

## 수요대응형 교통수단 시뮬레이션 방안: Real-Time Shared-Taxi 적용예시

### A Simulation Model for Evaluating Demand Responsive Transit: Real-Time Shared-Taxi Application

정재영 Jung, Jae Young | University of California, Irvine Ph.D. Candidate (E-mail : jyoungjung@gmail.com)

#### ABSTRACT

Demand Responsive Transit (DRT) services are becoming necessary as part of not only alternative transportation means for elderly and mobility impaired passengers, but also sustainable and flexible transportation options in urban area due to the development of communication technologies and Location Based Services (LBS). It is difficult to investigate the system performance regarding vehicle operational schemes and vehicle routing algorithms due to the lack of commercial software to support door-to-door vehicle simulation for larger area. This study proposes a simulation framework to evaluate innovative and flexible transit systems focusing on various vehicle routing algorithms, which describes data-type requirements for simulating door-to-door service on demand. A simulation framework is applied to compare two vehicle dispatch algorithms, Nearest Vehicle Dispatch (NVD) and Insertion Heuristic (IH) for real-time shared-taxi service in Seoul. System productivity and efficiency of the shared-taxi service are investigated, comparing to the conventional taxi system.

#### KEYWORDS

*demand responsive transit, DRT, simulation modeling, real-time shared-taxi, vehicle routing problem*

#### 요지

수요대응형 교통수단(Demand Responsive Transit)은 변화하는 이동수요에 대응하는 탄력적인 교통수단으로 단순히 노약자와 장애인을 위한 복지교통 서비스의 영역이 아니라, 무선통신과 위치정보서비스(Location Based Service: LBS)의 발달로 인하여 도심형 수단으로 보다 효율적인 교통수단으로 자리매김하고 있다. 그러나 문전서비스(Door-to-Door)를 제공하는 수요대응형 교통수단 시뮬레이션에 적합한 상용툴의 부재로 인하여 알고리즘이나 차량 운행 요소를 면밀하게 분석하기 힘든 어려움이 있었다. 본 연구는 수요대응형 교통수단에 연관된 다양한 차량 운영계획과 알고리즘을 구현, 평가할 수 있는 시뮬레이션 환경을 제안한다. 문전서비스(Door-to-Door) 기반의 차량 운행 모형을 적용하기 위하여 확보되어야 하는 시뮬레이션 입력 데이터를 정의하고 있으며, 수요대응형 교통수단의 대표적인 범주에 속하는 실시간 합승 택시(Shared-Taxi) 서비스를 서울시 교통망과 택시 수요를 이용하여 적용하였다. 합승 택시 운행 계획을 위하여 Nearest Vehicle Dispatch(NVD)와 Insertion Heuristic(IH), 두 종류의 알고리즘을 제안하였으며, 제안된 시뮬레이션을 통하여 성능을 비교하였다. 또한, 합승(Ride-sharing)을 허용하지 않는 일반적인 택시와의 비교를 통하여 시스템 효율 향상과 서비스 품질 변화를 분석하였다.

#### 핵심용어

수요대응형 교통수단, 시뮬레이션 모형, 택시 합승, 차량 운행 경로 문제

## 1. 서론

최근, 탄력적인 노선과 배차 간격 혹은 동적수요에 따라 운영되는 수요대응형 교통수단(Demand Responsive Transit)은 많은 주목을 받고 있다. 일반적인 고정된 노선과 배차간격에 의해 운행되는 전통적인 대중교통체계의 특징으로는 수요가 높은 노선에서는 높은 배차간격으로 인하여 서비스의 품질과 수익성이 높은 반면, 수요가 낮은 노선에서는 비효율적인 노선 형태와 낮은 배차간격으로 인하여 수익성은 물론 승객 서비스의 품질까지도 떨어지는 특징이 있다. 게다가 낮은 수요가 발생하는 시간대에 매년 발생량이 큰 차량을 운행하는 것 또한 잠재적인 환경 문제로 지적되어 왔다. 이와는 달리 수요대응형 교통수단은 노선과 배차시간에 제약받지 않고 동적으로 발생하는 수요에 따라 고정된 노선 없이 운영되는 대중교통 서비스를 말하는 것으로, 준대중교통(paratransit)의 형태로도 알려져 있다(한국교통연구원, 2008).

