구매하는 정도를 나타내는 모수  $q$ 가 있다. 이러한 모수 추정은 Bass 모형을 활용하여 신제품 확산 모형을 구축하는 첫 걸음이자 가장 중요한 단계라 할 수 있다.

Bass 모형의 모수를 추정하는 방식에 대하여 많은 연구자들이 제안 하였는데 그 중 Mahajan 등(1985)은 일반최소자승법(the ordinary least squares, OLS), 최우도추정법(the maximum likelihood estimation, MLE), 비선형최소자승법(the nonlinear least squares, NLS) 등의 모수 추정 방식을 비교 연구 하였다. 저자들은 NLS 즉, 비선형최소자승법이 가장 우수하다고 분석 하였으며 일반최소자승법(OLS)은 가장 쉽게 모수 추정을 할 수 있는 방법이라고 평가 하였다. 하지만 이러한 모수를 추정하기 위하여 필요한 과거 data는 신제품 혹은 신기술의 특성상 충분치 못한 경우가 대분이다. 이에 송영화 등(2005)은 Bass이론 등을 이용하여 위성 DMB 서비스에 대한 고객 수요예측 및 수용특성을 파악 할 때 위성 DMB 서비스와 같은 혁신 성향이 강한 신규 서비스의 수요예측은 고객의 성향에 따라 수용이 점진적으로 확산되며 확산단계별 고객 수용형태가 차별화 된다고 보았다. 따라서 Bass이론 등을 바탕으로 잠재 고객들을 대상으로 한 시장 조사 결과를 효과적으로 결합하여 신규 서비스에 대한 시장수용성과 변동성을 동태적으로 추정하였다. 박윤서(2005)는 기존의 Bass모형에 기반하여 확률이론을 바탕으로 종속적 관계를 가지는 상품들에 대한 수요확산 모형을 개발하고자 하였다. Bass의 hazard 함수 모형에 확률이론을 활용하여 확장 가능성을 확인하고 종속적 관계에 있는 신상품들에 대한 수요 확산모형을 개발하여 국내 이동전화와 무선인터넷 사례에 적용하여 실증적 분석을 수행하였다. 김우성(2007)은 미래 수요를 예측할 만한 데이터가 부족한 LCD의 국내 판매량 데이터를 예로 들어 잠재시장 규모와 그 도달시점을 예측하고자 하였다. 이를 위해 소수의 판매량 데이터로 예측이 가능하도록 수요 측 요인과 더불어 기업 측 요인을 반영한 확산 모형을 제시 하였다. 이봉주 등(2012)은 신규 사회서비스의 시장규모 예측을 위하여 Bass Model을 활용 하였다. 이를 위해 사회서비스 수요 설문 조사를 시행하고 Bass Model을 활용하여 아동발달 영역의 6개 유망사회서비스에 대하여 혁신 계수에 따른 서비스 확산 속도와 잠재 수요를 예측하였다. 채현석(2013)은 Bass 모형을 이용한 전기자동차 수요예측에 대한 모수 추정을 위해 유사추론방식으로 도요타 하이브리드 자동차의 세계 판매량 데이터를 이용하여 일반최소자승법에 의한 혁신계수와 모방계수를 추정 활용하였다.

이상에서 본바와 같이 Bass 모형 적용을 위한 모수 추정 방식은 과거 데이터가 충분하지 않은 관계로 여러 가지 방안이 활용되었으나, 본 연구에서는 접이식 컨테이너가 기존에 사용하고 있는 컨테이너의 특수한 형태임을 감안하여 유사추론방식을 활용하였다. 즉, 접이식 컨테이너와 유사한 특수 컨테이너의 판매량을 기초하여 모수를 추정하였다.

