JDCS
디지털콘텐츠학회논문지
Journal of Digital Contents Society
Vol. 25, No. 11, pp. 3515-3523, Nov. 2024

Check for updates

수요응답형 교통수단 스케줄링 알고리즘의 시각적 분석

김 동 규[1] · 신 동 화[2*]
[1]광운대학교 컴퓨터정보공학부 학사과정
[2]광운대학교 컴퓨터정보공학부 조교수

Visual Analysis of Demand-Responsive Transport-Scheduling Algorithm

Dongkyu Kim[1] · DongHwa Shin[2*]

[1]Bachelor's Course, Department of Computer and Information Engineering, Kwangwoon University, Seoul 01897, Korea
[2]Assistant Professor, Department of Computer and Information Engineering, Kwangwoon University, Seoul 01897, Korea

[요 약]

수요응답형 교통체계(DRT)는 기존 대중교통 시스템의 한계를 보완하며 교통 사각지대 문제를 해결하는 데 중요한 역할을 하고 있다. DRT의 효과적 운영을 위해서는 승객들의 요청에 대해 어떤 DRT 차량을 배정해야 하는지에 대한 결정을 최적화하는 것이 중요하다. 많은 연구자가 이러한 DRT 시스템의 스케줄링을 연구하고 있지만, 알고리즘의 수행 과정을 효과적으로 이해, 분석하기 위한 도구는 부재하다. 본 연구는 이러한 DRT 시스템의 스케줄링 알고리즘 연구자가 알고리즘 수행 과정을 탐색적으로 분석할 수 있도록 하는 대화형 시각화 시스템을 제안한다. 이 시스템은 승객 요청 분포, 교통 네트워크, 그리고 스케줄링 알고리즘 수행 과정을 확인할 수 있는 시각화 모듈로 이루어져 있으며, 상호작용 기법을 통해 연구자들이 알고리즘 수행 과정을 능동적으로 탐색할 수 있도록 한다. 제안하는 시스템에 대한 평가로서, 시스템을 활용하여 알아낼 수 있었던 스케줄링 알고리즘 분석 결과들을 밝힌다.

[Abstract]

Demand-responsive transport (DRT) systems address the limitations of traditional public transportation and play a key role in solving mobility challenges in underserved areas. Efficient DRT operations rely on optimizing vehicle-assignment decisions in response to passenger requests. Although considerable research has been conducted on DRT scheduling, tools for effectively understanding and analyzing the execution processes of these algorithm are lacking. Therefore, this study proposes an interactive visualization system that allows researchers to explore and analyze DRT scheduling algorithms. It includes visualization modules for passenger-request distribution, the transportation network, and algorithm execution. These interactive features can allow researchers to actively investigate the DRT scheduling process. Additionally, experimental evaluations of the proposed system demonstrate its effectiveness in providing key insights into DRT scheduling algorithms.

색인어 : 수요응답형 교통수단, 스케줄링 알고리즘, 정보 시각화, 시각적 분석, 인간-컴퓨터 상호작용

Keyword : Demand-Responsive Transport, Human-Computer Interaction, Information Visualization, Scheduling Algorithm, Visual Analysis

http://dx.doi.org/10.9728/dcs.2024.25.11.3515

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

Received 31 August 2024; **Revised** 07 October 2024
Accepted 14 October 2024

***Corresponding Author; DongHwa Shin**

Tel: +82-2-940-8474
E-mail: dhshin@kw.ac.kr

Ⅰ. 서 론

현대 사회에서 이동성(mobility)은 인간 활동의 필수적인 요소로 자리 잡고 있다. 도시화와 함께 인구 밀집도가 증가함에 따라 교통체계는 더욱 복잡해지고 있으며, 이러한 상황에서 효율적이고 유연한 대중교통 시스템의 필요성이 더욱 주목받고 있다. 전통적인 대중교통 시스템은 일정한 경로와 일정에 따라 운영되기 때문에, 다양한 지역의 이동 수요를 완전히 충족시키는 데 한계가 있다. 특히, 교통 사각지대에 있는 지역은 대중교통망의 접근성이 떨어지며, 이에 따라 주민들은 이동권에서 소외되는 상황이 발생한다. 이러한 교통 사각지대 문제를 해결하기 위해 새로운 교통 모델에 관한 연구가 활발히 진행되고 있으며, 그중 수요응답형 교통체계(DRT; Demand Responsive Transport)는 매우 주목받고 있는 혁신적인 해결책이다[1].

DRT는 승객이 필요로 할 때 직접 요청을 통해 차량을 호출하고, 요청에 따라 소형 승합차 또는 유사한 차량을 운행하는 방식으로, 전통적인 대중교통 시스템이 제공하지 못하는 유연성과 효율성을 제공한다. DRT 시스템은 전화나 모바일 애플리케이션을 통해 운영되며, 승객은 자신이 원하는 시간과 장소에서 차량을 호출할 수 있다. 이 시스템은 교통 사각지대에 있는 지역 주민들의 이동성을 향상하는 데 이바지하며, 이에 따라 사회적 이동 격차를 줄이는 중요한 역할을 한다.

실제 사례로, 청주에서 도입된 '콜버스'라는 이름의 DRT 시스템이 있다[2]. 이 시스템은 성공적인 운영을 통해 운영비용 절감, 손실 부담금 감소, 승객 만족도 증가 등의 긍정적인 성과를 거두었다. 이러한 성과는 DRT 시스템이 기존의 대중교통 시스템을 보완하고, 특정 지역에서의 교통 문제를 해결할 수 있는 잠재력을 보여준다. 그러나 이러한 성과에도 불구하고, DRT 시스템의 효과적인 운영을 위해서는 해결해야 할 과제가 여전히 존재한다.