이러한 서비스의 큰 특징으로는 정해진 운행계획표 없이 이용 승객들의 통행 기종점을 기반으로 차량 노선과 일정을 최적화하여 운행하는 것이라 볼 수 있다. 과거의 수요대응형 교통수단은 일반적으로 노약자와 장애인의 이동편의를 위하여 적용되었다. 그러나 현재에는 무선통신 기술과 위치정보 서비스(LBS: Location Based Services) 기술발달로 인하여 동적인 수요에 따라 실시간으로 차량 운행일정을 반영할 수 있는 서비스가 시작되고 있다.

수요대응형 교통서비스는 운영형태에 따라 크게 두 가지로 나누어진다. 첫째는 고정 노선에 기반한 탄력적 기법으로, 운행 차량은 기본적인 노선과 정류장이 있으나 수요에 따라 운행 노선과 정류장이 탄력적으로 변경되는 수단이며, 둘째로 문전 서비스(Door-to-Door)의 운영형태로 정해진 노선과 정류장이 없이 오직 이용자 수요에 따라 차량의 동적으로 운영일정을 변경하는 방법이다. 전자의 경우, 승차 용량이 큰 버스가 사용되며, 후자의 경우에는 미니밴이나 택시와 같은 중,소형 차량이 이용된다. 본 연구에서는 후자의 운영형태에 초점을 맞춘다.

전통적인 대중교통 수단의 운행계획 및 평가에서는 잠재수요에 따른 최적화된 노선설계와 정류장 위치 선정이 중요한 내용으로 알려져 있는 반면, 수요대응형 교통수단의 운행 모델은 실시간으로 발생하는 이용자 수요에 대응하는 차량 운행계획을 수립하는 알고리즘이 매우 중요하다. 수요대응형 교통체계의 최적화에 이용

되는 기법으로는 수리적 모델은 물론 휴리스틱 모델 등 여러 가지 기법이 사용되고 있다. 일반적으로 노선과 배차간격에 의한 교통수단의 평가는 노선과 정류장의 위치 제한적인 특징으로 인하여 정적인 확률모형에 의해 평가될 수 있었다. 하지만, 수요대응형 교통수단의 경우 이용자의 기종점이 교통망 전체에 랜덤하게 분포할 수 있고, 정해진 노선 없이 개별 차량의 운행 일정이 이용자들의 기종점 통행 특성을 반영해야 하는 바, 시뮬레이션을 통한 평가는 필수적이다.

## 2. 선행 연구 고찰

수요대응형 교통수단의 시뮬레이션 평가 방법은 크게 세 가지로 구분되어져왔다. 첫째로, 실제 교통망을 사용하지 않고, 평가지역을 격자형 구역(Cell)이나 단순거리를 이용하여 평가하는 방법이다. 이것은 과거 알고리즘 평가 기법에 주로 사용되어오던 가장 간단한 방법이다. 별도의 교통망이 필요 없기 때문에 실제 관심지역이 아닌, 알고리즘 자체에 초점을 둔 연구에서 많이 사용된다. 실제 수요 데이터가 존재하더라도 실제 교통망의 특성이 반영되지 않는다는 단점이 있다. Quadrioglio 외(2008)의 연구에서는 Los Angeles County의 수요대응형 교통수단의 시뮬레이션 모델을 위해 실제 교통망이 아닌 격자형 가로망과 Cell 타입의 수요 발생 지점을 가정으로 한 시뮬레이션 모델을 제안하였다. 차량의 운행 거리는 직선거리에 근거하기 때문에 지역적인 특성이나 교통상황을 반영하기 어렵다.