## 2-1 Bass 모형

Bass모형은 확률함수인 위험함수(hazard function)로부터 유도되는데 구매가 아직 일어나지 않았다고 주어진 경우 하나의 구매가 t 시점에 일어날 확률, 즉 t 시점까지 비 구매자로 남아있던 개인이 다음 시점에서 구매자가 될 확률을 말한다. 따라서 위험함수는 가입위험률(subscription hazard rate)이 선형함수로 표현된다는 가정하에 다음과 같이 정의된다.

$$\frac{f(t)}{1-F(t)} = p + qF(t) \quad \text{단 } F(0) = 0 \quad (1)$$

여기서,

$f(t)$ : t 시점에서의 구매에 대한 확률 밀도함수

$F(t)$ : t 시점까지의 구매에 대한 누적 확률밀도함수

p : 혁신 계수 (innovation coefficient)

q : 모방 계수 (imitation coefficient)

한편 잠재구매자 전체수를 M이라하고 시점 t에서의 구매자수를 N(t)라 할 때 다음 식이 성립된다

$$N(t) = MF(t) \quad (2)$$

또한 식 (2)로부터

$$f(t) = (p + qF(t))(1 - F(t)) \quad (3)$$

이 됨을 알 수 있으므로 시점 t에서의 구매자 수를 S(t)라 하면 다음과 같다.

$$\begin{aligned} S(t) &= \frac{dN(t)}{dt} = Mf(t) = M(p + qF(t))(1 - F(t)) \\ &= M(p + q\frac{N(t)}{M})(1 - \frac{N(t)}{M}) \\ &= (p + \frac{q}{M}N(t))(M - N(t)) \end{aligned} \quad (4)$$

이를 다시 이산화하여 정리하면 아래의 식과 같다.

$$S(t+1) = (p + \frac{q}{M}N(t))(M - N(t)) \quad (5)$$

여기서  $p(M - N(t))$ 은 혁신자(innovator)들에 의한 구매를 의미하고  $\frac{q}{M}N(t)(M - N(t))$ 은 모방자(imitator)들에 의한 구매를 의미한다.

한편 식 (3)을 미분방정식으로 풀면 다음과 같은 형태의 성장곡선 즉 누적분포함수(cumulative distribution function)가 도출된다.

$$F(t) = \frac{1 - e^{-(p+q)t}}{1 + \frac{q}{p}e^{-(p+q)t}} \quad (6)$$

또한 t 시점에서의 구매량인 S(t)를 시점 t로 미분한 식을 0으로 하는 시점  $T^*$ , 즉 최고 구매확률 시점은 아래와 같이 정리 된다.

$$T^* = -\frac{1}{p+q} \ln\left(\frac{p}{q}\right), \quad S(T^*) = \frac{1}{4q}(p+q)^2 \quad (7)$$

이와 같이 Bass 모형에 필요한 모수는 혁신 계수(p), 모방 계수(q) 및 잠재적 시장 규모(m) 등 이다. 앞서 Mahajan 등(1985) 밝힌 바와 같이 일반최소자승법(the ordinary least squares, OLS)이 가장 간단히 모수를 추정 할 수 있으며 비선형최소자승법(the nonlinear least squares, NLS)이 가장 정확하다.

## 2.2 Bass 모형의 모수 추정

### 1) 일반최소자승법

Bass(1969)는 혁신 확산 모형 발표 당시 일반최소자승법에 의한 모수 추정 방법을 제안 하였다. 저자는  $X(i)$ 를 시 구간(time interval)  $(t_{i-1}, t_i)$  에서 증가된 구매자들의 기대치 즉,  $X(i) = N(t_i) - N(t_{i-1})$ 라고 하고 다음과 같은 이산적 수식을 유추하였다.

$$\begin{aligned} X(i) &= pm + (q-p)N(t_{i-1}) - (q/m)N^2(t_{i-1}) + \varepsilon(i) \\ &= \alpha_1 + \alpha_2 N(t_{i-1}) + \alpha_3 N^2(t_{i-1}) + \varepsilon(i) \end{aligned} \quad (8)$$

단,  $\alpha_1 = pm$ ,  $\alpha_2 = (q-p)$ ,  $\alpha_3 = -q/m$ ,  $E[\varepsilon(i)] = 0$ ,  $var[\varepsilon(i)] = \sigma^2$ ,  $\varepsilon(i)$ 는 모든  $i \neq j$  에 대해  $\varepsilon(i)$ 와 독립이다. 이때 주어진  $\hat{\alpha}_1, \hat{\alpha}_2, \hat{\alpha}_3$  에 의하여 모수를 추정하면 다음과 같다.