큰 과제 중 하나는 여러 지역에서 발생하는 다양한 승객 요청들을 어떻게 최적의 방식으로 처리할 것인가 하는 문제이다. DRT 시스템에서는 각 차량이 실시간으로 발생하는 승객 요청에 신속하게 대응해야 해서, 차량의 스케줄링 문제는 매우 복잡한 최적화 문제로 귀결된다. 다시 말해, 들어오는 요청에 대해 가용할 수 있는 DRT 차량 중 어느 차량을 배정할 것인지에 대한 문제가 있다.

기존에 여러 DRT 스케줄링 알고리즘이 연구됐으며, 현재도 DRT의 활성화를 위해 활발히 연구되고 있다[3]-[5]. 그러나 더 나은 DRT 스케줄링 알고리즘을 효과적으로 연구, 개발하기 위해서는 스케줄링 알고리즘이 수행되는 각 과정과 절차에 대한 자세한 관찰과 분석이 필요하다. 스케줄링의 결과와 성능만을 파악하기보다는, 그 과정에 대한 자세한 분석을 통해서 왜 그러한 결과가 도출되었는지 잘 이해할 수 있다면, 그를 통해 얻은 통찰을 활용하여 더 나은 알고리즘을 개발할 수 있는 단초를 마련할 수 있는 것이다.

따라서 본 연구에서는 DRT 스케줄링 알고리즘의 자세한 수행 과정을 연구자가 효과적으로 확인할 수 있는 대화형 시각화 시스템을 제안한다. 정보 시각화 및 시각화 시스템은 매우 다양한 분야에서 데이터를 분석하는 데에 활용되고 있다[6]-[8]. 그와 같이 우리의 시스템을 통해 연구자들이 스스로 고안한 다양한 스케줄링 알고리즘을 평가하고 분석할 수 있도록 하는 것이다.

본 논문의 구성은 다음과 같다. 2장에서는 시스템 개발 및 검토에 활용한 데이터의 명세와 그 처리 방법을 소개하며, 3장에서는 우리가 제안하는 대화형 시각화 시스템 내의 여러 시각화에 관해 소개한다. 4장에서는 이를 토대로 도출할 수 있었던 분석 시나리오 및 결과를 밝힌다. 5장에서는 본 연구의 결론 및 향후 진행할 수 있는 연구 계획을 서술하며 마무리한다.

Ⅱ. 데이터 명세 및 DRT 스케줄링 알고리즘

본 연구에서 진행하는 모든 DRT 스케줄링에 대한 분석은 교통 계획 연구 분야에서 매우 잘 알려진 수폴스 대중교통 네트워크(Sioux Falls transportation network)[9] 내에서 이뤄진다. 이 네트워크는 실제 도시 교통망의 축소판으로, 다양한 스케줄링 및 최적화 알고리즘을 평가하는 데 유용한 표준 데이터 셋이다. 또한 스케줄링을 수행하기 위해서 임의로 생성한 승객들의 요청 데이터를 활용한다. DRT 스케줄링 알고리즘은 이렇게 주어진 승객 요청 데이터를 대상으로, 요청에 대해 여러 DRT 차량 중 한 대를 선택하여 배정함으로 목적식을 최적화하는 것으로 생각할 수 있다. 이 데이터에 대한 자세한 설명은 이번 장에서 후술한다.

2-1 교통 네트워크 데이터

수폴스 네트워크는 총 24개의 노드(node)와 76개의 링크(link)로 구성된다. 수폴스 네트워크 내에서 DRT 차량이 승객을 태우거나 내릴 수 있는 지점은 각 노드로 표현된다. 각 노드는 X, Y 좌표를 가지며, 이 좌표들은 네트워크 내에서 해당 노드의 위치를 지정하는 데 사용된다. 또한, 링크는 노드와 노드를 연결하는 개념으로 실제 연결된 두 노드 간 소요되는 이동 시간을 가중치로 갖는다.

하지만 모든 노드 간에 링크가 존재하는 것은 아니다. 따라서 직접적으로 링크가 존재하지 않더라도, 특정한 노드에서 다른 노드로 최단 시간을 들여 이동할 수 있는 경로를 구하고, 해당 경로를 구성하는 링크들의 가중치의 총합을 계산하였다. 이는 곧 출발 노드(origin node)에서 도착 노드(destination node)까지의 이동 시간이자, 비용이며 분 단위로 기록하였다. 이 과정에서 최단 시간이 걸리는 경로를 구하기 위해 매우 잘 알려진 다익스트라 알고리즘[10]을 활용하였다. 이러한 모든

노드로부터 다른 모든 노드까지의 비용을 모두 포괄하여 우리는 OD(Origin-Destination) 비용 데이터라고 칭한다.

2-2 승객 요청 데이터

DRT의 잠재적 승객들은 이 24개의 노드 중 하나에서 DRT에 탑승하여 다른 23개의 노드 중 하나로 이동을 희망하는 요청을 보낸다. 본 연구에서는 분석 대상 요청 개수로 총 80개를 사용하였으며, 약 1시간가량의 시간 동안 발생할 것을 상정하여 데이터를 임의로 생성하였다. 각 요청은 다음과 같은 필드를 포함한다.

표 1. 승객 요청 데이터의 필드
Table 1. Data fields of passenger request datset

Attribute Name	Description	Data
id	Unique ID of the request	1
request_hour	Hour when the passenger made the request (0-23)	11
request_min	Minute when the passenger made the request (0-59)	7
from_node_id	ID of the node where the passenger starts their journey	1
to_node_id	ID of the node where the passenger ends their journey	2
population	Number of passengers associated with the request	3
d1_hour	Earliest allowed departure hour (0-23)	11
d1_min	Earliest allowed departure minute (0-59)	7
d2_hour	Latest allowed departure hour (0-23)	11
d2_min	Latest allowed departure minute (0-59)	17
a1_hour	Earliest allowed arrival hour (0-23)	11
a1_min	Earliest allowed arrival minute (0-59)	11
a2_hour	Latest allowed arrival hour (0-23)	11
a2_min	Latest allowed arrival minute (0-59)	31