둘째로, 미시적 교통 시뮬레이션(Microscopic Traffic Simulation)을 이용한 분석모델이 있다. 이 방법은 동적인 교통망의 혼잡 상황을 정밀하게 표현할 수 있기 때문에 네트워크의 특성을 고려한 차량의 이동 시간과 출발 상황이라는 측면에서 볼 때, 다른 모형에 비해 매우 정밀하다. 하지만, 단점으로는 도로 신호체계와 기하구조를 포함한 도로 교통망의 세부내용까지 입력해야 하는 것은 물론, 수요예측 단계에서 발생한 기종점 데이터를 이용하여 시뮬레이션의 동적 통행 수요 데이터를 추정하고 관련 파라미터를 조정하는 번거로움이 따른다. 대부분의 수요대응형 교통수단의 분석은 광역 네트워크를 기반으로 이루어지는 것을 감안할 때, 미시적 교통 시뮬레이션의 활용은 상당히 많은 부가적인 노력과 시간이 필요한 것이 사실이다. 또한 광역 교통망을 다루는 경우, 시뮬레이션 수행 시간이 지나치게 길어짐으로 인한 단점도 존재한다. 이러한 방식은 Venglar 외

(1995), Abdulhai 외(2002)의 연구들과 같이 경전철(Light Rail Transit), BRT(Bus Rapid Transit), Street Car와 같은 문전서비스가 아닌 정해진 경로와 정거장을 기반으로 한 수요대응형 교통수단의 시뮬레이션으로 주로 이용되었다.

한편, Cortés 외(2005)의 연구에서는 수요대응형 교통수단의 문전서비스를 위해 미시적 교통 시뮬레이션 기법을 제안하였다. 그러나 문전서비스를 위한 수요 발생 지점의 제약 및 스케줄링 알고리즘의 개발 제약으로 인하여 실시간 차량 운행계획 알고리즘은 별도의 단순화된 교통망을 이용하여 구성되었다.

마지막으로, 시뮬레이션의 대상지역의 GIS 데이터 혹은 교통망을 이용하는 방법으로, 연구 목적과 특성에 맞게 직접 특화된 분석 방법이다. 미시 모델과 달리 이러한 방법은 즉시 사용 가능한 상용틀이 존재하지 않기 때문에 차량 운행계획 요소는 물론 시뮬레이션을 위한 입력 자료 설계, 차량 이동 모듈 개발과 같은 많은 요소와 노력이 필요하다.

본 연구에서는 마지막으로 제안된 방식에 근거한 최적화되고 범용적인 시뮬레이션 구성 방안을 제안한다. 특히, 기존 연구들에서 알려진 문제점들을 극복하기 위하여, 본 연구에서 요구되는 시뮬레이션을 위한 기본 조건들은 다음과 같다. 1) 차량은 실시간으로 출발이나 수요에 대응하여 동적으로 경로를 변경할 수 있어야 한다. 2) 시뮬레이션을 구성하는 교통망은 단순히 대중교통수단을 위한 노드와 링크 체계가 아닌 일반 차량의 경로와 대응되는 넓은 영역과 세부적인 도로망이며, 추가적인 입력작업 없이 시뮬레이션으로 입력 가능하여야 한다. 3) 시뮬레이션 구조는 다양한 종류의 운행 차량과 알고리즘을 수용할 수 있도록 유동성(Flexibility)을 제공하여야 한다. 4) 차량과 승객들이 정보가 시각적으로 표시되거나 결과를 표출하는 적절한 인터페이스가 필요하다. 5) 운행 차량의 대수와 동적 수요 규모의 확장성(Scalability)이 보장되어야 한다.

### 3. 시뮬레이터 구조

#### 3.1. 시뮬레이터 구성

연구에서 제안된 시뮬레이션 방안의 가장 큰 특징은 기존의 교통망으로 이용되는 데이터와 수요 데이터를 적절한 형태로 변환하여 시뮬레이터에 입력되기 때문에 시뮬레이션을 위한 별도의 네트워크 구축 작업이 필요하지 않다는 것이다. 예를 들면, 상용 GIS(Geographic

Information System) 데이터, 차량용 네비게이션 지도, 교통 계획 분석 모델에서 사용되는 교통망과 같은 다양한 형태의 네트워크를 이용할 수 있는 장점이 있다. 즉, 다양한 형태의 데이터 포맷을 처리, 변환할 수 있도록 중간 형태의 포맷(Intermediate Format)이 고안되었다.