$$\hat{p} = \frac{-\hat{\alpha}_2 + \sqrt{\hat{\alpha}_2^2 - 4\hat{\alpha}_1\hat{\alpha}_3}}{2} \quad (9)$$

$$\hat{q} = \frac{\hat{\alpha}_2 + \sqrt{\hat{\alpha}_2^2 - 4\hat{\alpha}_1\hat{\alpha}_3}}{2} \quad (10)$$

$$\hat{m} = \frac{-\hat{\alpha}_2 - \sqrt{\hat{\alpha}_2^2 - 4\hat{\alpha}_1\hat{\alpha}_3}}{2\hat{\alpha}_3} \quad (11)$$

일반최소자승법의 장점은 가장 손쉽게 적용 가능하다는데 있다. 이에 따라 많은 확산 모형에서 모수 추정에 사용되어 지고 있다. 하지만 Schmittlein 등(1982)은 일반최소자승법에 대해 3가지 문제점을 가지고 있다고 평가 하였다. 첫째, 불안정하거나 잘못된 부호를 가진 추정치를 얻기 쉬운데, 이는 주로 시계열 자료가 희소하고, 변수  $N(t_{i-1})$  와  $N^2(t_{i-1})$  사이에 다중공선성(multi-collinearity)이 존재할 때 발생한다. 둘째, 추정치들의 분포(표준오차)를 직접적으로 제공해 주지 않는데, 이는 모수 p, q, m이 추정량  $\alpha_1, \alpha_2, \alpha_3$ 의 비선형함수이기 때문이다. 셋째, 연속형 모형을 추정하는데 이산적 시계열자료를 사용하므로 시 구간 편의(time interval bias)가 발생할 수 있다. 즉, 이론적으로 추정식의 좌변은  $N(t)$ 의 도함수(derivative)이어야지  $X(t)$ 로 표현되는 차이(difference)는 아님으로,  $X(t)$ 는 변곡점 전까지의 시 구간(time interval)에 대해서는  $dN_t/dt$ 를 과소추정하고, 그 이후 부분에서는 과대추정하기 쉽다.

### 2) 비선형최소자승법

비선형최소자승법(NLS, the nonlinear least squares)은 최우도추정법(MLE, maximum likelihood estimation)의 단점을 보완하기 위하여 Srinivasan 등(1986)이 제안하였다.

먼저 Bass 모형의 누적분포함수를 이용하여 저자들은 혁신 계수 p, 모방 계수 q, 그리고 시장의 잠재적 규모인 m 이 세 가지 모수가 아래의 수식에 따라 시 구간(time interval)  $(t_{i-1}, t_i)$ 에서 증가된 구매자들의 기대치  $X(i)$ 를 이용해 추정 가능 하다고 하였다.

$$X(i) = m(F(t_i) - F(t_{i-1})) + \varepsilon_i \quad (12)$$

$$X(i) = m \left( \frac{1 - e^{-(p+q)t_i}}{1 + (q/p)e^{-(p+q)t_i}} - \frac{1 - e^{-(p+q)t_{i-1}}}{1 + (q/p)e^{-(p+q)t_{i-1}}} \right) + \varepsilon_i \quad (13)$$

여기서  $\varepsilon(i)$ 는 오차항이며 이에 오차 제곱합을 최소화하는 목적함수인 식(14)을 만족하는 혁신 계수 p, 모방 계수 q, 그리고 시장의 잠재적 규모인 m이 추정된다.

$$\min_{p, q, m} \sum_{i=1}^T [\hat{X}(i) - X(i)]^2 \quad (14)$$

비선형최소자승법은 일반최소자승법에 의하여 범 할 수 있는 오류인 시 구간편의(time interval bias)를 극복 할 수 있으며 더욱이 오차항은 외부 변수 즉, 경제 상황이나 마케팅 변수 등에 영향을 받지 않고 오직 표집오차(sampling error)에 의하여 영향을 받음으로 보다 현실적인 모수 추정이 가능하다.

### 3. 자료 수집 및 수요예측

#### 3-1 자료 수집

접이식 컨테이너 확산을 위한 Bass 모형을 구축하기 위해서는 앞서 살펴 본 바와 같이 혁신 계수(p), 모방 계수(q) 그리고 잠재적 시장 규모(m)에 대한 모수 추정이 필요하다. 모수 추정 과정은 일반최소자승법(OLS, the ordinary least squares)에 의하여 초기 값을 추정한 후 이를 근거로 하여 비선형최소자승법(NLS, the nonlinear least squares)을 활용하도록 한다.