표 1은 승객이 DRT 서비스를 이용하여 특정 노드에서 다른 노드로 이동을 요청하는 상황을 나타내는 여러 데이터 필드를 포함한다. 각 요청은 고유한 ID와 함께, 요청이 이루어진 시간(시, 분), 출발지 노드와 도착지 노드의 ID, 그리고 해당 요청에 포함된 승객 수(population) 등의 필드를 포함한다. 이 데이터는 DRT 차량이 요청을 처리하는 데 있어 중요한 요소들을 제공하며, 특히 population 필드는 DRT 차량의 수용 가능 인원을 고려할 때 핵심적인 역할을 한다. 이는 요청이 차량의 적재 한도를 초과하지 않도록 하는 데 필수적인 정보이기 때문이다. 추가로, 표 1에는 각 요청에 대한 탑승 및 하차의 허용 시간대(d1_hour, d1_min, d2_hour, d2_min, a1_hour, a1_min, a2_hour, a2_min)가 명시되어 있어, DRT 스케줄링 알고리즘이 최적의 경로와 시간대를 계산하는 데 활용된다.

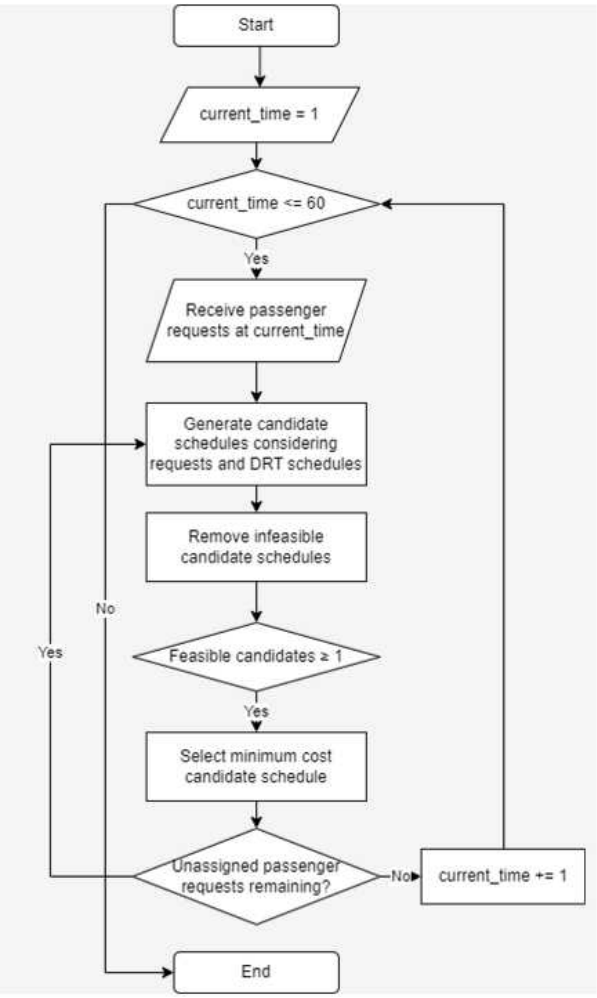

그림 1. 스케줄링 알고리즘 구성도
Fig. 1. Scheduling algorithm flowchart

2-3 스케줄링 알고리즘

본 연구는 DRT 스케줄링 알고리즘의 실행 과정을 분석하기 위한 연구이다. 따라서, 분석의 대상이 되는 DRT 스케줄링 알고리즘의 선정이 필요하다. 본 절에서는 우리가 활용한 DRT 스케줄링 알고리즘을 간단히 밝힌다.

DRT 차량은 모두 고유의 스케줄을 가질 수 있다. 스케줄이란 어느 노드로 이동하여 승객을 태우고, 다음 일정으로 또 다른 노드로 이동하여 승객을 내리거나 태우는 등의 일련의 과정을 의미한다. 이때 새로운 요청이 들어온다면, DRT 차량은 해당 요청을 받아서 처리하기 위해 기존 스케줄을 조정해야 할 필요가 생길 것이다.

예를 들어, 기존에 ID 1번 DRT 차량이 노드 번호 <3-4-2-5> 순으로 이동하는 스케줄을 가졌다고 하자. 또한, 새로운 요청은 6번 노드에서 승객을 태우고 8번 노드에서 승객을 내려야 하는 요청이라고 하자. 그러면 이 DRT는 기존 스케줄을 모두 마무리하고 새로운 요청을 처리하는, 즉, <3-4-2-5-6-8>이라는 새로운 스케줄을 고려할 수 있을 것

그림 2. 대화형 시각화 시스템의 유저 인터페이스
Fig. 2. User interface of the interactive visualization system

이다. 또는 기존 스케줄들을 모두 뒤로 미루고는 우선 새로운 요청을 처리하는, <6-8-3-4-2-5> 와 같은 새로운 스케줄을 가질 수도 있을 것이다. 아니면 적절히 기존 스케줄의 중간에 새로운 일정을 끼워 넣는 방식의 경우의 수도 생각해볼 수 있다.

다만, 상술한 바와 같이 스케줄 후보를 만드는 과정에서 후보로서 제한이 되는 경우가 있다. 이러한 제한은 주로 세 가지 주요 요인으로 발생한다. 첫째, 차량의 최대 탑승 인원 초과로 인한 제한이 있다. 차량의 현재 탑승 인원과 새로운 요청의 승객 수를 더했을 때, 차량의 최대 탑승 인원을 초과하면 해당 스케줄은 유효하지 않다. 둘째, 요청 시간 제약으로 인해 제한이 발생할 수 있다. 요청된 승객의 출발 또는 도착 시간이 허용된 시간 범위를 초과할 경우, 해당 스케줄은 유효하지 않게 된다. 마지막으로, 스케줄링 과정에서 탑승 및 하차 시점의 오류가 있거나 최종적으로 차량의 탑승 인원이 0이 되지 않는 경우, 해당 스케줄은 실패한 것으로 간주된다. 이러한 다양한 제한 사항들은 후보 스케줄을 제한하는 주요 원인으로 작용하며, 스케줄링 과정에서 유효한 후보를 찾는 데 어려움을 더하게 된다.