그림 1은 다양한 형태의 원본 교통망과 수요 데이터가 중간형태의 데이터베이스로 저장 가능하며, 시뮬레이션에 필요한 형태로 재가공되어 입력되는 과정을 보여준다. 문전서비스의 경우 승객 기종점은 대상지역의 임의지점에서 발생하기 때문에 Zone 형상 정보 또한 필요하다. 시뮬레이션 프로그램은 차량 운행을 결정짓기 위한 알고리즘이 포함된 모듈, 경로 탐색 모듈, 지도/차량 표시 모듈, 통계 모듈을 기본적으로 포함하고 있다. 부가적으로 시뮬레이션 중 차량의 실시간 물리적인 이동을 가정하기 위하여 실시간 교통정보가 링크 속도의 형태로 입력될 수 있다.

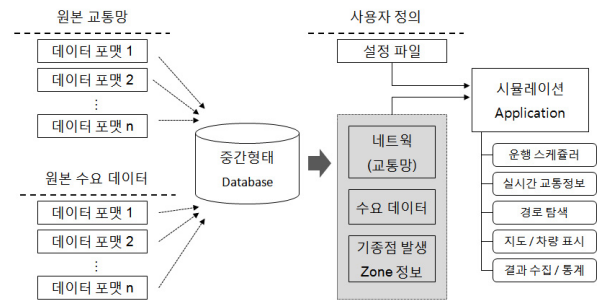


그림 1. 제안된 시뮬레이션 구조

#### 3.2. 교통망 데이터

중간 형태의 데이터에 모든 교통망 정보가 입력된 후, 별도로 구축된 데이터 변환기는 해당 데이터를 시뮬레이션을 위한 네트워크로 변환한다. 중간 형태의 데이터 구조와 달리 교통망 입력 네트워크의 형태로는 빠른 경로 탐색을 위하여 이진데이터(Binary data)형태의 지도 포맷이 고안되었다.

그림 2는 시뮬레이터의 네트워크로 사용된 원본 교통망의 예시를 보여주고 있다. 원본 교통망은 미국 캘리포니아 SCAG(Southern California Association of Governments)의 교통 계획 모델에서 사용되는 교통망으로 남가주지역의 모든 고속도로, 간선도로, 집분산도로를 포함하고 있으며, 회전정보와 일방통행정보 등 기본적인 교통망을 구성하기 위한 노드와 링크 정보를 포함하고 있다.

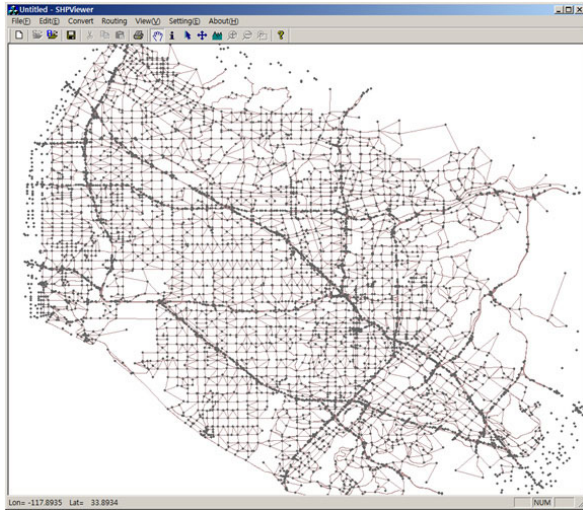


그림 2. 원본 네트워크 (Orange County, California, U.S.A.)