먼저, 접이식 컨테이너는 Bandara 등(2015)이 밝힌 바와 같이 현재 시장에서 도입 초기에 있다. 이러한 이유로 Bass 모형의 모수 추정을 위한 직접적인 판매 자료(data)는 존재 하지 않는다. 따라서 일반최소자승법(OLS, the ordinary least squares)에 의한 모수추정을 위한 기초 자료는 유사추론방식을 적용하여 접이식 컨테이너와 같은 유사 제품인 특수 컨테이너 판매량을 조사하였다. 조사 대상 기업은 전 세계 컨테이너 판매량의 49 %를 차지하는 컨테이너 1위<sup>1)</sup> 제조업체인 CIMC(China International Marine Container Group Co., Ltd.)의 Annual report 및 Semi-Annual Report를 기반으로 조사하였다. 이를 근거로 중량물 수송을 위한 Flat Rack Container와 액체 수송용 탱크 컨테이너 등을 포함한 특수 컨테이너(Special Container) 판매량을 2002년부터 2014년까지 13년간의 판매량을 반기별로 다음과 같이 정리하였다.

Table 1. CIMC Semi-Annual Sales Volume of Special Container

(unit : box)

Year	2002		2003		2004	
	1st Half	2nd Half	1st Half	2nd Half	1st Half	2nd Half
Special Container	11,723	14,457	14,289	19,777	23,150	32,668
Year	2005		2006		2007	
	1st Half	2nd Half	1st Half	2nd Half	1st Half	2nd Half
Special Container	39,817	24,175	54,000	33,361	55,900	32,946
Year	2008		2009		2010	
	1st Half	2nd Half	1st Half	2nd Half	1st Half	2nd Half
Special Container	52,500	50,200	27,000	16,200	24,500	37,400
Year	2011		2012		2013	
	1st Half	2nd Half	1st Half	2nd Half	1st Half	2nd Half
Special Container	46,200	30,900	33,000	40,100	30,800	33,100
Year	2014					
	1st Half			2nd Half		
Special Container	31,500			38,000		

#### 3-2 모수 추정 및 수요예측

상기 특수 컨테이너(Special Container) 판매량에 대한 모수 추정을 위하여 전술한 바와 같이 식(8)를 이용하여 일반최소자승법에 의하여 모수를 추정 하도록 한다.

먼저, CIMC의 특수 컨테이너 판매량에 대한 SPSS를 활용한 회귀 분석 결과는 다음과 같다.

1) CCIA (China Container Industry Association), first half of 2007, <http://www.chinaccia.com>

Table 2. Result of Regression Analysis

계수 <sup>a</sup>								
모형		비표준화 계수		표준화 계수	t	유의확률	B에 대한 95.0% 신뢰구간	
		B	표준오차	베타			하한	상한
1	(상수)	19759.730	4849.934		4.074	.000	9726.877	29792.582
	Y	.089	.031	1.909	2.844	.009	.024	.153
	Y2	-9.716E-8	.000	-1.669	-2.486	.021	.000	.000
a. 종속변수: X								

이상의 회귀 분석 결과를 이용하여  $\alpha_1 = pm$ ,  $\alpha_2 = (q-p)$ ,  $\alpha_3 = -q/m$ ,  $E[\varepsilon(i)] = 0$ ,  $var[\varepsilon(i)] = \sigma^2$ ,  $\varepsilon(i)$ 는 모든  $i \neq j$ 에 대해  $\varepsilon(i)$ 와 독립으로 이때 주어진  $\hat{\alpha}_1 = 19,759.730$ ,  $\hat{\alpha}_2 = 0.089$ ,  $\hat{\alpha}_3 = -9.716E-8$ 에 대하여 일반최소자승법에 의하여 식(9), (10), (11)를 사용하여 일반최소자승법에 의한 모수를 추정하면 다음과 같다.

$$\hat{p} = 0.01802$$

$$\hat{q} = 0.10653$$

$$\hat{m} = 1,096,414.3$$

상기 일반최소자승법에 의한 모수 추정 결과를 초기 값으로 하여 비선형최소자승법에 의한 모수 추정을 위해 식 (14)를 이용하여 다음과 같이 잔차 제곱 합을 최소화하는 모수 추정 결과를 얻었다.

$$\hat{p} = 0.01639$$

$$\hat{q} = 0.10568$$

$$\hat{m} = 1,127,237.7$$

이상 두 가지 모수 추정 방식에 의하여 추정된 모수에 따른 오차 제곱 합을 구하면 다음 표와 같다.