본 연구에서 활용한 알고리즘은 이러한 방식으로 만들 수 있는 모든 경우의 수를 전부 생성하여 고려 대상에 넣는 방식이다. 이렇게 DRT의 기존 스케줄에서 새로운 요청이 들어옴에 따라 고려할 수 있는 새로운 후보 스케줄들을 다수이며, DRT 차량이 한 대가 아니라 그 이상이기 때문에 모든 DRT의 가능한 모든 후보 스케줄들을 다 고려하여 그 중 비용이 최저인 것을 골라야 하는 것이다. 이러한 방식으로 시간대별로 들어오는 모든 요청을 알고리즘에 따라 수행하는 과정에서 수많은 후보 스케줄들이 고려될 것이다. 본 연구는 그러한 스케줄들을 모두 효과적으로 나타내고, 연구자들이 자세히 파악할 수 있도록 하는 것을 목표로 한다.

2-4 스케줄링 과정 데이터

스케줄링 과정 데이터는 각 DRT 차량이 특정 요청을 처리할 때 발생하는 경로 및 시간 정보를 포괄적으로 담고 있는 데이터이며, 상술한 DRT 스케줄링 알고리즘의 수행을 통해 얻는다. 이 데이터는 특정 시간대에 주어진 요청에 대해 DRT 차량 중 어떤 차량이 선택되고, 또한 다양한 경로 후보 중에서 어떤 경로를 선택하는지를 나타낸다. 스케줄링 데이터는 후보 경로(candidates), 각 후보 경로에 대한 비용(cost), 경로 세부 사항(rq_detail), 그리고 해당 경로를 수행하는 DRT 차량(drt)에 대한 정보를 포함한다.

각 후보 경로는 해당 요청을 처리하기 위한 여러 가능성 중 하나로, 경로의 시작과 끝을 포함한 이동 시각(s, e), 이동 경로의 출발 노드(start_node)와 도착 노드(target_node), 그리고 그 경로가 승객의 요구사항을 만족시키는지(satisfied)를 나타낸다. 이러한 경로 후보들은 그에 따르는 비용(cost)에 따라 평가된다.

비용은 주로 경로의 길이, 이동 시간, 승객 만족도 등을 고려하여 산출되지만, 본 연구에서는 우선 이동 시간만을 고려하였다.

스케줄링 데이터는 또한 각 경로 후보를 수행할 DRT 차량에 대한 정보를 포함한다. 이 정보는 각 차량이 현재 위치에서 해당 경로를 수행할 수 있는지, 그리고 차량의 현재 상태

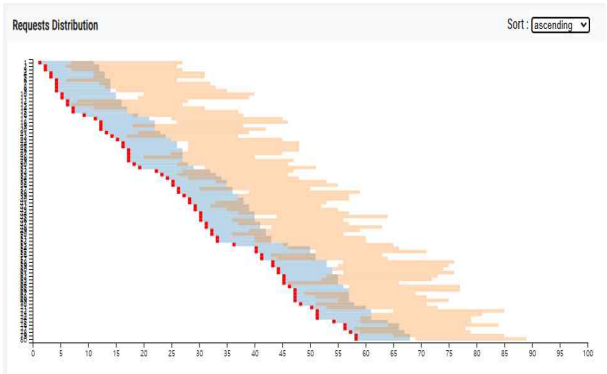

그림 3. 승객 요청 분포 시각화. X축은 시간, Y축은 요청 ID를 의미함.

Fig. 3. Request distribution view. X-axis represents time, and the Y-axis represents the request ID.

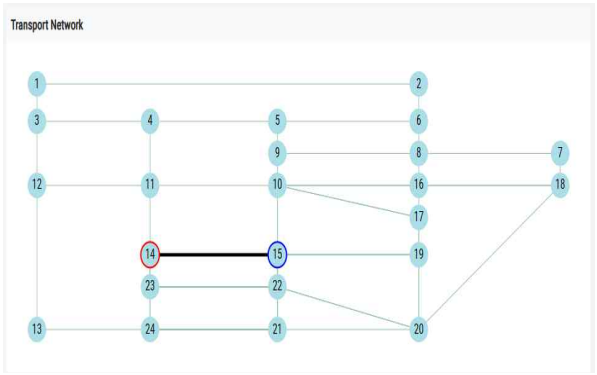

그림 4. 교통 네트워크 시각화. 14번과 15번 노드는 각각 출발, 도착 노드를 의미함.

Fig. 4. Transport network view. Nodes 14 and 15 represent the origin and destination nodes, respectively.