일반적인 교통 계획 분석에서 사용되는 교통망을 사용하는 경우, 가장 주의해야 할 내용으로는 존을 구성하는 Centroid와 Centroid Connector가 삭제되어야 하는 것이다. 이들은 교통 계획 모델에서 교통량을 발생시키기 위한 가상의 노드와 링크이므로 실제 차량이 통행할 수 없는 교통망 요소이기 때문이다. 따라서 입력 데이터 변환과정에서 해당 요소를 삭제하는 알고리즘이 추가된다. 그림 3은 시뮬레이션에 입력되는 Centroid와 해당 Connector가 삭제된 최종 교통망을 보여준다.

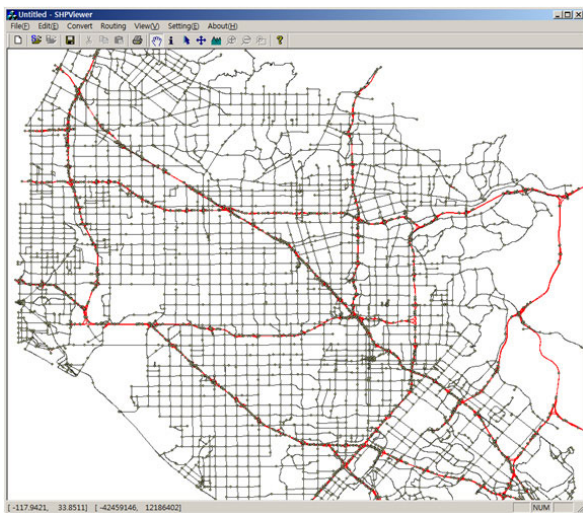


그림 3. 변환된 시뮬레이터 네트워크

### 3.3. 동적 수요 데이터

앞서 언급된 바와 같이, 문전 서비스를 위한 수요대응형 교통수단의 기중점은 버스/택시 정류장이 아닌 서비

스 지역 내의 2차원 지점 어느 곳에서도 가능하다. 제안된 시뮬레이션의 기중점 위치는 점(Point)으로 표현되며, (1) 실제 GPS를 이용한 경위도, (2) 주소를 이용한 위치, (3) 교통 계획 모델의 통행 분석존(Traffic Analysis Zone: TAZ)에 의한 랜덤 지점 발생을 지원하고 있다.

그림 4는 TAZ를 이용한 승객의 기점 발생 예시를 보여준다. 대부분의 교통 계획 모델의 TAZ는 폴리곤(Polygon)의 형태로써 통행 분포를 위한 정보를 포함하고 있다. 승객의 기중점은 폴리곤 내의 어떠한 지점에서 발생할 수 있으며, 해당 지점에서 가장 가까운 교통망의 링크(Link Id), 주행 방향(Direction), 진행거리(Distance)로 결정된다. 이 세 가지 요소는 이용자의 기중점을 판단하는데 가장 중요한 역할을 한다. 아울러, 확인된 승객의 기중점은 차량의 운행 계획 알고리즘으로 입력된다.

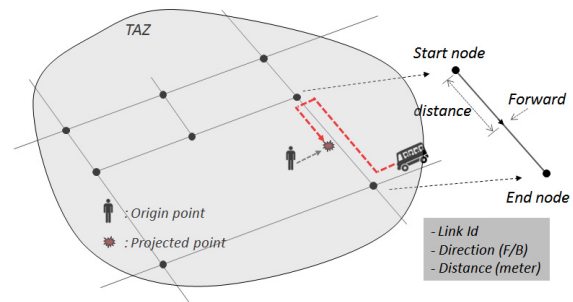


그림 4. TAZ내의 승객 기점 발생

## 4. 적용 교통수단 및 시나리오

### 4.1. Shared-Taxi (합승택시) 모형

택시 운송 수단은 도심 교통의 대표적인 수요대응형 준대중교통 수단으로 여겨진다. 택시는 대중교통수단과 고급 교통수단이라는 양면성을 가지고 있기 때문에, 아시아와 유럽 등 많은 대도시에서 많은 응용형태로 발전되고 있다. 대표적인 수단 중에 하나인 Shared-taxi는 기존 택시 서비스와 대중버스의 취약점을 극복하고자 고안된 방안으로, 다른 승객들과 합승(ride-sharing)을 도모하여 시스템의 생산성을 높이고 더불어 승객의 요금을 낮추는데 그 의미가 있다. 즉, 이용자들은 실시간으로 택시 서비스를 예약 혹은 요청할 수 있으며, 경우에 따라 같은 목적지로 가는 다른 승객과 합승을 할 수 있다.