Table 3. Sum of Squared Residual (SSR)

	Ordinary Least Squares	Nonlinear Least Squares
SSR	2,762,624,882	2,713,484,372

상기 결과에 따라 일반최소자승법 방식에 의한 모수 추정 보다 비선형최소자승법에 의한 모수 추정 결과가 좀 더 정확한 것을 알 수 있었다. 이에 상기 모수 추정 결과에 따른 접이식 컨테이너의 확산에 대하여 도입 초기 연도부터 50년간의 결과를 살펴보면 아래 그래프와 같다.

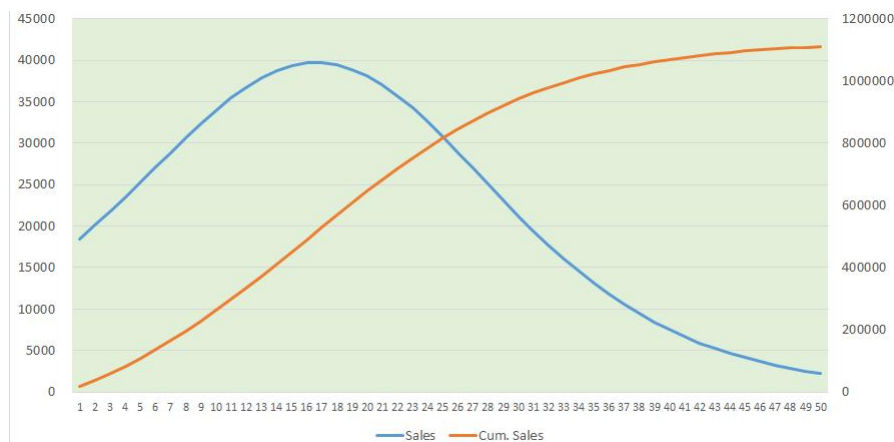


Figure 1. Sales Forecasting of Foldable Container

### 3-3. 수요예측 결과 검증

이상과 같이 구축된 Bass 모형에 의하여 예측된 값의 정확도를 평가하였다. 예측의 정확도를 평가 하는 지표로는 잔차제곱합(sum of squared residual : SSR), 평균평방오차(mean squared error : MSE), 평균절대편차 (mean absolute deviation : MAD), 평균 절대 백분율 오차 (mean absolute percentage error : MAPE), 평균자승백분율오차(mean squared percentage error : MSPE), 근평균제곱오차 (root mean squared error : RMSE) 평균자승백분율오차의 평방근 (root mean squared percentage error : RMSPE) 등이 있다(이종원, 2006)

한편 이종원(2006)은 타일(Theil)의 U-통계량을 이용하여 예측모형의 정확도를 평가 할 수 있다고 하였는데, 타일(Theil) U-통계량 값이 1이면 특정한 모형의 예측력이 단순 예측 방법과 동일함을 의미하고 U 값이 1보다 작으면 단순예측방법보다 예측력이 우수하며, U 값이 1보다 크면 단순예측방법보다 예측력이 낮은 것으로 설명하였다. 이러한 점을 고려 한 타일(Theil) U-통계량은 다음과 같은 식으로 표현된다.

$$U = \sqrt{\frac{\sum (\frac{F_{t+1} - A_{t+1}}{A_t})^2}{\sum (\frac{A_{t+1} - A_t}{A_t})^2}} \quad (15)$$

여기서, A = 실제 값, F = 예측 값이며 상기 식에 의한 Bass 모형의 타일 U-통계량은 0.9537으로 나타났다. 이를 근거로 보았을 때 단순예측방법 보다 본 연구에서 구축된 Bass 모형이 상대적으로 우수한 것으로 판단된다. 이와 함께 일반적으로 사용하고 있는 예측방법인 지수평활법 등과 비교하기 위하여 대표적인 지수평활법인 단순지수평활법(single exponential smoothing)과 윈터스 지수평활법(Winter's exponential smoothing)를 활용하여 잔차제곱합을 비교하였다.