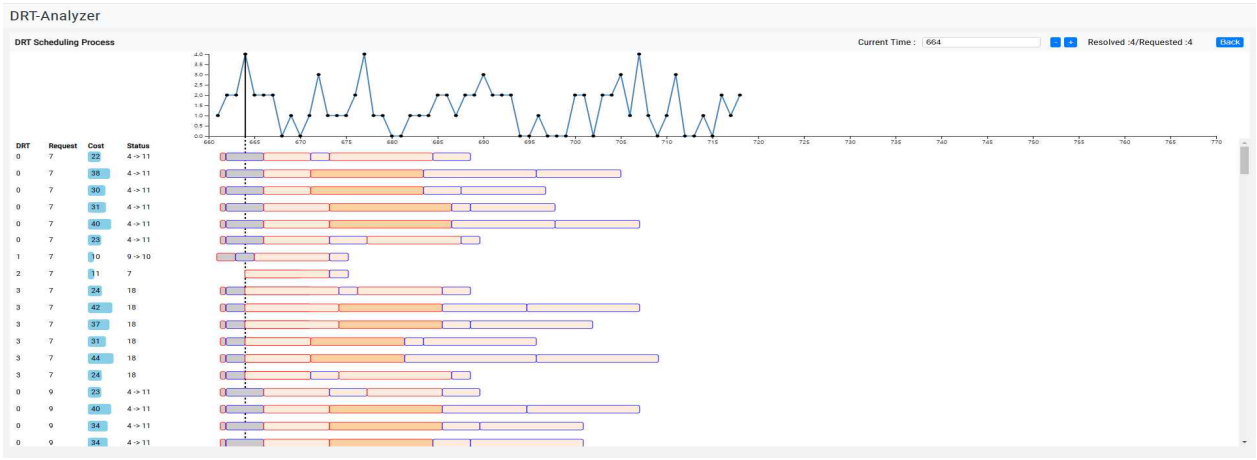

그림 5. DRT 스케줄링 과정 시각화

Fig. 5. DRT scheduling process view

에 따라 얼마나 효율적으로 해당 요청을 처리할 수 있는지를 평가하는 데 활용된다. 최종적으로 chosen_drt라는 필드는 이러한 경로 후보들 중에서 최적의 경로를 수행할 차량을 선택하는데, 이는 시스템이 가장 낮은 비용으로 승객의 요구를 만족시킬 수 있는 경로를 선택했음을 의미한다.

따라서, DRT 스케줄링 데이터는 각 DRT 차량이 어떤 경로를 따라 승객을 이동시키며, 그 과정에서 발생하는 모든 관련 정보를 제공함으로써, 스케줄링 알고리즘이 어떻게 작동하는지에 대한 모든 정보를 담고 있다. 이러한 데이터를 효과적으로 탐색할 수가 있다면 연구자는 DRT 시스템의 효율성을 높이기 위한 다양한 개선점을 도출할 수 있을 것이다.

Ⅲ. 시각화 및 상호작용 기법

본 연구에서 제시하는 수요응답형 교통수단 스케줄링 알고리즘 분석을 위한 시각화 시스템은 웹 기반 애플리케이션으로, React 프레임워크[11]를 기반으로 구현되었다. 그림 2는 제시하는 시각화 시스템의 전체 화면에 해당한다. React는 웹 프론트엔드(frontend) 개발에 널리 사용되는 프레임워크로, 사용자 인터페이스를 효율적으로 구성할 수 있는 장점이 있다. 또한, 시각화 및 상호작용 기능 구현을 위해 D3.js[12]를 활용하였다.

D3.js는 데이터 기반의 문서 조작을 가능하게 하는 라이브러리로 복잡한 데이터셋을 시각적으로 표현하는 데 유용하다.

3-1 승객 요청 분포 시각화

승객 요청 분포 시각화(requests distribution view)는 DRT 시스템에서 발생하는 승객 요청들의 시간적 분포를 단순화된 형태로 보여준다(그림 3). 이 시각화는 연구자들이 특정 시간대에 요청이 집중되는 패턴을 빠르게 파악할 수 있도록 하여, 서비스 개선과 스케줄링 최적화를 위한 중요한 정보를 제공한다.

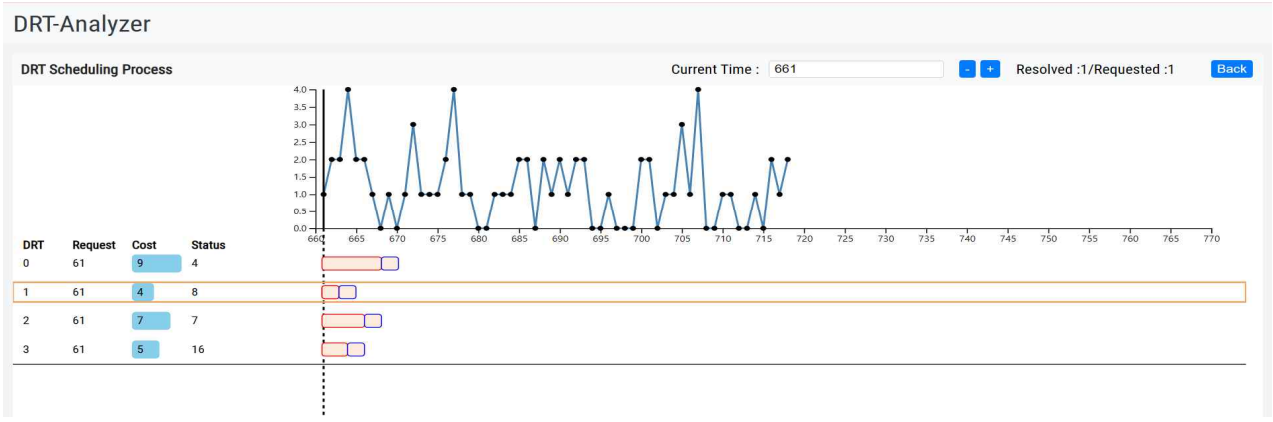

그림 6. 661분 시점. 최단 경로를 선택함.
Fig. 6. At 661 minutes. Shortest path is selected.

그림 7. 664분 시점. 더 긴 경로를 선택함.
Fig. 7. At 664 minutes. Longer path is selected.

시각화의 X축은 오전 11시를 시작으로 각 분(minute) 단위로 경과 시간을 나타내며, Y축은 1부터 80까지의 요청 ID를 나타낸다. 각 승객 요청은 다음 세 가지 주요 시각적 요소로 구성된다: (1) 요청 시간, (2) 탑승 허용 시간, (3) 하차 허용 시간. 요청 시간은 빨간색 점으로 표시되며, 탑승 허용 시간은 파란색의 수평 막대, 하차 허용 시간은 주황색의 수평 막대로 표현된다. 이와 같이 동일한 수평선 상에서 세 가지 요소가 결합되어 하나의 승객 요청을 나타내며, 이를 통해 사용자는 각 요청이 언제 이루어졌고, 탑승 및 하차가 가능한 시간이 언제인지 파악할 수 있다.