본 연구는 합승으로 인해 지나치게 증가하는 택시 대

기시간과 우회경로를 방지하기 위하여 두 개의 제약조건을 고려한다. 첫째는 최대 대기시간을 설정하는 것으로, 요청을 받은 택시는 15분 이내로 승객의 출발지에 도착한다. 이것은 새로운 요청이 지속적으로 발생하는 동적 시스템에서 무한 연기 문제(indefinite deferment of unassigned passengers)를 방지하는 효과가 있다 (Psaraftis, 1988). 둘째는 합승에 의한 추가적인 통행시간 증가를 최소화하기 위하여 최대 우회(Maximum Detour Factor) 허용치를 설정한다. 즉, 모든 승객은 자신의 기종점 최단경로의 두 배 이상을 넘지 않는 범위 내에서 승합을 허용한다. 따라서 택시 운행 계획 알고리즘은 실시간으로 이용자의 서비스 요청을 받아들이고 현재 운행중인 택시의 스케줄에 기반하여 새로운 운행 계획을 동적으로 수립하게 된다. 모든 차량은 실시간으로 업데이트되는 스케줄에 따라 운행한다. 만약 새로운 서비스 요청을 만족하는 차량이 없다면 해당 이용자의 서비스 요청은 거부된다.

실시간 shared-taxi 문제는 차량 운행 경로 문제 (Vehicle Routing Problem) 의 일종으로 전형적인 여러 지점에서 다른 여러 지점으로 가는 형태(Many-to-Many)로 정의될 수 있으며, 본 연구의 시뮬레이션 적용을 위하여 두 가지 알고리즘을 제안한다.

## 4.2. 적용 알고리즘

### 4.2.1. NVD (Nearest Vehicle Dispatch)

NVD(Nearest Vehicle Dispatch)는 가장 보편적으로 사용되는 알고리즘으로 직관적이며 원시적인 알고리즘이다. 택시 운영 시스템에 새로운 서비스가 요청되면, 요청된 승객의 위치에서 가장 가까운 차량을 찾아서 해당 차량의 운행계획을 갱신한다. 만약 가장 가까운 차량이 주어진 제약조건을 만족하지 않는다면 그 다음으로 가까운 차량을 찾는 방식이다. 이 방식의 장점은 승객의 서비스 대기시간을 최소화할 수 있으며, 알고리즘이 매우 단순하여 빠른 수행 속도( $O(n^2)$ )를 보여준다. 하지만 승객의 기종점을 동시에 고려하지 않기 때문에 기존 승객의 목적지와와의 거리가 먼 경우, 짧은 경로 우회가 발생된다.

만약, 조건을 만족시키는 차량이 선택되고 새로운 승객이 차량 운행에 입력되는 경우 해당 차량은 실시간으로 기존의 운행 일정과 신규 승하차 일정을 최적화 한다. 최대 승차인원이 4명인 경우라고 가정할 때, 대부분의 개별 차량은 5~7개 내외의 승하차 운행일정을 포함하고 있기 때문에 이 문제는 제약조건이 포함된

PDP(Pickup and Delivery Problem) 형태의 문제로서 최적해가 도출될 수 있다.

### 4.2.2. IH(Insertion Heuristic)

삽입 휴리스틱(IH: Insertion Heuristic)은 새로운 입력값으로 인하여 증가하는 시스템 비용, 즉 한계비용 (Marginal Cost)을 최소화하는 기법을 말하며 최소 비용 삽입(cheapest insertion)이라고 알려져 있다. 목적 함수가 모든 승객의 대기시간과 통행시간에 초점을 두는 경우, 새로운 승객은 물론 기존의 승객들에게 발생하는 추가적인 대기시간과 통행시간을 최소화할 수 있는 차량 운행 일정을 실시간으로 수립하는 것이다. 본 연구에서 제안된 알고리즘은 다음과 같다.