먼저, 단순지수평활법은 전기의 실제 값과 예측 값의 차이인 잔차의 일정비율에 전기의 예측 값을 더하는 방법으로 다음 식과 같다.

$$F_{t+1} = F_t + \alpha (A_t - F_t) \quad (16)$$

$$\text{단, } 0 < \alpha < 1$$

여기서 A = 실제 값, F = 예측 값이며  $\alpha$ 는 평활계수이다. 다음으로 윈터스 지수평활법은 가법(additive seasonality)과 승법(multiplicative seasonality)으로 구분된다. 윈터스 가법 모델은 계절 변동이 있는 시계열이 시간 흐름과 관계없이 일정한 변동을 갖는 수치에 대한 예측 기법이며, 윈터스 승법 모델은 시계열이 가진 수평, 추세, 계절 패턴에 대한 모수(parameter)를 각각 평활하는 세 방정식을 연립해 구한 값을 윈터스 평활모델의 방정식에 대입해 예측하는 방식이다. 본 연구에서는 윈터스 승법모델을 사용하였으며 이를 수식으로 나타내면 아래와 같다.

$$S_t = \alpha \frac{X_t}{I_{t-L}} + (1-\alpha)(S_{t-1} + b_{t-1}) \quad (17)$$

$$b_t = \gamma(S_t - S_{t-1}) + (1-\gamma)b_{t-1} \quad (18)$$

$$I_t = \beta \frac{X_t}{S_t} + (1-\beta)I_{t-L} \quad (19)$$

$$F_{t+m} = (S_t + b_t m)I_{t-L+m} \quad (20)$$

여기서 m = 예측기간, L = 계절 주기,  $X_t$  = 실제 값,  $I_t$  = 계절 패턴 평활식,  $b_t$  = 추세 패턴 평활식,  $S_t$  = 계절성이 없는 수평 패턴 평활식이다. 이상의 대표적인 지수평활법에 의하여 앞서 제시한 정확도 비교 방법 중 Bass 모형과 같은 기간 예측 한 후 잔차제곱합(SSR)을 계산 한 결과 아래와 같이 Bass 모형에 의한 SSR 값이 가장 적은 결과를 보여 Bass 모형에 의한 예측이 좀 더 우수한 것을 확인하였다.

Table 4. Comparison of sum of squared residual : SSR

	Bass Model	Single Exponential Smoothing	Winter's exponential smoothing
SSR	2,713,484,372	3,342,268,989	3,805,937,507

#### 4. 결론 및 시사점

본 연구에서는 국가 간 교역 불균형에 의하여 심화되어 가고 있는 공 컨테이너 재배치로 인한 비효율을 제거하기 위하여, 현재 각국에서 개발되고 있는 접이식 컨테이너에 대한 수요예측을 Bass 모형을 기반으로 구축하고 이를 평가하였다. 이를 위해 유사추론방식을 적용하여 CIMC 특수 컨테이너 판매실적을 기반으로, 일반최소자승법을 사용하여 초기 모수를 추정한 후, 비선형최소자승법으로 혁신 계수(p), 모방 계수(q) 및 잠재적 시장 규모(m) 등의 모수를 추정하였다. 또한 테일의 U-통계량을 통해 수요예측모형을 평가한 결과 단순예측방법 보다 설명력이 우수한 것을 확인하였다. 이와 더불어 일반적인 시계열 예측방법과 비교하고자 단순지수평활법과 윈터스 지수평활법을 사용하여 각각 잔차제곱합 값을 구한 후 비교하였다. 이를 통해 Bass 모형을 기반으로 한 수요예측 모형이 타 예측 방법에 비해 좀 더 정확하게 접이식 컨테이너 확산에 따른 수요를 예측 할 수 있는 것으로 나타났다.