이 시각화 도구는 몇 가지 상호작용 기법을 제공한다. 사용자는 요청 시간을 기준으로 오름차순 또는 내림차순으로 정렬할 수 있어, 시간대별로 요청이 어떻게 분포되는지를 한눈에 확인할 수 있다. 또한, 사용자가 시각화 내의 특정 요청 막대에 마우스 커서를 올리면, 해당 요청의 ID, 출발 및 도착 노드 정보, 요청 시간, 탑승 및 하차 가능 시간과 같은 세부 정보를 툴팁(tooltip) 형태로 제공한다. 이렇게 세부 정보가 필요한 경우에 연구자는 각 요청의 구체적인 내용을 신속하게 확인하고, 깊이 있는 분석을 수행할 수 있다.

3-2 교통 네트워크 시각화

교통 네트워크 시각화(transport network view)는 교통 네트워크 내에서 노드들 간의 연결 구조를 시각화한다(그림 4). 이 시각화는 복잡한 교통 네트워크를 단순화된 형태로 이해할 수 있도록 단순화하여 표현함으로써, 교통 연구자들이 네트워크의 상태를 쉽게 파악하고 교통 흐름의 최적화를 위한 결정을 내릴 수 있도록 돕는다.

이 시각화에서는 수폴스 네트워크 데이터로부터 노드들의 위치 데이터(X, Y 좌표)를 받아와 각 노드를 해당 위치에 파란 원형으로 시각화한다. 이 원형 노드는 교통 네트워크 내의 특정 지점을 나타내며, 사용자는 이를 통해 노드 간의 상대적 위치를 쉽게 이해할 수 있다. 각 노드 간의 경로는 직선으로 연결되며, 이는 실제 교통 네트워크에서의 링크를 의미한다.

그림 8. 687분 시점. 선택된 DRT가 없음.
Fig. 8. At 687 minutes. No DRT was selected.

링크는 노드 간의 물리적 연결을 나타내며, DRT 차량이 이 경로를 따라 이동할 수 있는 경로를 시각적으로 보여준다.

이 시각화의 중요한 특징 중 하나는 상술한 승객 요청 분포 시각화와 연동하여 활용할 수 있다는 것이다. 사용자가 승객 요청 분포 시각화 내의 특정 요청에 해당하는 수평 그래프 부분에 마우스를 올릴 때, 해당 요청에 대응되는 출발지 노드와 도착지 노드 간의 경로를 강조하여 나타내는 기능이다. 이를 통해 사용자는 특정 요청이 교통 네트워크 내에서 어떠한 경로를 통해 이루어지는지를 단순화된 형태로 이해할 수 있다. 출발지 노드는 빨간색 테두리로 강조되며, 도착지 노드는 파란색 테두리로 강조되어 해당 경로를 명확히 구분할 수 있도록 한다. 이와 같은 시각적 피드백은 연구자들이 특정 요청이 전체 네트워크에서 차지하는 위치를 더 쉽게 파악할 수 있게 한다.

3-3 DRT 스케줄링 과정 시각화

DRT 스케줄링 과정 시각화 (DRT Scheduling Process view)는 DRT 스케줄링 알고리즘이 수행되는 전체 과정을 시각화하는 도구이다(그림 5). 승객의 요청이 들어오면 가용 DRT들 각각이 갖는 요청 큐(queue) 내에서 새로운 해당 요청이 들어오면 가용 DRT들 각각이 갖는 요청 큐(queue) 내에서 새로운 해당 요청이 들어갈 자리를 찾는 과정을 시각적으로 표현한다. 이 시각화에서는 각각의 요청 큐 후보를 위에서 아래로 나열하여, 연구자가 스케줄링 알고리즘이 어떤 후보군 내에서 어떤 결정을 내리는지를 명확히 확인할 수 있다. 이러한 시각화는 DRT 스케줄링의 복잡성을 효과적으로 이해하고, 알고리즘의 성능을 개선하는 데 필요한 통찰을 제공한다.

각 행은 하나의 요청 큐 후보를 나타내며, DRT의 ID, 새로 추가된 요청 ID, 해당 후보 요청 큐의 비용이 시각적으로 표시된다. 스케줄링 알고리즘에 의해 채택된 후보는 주황색 테두리로 강조되어 표시되며, 연구자는 이 후보가 선택된 이유를 비용 등 다양한 요소를 통해 분석할 수 있다. 특히, 주황색 테두리로 선택된 DRT는 회색으로 표현되어 이전의 이동 경로와 현재 상태를 시각적으로 추적할 수 있도록 하였다.

또한, 요청 큐 자체를 시각적 요소로 나타내기 위해 주황색 계열의 블록을 사용하여 DRT가 승객을 이동시키는 과정을 시각화하였다. 승차 및 하차 이벤트는 각각 빨간색과 파란색 테두리로 표시되어, DRT가 승객을 어디서 태우고 어디서 내려주는지를 명확히 보여준다.

이와 함께, 새로운 기능으로 status 칸을 추가하여 DRT의 현재 위치나 상태를 나타내도록 하였다. DRT가 이동 중일 경우 '->' 기호를 사용하여 어디서 어디로 이동하는지를 표시하며, 이 정보를 통해 DRT의 실시간 상태를 쉽게 파악할 수 있다.

시각화 요소 중 하나로 추가된 차트는 x축에 시간을, y축에 해당 시간에 선택된 DRT의 개수를 나타낸다. 이 차트는 660분부터 770분까지의 시간을 나타내며, DRT가 시간에 따라 어떻게 동작하는지를 한눈에 파악할 수 있도록 한다. 또한, 사용자가 드래그(drag) 기능을 통해 current time을 변경할 수 있게 하여, DRT의 이동 경로와 상태를 실시간으로 확인하고 분석할 수 있도록 하였다.