새로운 기술과 서비스에 기반 한 신제품 개발은 산업의 발전과 경제 성장에 매우 중요한 역할을 담당하고 있다. 그러나 실제 존재하지 않는 새로운 제품에 대한 수요예측은 자료 부족 등의 이유로 인하여 정확성도 낮은 것이 사실이다. 이에 본 연구에서는 유사추론방식에 의하여 모수를 추정하고 이를 근거로 수요를 예측해 보았다. 본 연구의 결과는 물류산업의 효율성을 제고할 수 있는 접이식 컨테이너의 사업화 추진 시 생산 설비 등의 과잉 투자 혹은 과소 투자에 의한 기회 손실 등의 문제를 초래하지 않도록 실무분야에서 많은 응용이 기대된다. 또한 접이식 컨테이너 도입과 활용에 따른 관련 산업계에 미치는 영향과 파급효과를 사전에 검토 할 수 있는 토대를 제공함으로써 해운 관련 기관의 경영 전략과 국가 해운 활성화 정책 수립에 활용 될 수 있을 것이다.

또한, 향후 접이식 컨테이너 생산이 본격화되고 경영노력 등에 의하여 가격이나 광고 등 마케팅 활동이 활발해지게 되면 이에 따른 수요예측방법도 차별화하여 연구 할 필요성이 있을 것이다.

#### 감사의 글(Acknowledgement)

본 연구는 국토교통부 교통물류연구사업의 연구비지원(20TLRP-C126986-04)에 의해 수행되었습니다.

[This research was supported by a grant(20TLRP-C126986-04) from Transportation logistics R&D Program funded by Ministry of Land, Infrastructure and Transport of Korean government.]

## 참고문헌

- 김우성. (2007). '신상품 수요예측을 위한 Bass모형 확장에 관한 연구', 박사학위논문, 동국대학교.
- 박윤서. (2005). 종속적 신상품의 수요확산모형: 무선인터넷 사례를 중심으로. 한국경영과학회 2005년 춘계 학술대회논문집, 1082-1088.
- 송영화, & 한현수. (2005). 혁신채택 및 확산이론의 통신방송융합 (위성 DMB) 서비스 수요추정 응용. 경영과학, 22(1), 179-197.
- 이봉주, 김남희, & 서정민. (2012). 사회서비스 수요예측을 위한 Bass 확산모형 적용 가능성에 대한 탐색적 연구: 아동분야 유망사회서비스 유형을 중심으로. 한국사회복지행정학, 14(2), 27-55.
- 채현석. (2013). 'Bass 확산모형을 이용한 전기자동차 수요예측', 석사학위논문, 경북대학교.
- Bandara, Y. M., Garaniya, V., Chin, C., & Leong, Z. H. (2015). Improving Logistics Management Using Foldable/Collapsible Containers: A Case Study. The Asian Journal of Shipping and Logistics, 31(1), 161-185.
- Bass, F. M. (1969). A New Product Growth Model for Consumer Durables. Management Sciences. Institute for Operations Research and the Management Sciences. Evanston, XV (5).
- CCIA (China Container Industry Association), first half of 2007. ([www.chianccia.com](http://www.chianccia.com))
- CIMC ([www.CIMC.com](http://www.CIMC.com)) : CIMC/Investor Relationship/PeriodicReport /Annual Report/Semi-annual Report. 2002 - 2014
- Mahajan, V., Mason, C. H., & Srinivasan, V. (1985). An evaluation of estimation procedures for new product diffusion models. Graduate School of Business, Stanford University.
- Mahajan, V., Muller, E., & Bass, F. M. (1995). Diffusion of new products: Empirical generalizations and managerial uses. Marketing Science, 14(3\_supplement), G79-G88.on)
- Ozkaya, E., Demand management in global supply chains, Ph. D. Thesis, Georgia Institute of Technology, 2008.
- Schmittlein, D. C., & Mahajan, V. (1982). Maximum likelihood estimation for an innovation diffusion model of new product acceptance. Marketing science, 1(1), 57-78.
- Srinivasan, V., & Mason, C. H. (1986). Technical note-nonlinear least squares estimation of new product diffusion models. Marketing science, 5(2), 169-178.



## 물류과학기술연구 제1권 제1호(통권 제1호)

- 
- 인 쇄 : 2020년 12월 21일
  - 발 행 : 2020년 12월 24일
  - 발행인 : 하 현 구
  - 편집인 : 김 태 복
  - 발행처 : (사)한국물류과학기술학회(<https://www.klst.or.kr>)  
경기도 의왕시 철도박물관로 176  
전화 : 031-460-5873
- 

(비매품)

□ 이 학술지의 저작권은 (사)한국물류과학기술학회가 갖고 있습니다.

Copyright 2020. The Society of Logistics Science and Technology all rights reserved.