Cost의 경우, 가장 큰 cost를 기준으로 비율에 따라 막대의 길이를 조정하여 시각적으로 나타내었으며, 승차나 하차 시간이 동일한 경우 이를 명확히 표현하기 위해 일정한 width를 추가하여 표시하였다. 또한, 차트에서 호버링(hovering) 기능을 통해 time과 resolved 정보를 제공하여 연구자가 더 단순화된 형태로 데이터를 해석할 수 있도록 하였다.

추가적으로, 시각화에서 색상은 중요한 역할을 한다. 비용 부분에서 파란색으로 표시된 후보들은 정상적인 후보군이며, 빨간색으로 표시된 후보들은 제한된 후보군이다. 주황색 계열의 색상의 블록들은 DRT가 운반하는 승객 수에 따라 다르게

표시되며, 승객 수가 많을수록 더 진한 색으로 표현된다.

이러한 시각화 시스템은 DRT 시스템 연구에서 필수적인 도구로, 복잡한 데이터를 효과적으로 분석하고 알고리즘의 성능을 최적화하는 데 중요한 역할을 한다. 이와 같은 시각화 기법을 통해 연구자는 DRT 시스템의 효율성을 극대화할 수 있는 다양한 전략을 개발할 수 있을 것이다.

Ⅳ. 분석 및 논의

본 연구에서 제안하는 대화형 시각화 시스템을 활용하여 DRT 스케줄링 알고리즘의 수행 과정을 시각적으로 분석한 결과에 해당하는 다양한 분석 시나리오를 제시함으로써 시스템의 효과를 보이고자 한다.

본 분석 시나리오에서는 특정 시간대에 발생한 승객 요청을 처리하기 위한 DRT 차량의 스케줄링 과정을 중심으로 분석을 수행하였다. 각 시나리오에서는 DRT 스케줄링 알고리즘이 승객의 요청에 어떻게 대응하는지 추적하고, 그 과정에서 발생하는 다양한 변수들을 평가하였다. 특히, 비용, 경로, DRT 차량의 현재 상태 등의 요소가 어떻게 결합하여 최종적인 스케줄링 결과를 도출하는지에 대한 분석을 진행하였다.

661분 시점(그림 6)에서 DRT 스케줄링 과정에서는 요청 61에 대해 네 가지 DRT 후보가 평가되었다. 각 후보의 거리는 DRT 0이 9, DRT 1이 4, DRT 2가 7, DRT 3이 5로 계산되었으며, 시스템은 가장 짧은 거리 값을 가진 DRT 1을 선택했다. DRT 1은 출발 노드 8에서 도착 노드 9로 이동한 후, 최종적으로 노드 10에 도착하는 경로를 제공하였다. 이 경로는 4의 거리 값을 나타내었고, 다른 후보들에 비해 가장 효율적이었기 때문에 시스템에 의해 선택되었다. 이 시점에서의 분석은 DRT 시스템이 거리 최적화를 우선시하여 가장 짧은 경로를 선택했음을 보여주며, 이는 시스템이 효율적인 운행을 위해 거리 절감을 중요한 요소로 고려하고 있음을 시사한다.

664분 시점(그림 7)에서는 요청 7에 대해 여러 후보 경로가 평가되었다. 이 과정에서 DRT 1이 14의 거릿값을 가진 가장 짧은 경로를 제시했으나, 해당 경로가 최대 수용 인원을 초과하거나 수행할 수 없는 경로로 판정되어 선택되지 않았다. 대신, 경로 길이가 20으로 상대적으로 긴 DRT 2가 제한 없이 수행할 수 있는 경로를 제공하여 최종적으로 선택되었다. 이 시나리오는 DRT 시스템이 단순히 짧은 경로를 선택하기보다는 경로의 실행 가능성과 안정성을 중시하며, 실패할 가능성이 없는 경로를 우선적으로 선택하는 운영 방식을 채택하고 있음을 보여준다.

687분 시점(그림 8)에서는 요청 34에 대해 DRT 시스템이 여러 후보 경로를 평가했으나, 모든 후보 경로가 제한 조건을 충족하지 못해 최종적으로 선택된 DRT가 없었다. 이 시나리오는 시스템이 거리 값뿐만 아니라 경로의 실현 가능성을 철저히 고려하여, 수행 불가능한 경로가 포함된 경우 어떤 DRT도 선택하지 않는다는 점을 명확히 보여준다. 이로써 시스템은 불필요한 경로 선택으로 인한 오류를 방지하고, 운영의 신뢰성을 유지하는 데 중점을 두고 있음을 알 수 있다.

Ⅴ. 결론 및 향후 연구

본 연구에서는 DRT의 스케줄링 알고리즘을 분석하고 시각화하기 위한 대화형 시스템을 제안하였다. 이 시스템은 수폴스 네트워크를 기반으로 다양한 DRT 요청 데이터와 노드 간 이동 시간을 활용하여, 연구자들이 복잡한 스케줄링 과정을 단순화된 형태로 이해할 수 있도록 설계되었다. 특히, 승객 요청 분포, 교통 네트워크, DRT 스케줄링 과정 시각화와 같은 모듈을 통해 각종 데이터를 시각적으로 표현하고, 스케줄링 알고리즘의 동작 과정을 탐색할 수 있도록 하였다.

이 시스템의 주요 기여는 다소 복잡하고 방대할 수 있는 DRT 스케줄링 과정의 탐색이 용이하게 시각적으로 표현함으로써, 연구자가 스케줄링 알고리즘의 동작에 대한 이해도를 높이고, 더 나은 알고리즘 개발을 위한 통찰을 얻을 수 있도록 지원하는 데 있다.

향후 연구에서는 다음과 같은 방향으로 본 연구를 확장할 수 있다. 첫째, 사용자 인터페이스의 개선을 통해 기술적 접근성이 낮은 사용자들도 쉽게 시스템을 활용할 수 있도록 할 필요가 있다. 현재 시스템은 연구자들을 주 사용자층으로 설정하고 설계되었으나, 일반적인 교통 운영자나 정책 결정자도 쉽게 접근할 수 있도록 인터페이스를 개선하는 것이 중요하다. 둘째, 본 연구에서 제안한 시스템을 실제 운영 환경에 적용하고, 그 효과를 실증적으로 평가하는 연구가 필요하다. 시뮬레이션 기반의 연구에서 도출된 결과가 실제 교통 시스템에 얼마나 효과적으로 적용될 수 있는지를 검증하는 것은 매우 중요한 단계이다. 이를 통해 시스템의 실질적인 유용성과 한계를 파악하고, 추가적인 개선 방안을 도출할 수 있을 것이다. 셋째, 다양한 데이터 소스와 알고리즘을 시스템에 통합하여 분석 능력을 확장하는 연구가 필요하다. 예를 들어, 실시간 교통 데이터, 날씨 정보, 승객의 행태 데이터 등을 통합하여 더 정교한 스케줄링 알고리즘을 개발하고, 이를 시각화 시스템에 적용함으로써 DRT 시스템의 운영 효율성을 극대화할 수 있을 것이다. 특히, 현재 제안된 DRT스케줄링 알고리즘은 중소규모의 승객 요청에 적합하게 설계되었으나, 승객 수와 차량 수가 증가하거나 다양한 데이터 소스가 추가된다면 계산 복잡도가 급격히 증가하여 NP-hard 문제로 확장될 수 있다는 한계를 명확히 인식하고 이에 대한 개선이 필요하다. 마지막으로, 본 연구에서 제안된 시각화 시스템을 사용한 다양한 스케줄링 알고리즘의 비교 연구가 필요하다. 이를 통해 어떤 알고리즘이 특정 환경에서 가장 효과적으로 작동하는지에 대한 종합적인 이해를 도출할 수 있으며, 이는 DRT 시스템의 최적화에 중요한 시사점을 제공할 것이다.

참고문헌

[1] L. Quadrifoglio, M. M. Dessouky, and F. Ordóñez, “A Simulation Study of Demand Responsive Transit System Design,” *Transportation Research Part A: Policy and Practice*, Vol. 42, No. 4, pp. 718-737, May 2008. https://doi.org/10.1016/j.tra.2008.01.018

[2] Varo. DRT System [Internet]. Available: https://varodrt.com/.

[3] K. Tsubouchi, K. Hiekata, and H. Yamato, “Scheduling Algorithm for On-Demand Bus System,” in *Proceedings of 2009 Sixth International Conference on Information Technology: New Generations*, Las Vegas: NV, pp. 189-194, April 2009. https://doi.org/10.1109/ITNG.2009.224

[4] P. Li, L. Jiang, S. Zhang, and X. Jiang, “Demand Response Transit Scheduling Research Based on Urban and Rural Transportation Station Optimization,” *Sustainability*, Vol. 14, No. 20, 13328, October 2022. https://doi.org/10.3390/su142013328

[5] D. S. Kessler, *Computer-Aided Scheduling and Dispatch in Demand-Responsive Transit Services*, Washington, DC: Transportation Research Board, 2004.

[6] S.-H. Bak, J. H. Kim, and H.-B. You, “Implementation of Public Data Contents Using Big Data Visualization Technology - Map Visualization Technique,” *Journal of Digital Contents Society*, Vol. 18, No. 7, pp. 1427-1434, November 2017. https://doi.org/10.9728/dcs.2017.18.7.1427

[7] J. Park and J. Kim, “Solving the Abusive Comments Problem through ML-Based Visualization,” *Journal of Digital Contents Society*, Vol. 21, No. 4, pp. 771-779, April 2020. https://doi.org/10.9728/dcs.2020.21.4.771

[8] K. Kwak, J. Park, and H. Song, “Inter-Hospital Patient Transfer Network Visualization,” *Journal of Digital Contents Society*, Vol. 24, No. 2, pp. 411-419, February 2023. https://doi.org/10.9728/dcs.2023.24.2.411

[9] A. Babazadeh, H. Poorzahedy, and S. Nikoosokhan, “Application of Particle Swarm Optimization to Transportation Network Design Problem,” *Journal of King Saud University - Science*, Vol. 23, No. 3, pp. 293-300, July 2011. https://doi.org/10.1016/j.jksus.2011.03.001

[10] E. W. Dijkstra, “A Note on Two Problems in Connexion with Graphs,” *Numerische Mathematik*, Vol. 1, pp. 269-271, December 1959. https://doi.org/10.1007/BF01386390

[11] React. The Library for Web and Native User Interfaces [Internet]. Available: https://react.dev/.

[12] M. Bostock, V. Ogievetsky, and J. Heer, “D³ Data-Driven Documents,” *IEEE Transactions on Visualization and Computer Graphics*, Vol. 17, No. 12, pp. 2301-2309, December 2011. https://doi.org/10.1109/TVCG.2011.185

김동규(Dongkyu Kim)

2022년 : 광운대학교 입학

2022년～현 재: 광운대학교 컴퓨터정보공학부 학사과정

※관심분야 : 휴먼-컴퓨터 인터액션(Human-Computer Interaction), 데이터 시각화, 시각적 분석 등

신동화(DongHwa Shin)

2013년 : 한국과학기술원(KAIST) 전산학과 (공학사)

2021년 : 서울대학교 전기·컴퓨터공학부 대학원 (공학박사- 휴먼-컴퓨터 인터액션)

2021년～2022년: 서울대학교 컴퓨터연구소 연수연구원

2022년～2023년: 서울대학교 BK21 FOUR 지능형컴퓨팅사업단 연수 연구원

2023년～현 재: 광운대학교 컴퓨터정보공학부 조교수

※관심분야 : 휴먼-컴퓨터 인터액션(Human-Computer Interaction), 데이터 시각화, 시각적 분석 등