- 1) 새로운 승객의 서비스 요청  $z_i$ 이 시스템에 접수되고 이용자의 기종점 위치가 확인된다.
- 2) 시스템은 현재 운행중인 차량들 중에서  $z_i$ 의 승하차 스케줄( $e^{i,p}, e^{i,d}$ )의 제약조건을 만족하는 차량  $j(j \in J)$ 를 찾은 후, 해당 차량의 운행 일정( $E_j$ )에 신규 스케줄을 추가함으로써 발생하는 한계비용,  $MC_j(e^{i,p}, e^{i,d})$ 을 계산한다. 각 차량의 한계비용을 계산하는 과정에서 최적의 승하차 삽입 위치( $m, n$ ) 또한 계산된다.
- 3) 더 이상의 제약조건을 만족하는 차량이 없는 경우, 최소의 한계비용을 가진 차량을 선택한다. 만약 제약조건을 만족하는 차량이 없는 경우, 신규 서비스 요청은 실현 가능 해(feasible solution)가 아니므로 거부된다.
- 4) 선택된 차량의 운행 일정(경로)에 새로운 스케줄( $e^{i,p}, e^{i,d}$ )이 추가된다.

$$C_j(E_j) = \sum_{k \in K} [WT(e_k^p) + TT(e_k^d)] \quad (1)$$

$$C_j(E_j, e^j) = \text{Min} \sum_{k \in K+2} [WT(e_k^p) + WT(e_m^{i,p}) + TT(e_k^d) + TT(e_n^{i,d})] \quad (2)$$

$$MC_j(E_j, e^j) = C_j(E_j, e^j) - C_j(E_j) \quad (3)$$

여기서,  $C_j(E_j)$  = 차량  $j$ 의 비용(모든 승하차 승객의 대기 및 통행 시간비용)

$WT(e_k^p)$  = 승차 일정  $e_k^p$ 의 승객 대기시간

$WT(e_m^{i,p})$  =  $m$ -th 삽입위치에 해당하는  $z_i$

의 승차 일정 대기시간( $0 < m$ )

$$TT(e_k^d) = \text{하차 일정 } e_k^d \text{의 승객 통행시간}$$

$$TT(e_n^{i,d}) = n\text{-th 삽입위치에 해당하는 } z_i \text{의 하차 일정 통행시간}(m < n)$$

식(1)의  $C_j$ 는 차량  $j$ 의 현재 일정을 바탕으로 예정된 모든 승객의 대기시간과 통행시간을 포함하고 있으며 식(2)는 새로운 일정을 포함한 예정된 모든 승객의 대기시간과 통행시간을 나타낸다. 식(3)의  $MC_j$ 는 현재 차량의 스케줄( $E_j$ ) 비용과 추가된 새로운 일정과의 차이값으로 계산된다. 새로운 일정이 추가될 때 한계비용은 항상 양의 값을 갖는다.

본 연구에서 제안된 IH 알고리즘은 선착순 선점 방식(First-Come First-Served: FCFS)에 기반하고 있으며, 새로운 승객의 서비스 요청은 다른 신규 요청과 개별적이고 독립적으로 처리된다. 이러한 방식은 실시간 신규 요청들을 주기적으로 수집하여 최적화 시키는 방법에 비해 즉각적인 시스템 반응을 기대할 수 있으며, 계산 부하가 적은 장점이 있다. 하지만 시스템 전체 운영 측면에서 볼 때, 최적해를 보장하지 않는 단점이 있다.

### 4.3. 시뮬레이션 시나리오

본 연구에서 제안된 시뮬레이션 방안과 알고리즘을 적용하기 위해서 한국교통연구원에서 구축한 NGIS 기반의 서울시 교통망을 이용하였다. 사용된 원본 교통망은 EMME/2 교통 계획 모델의 형태이며, 제안된 데이터 변환 절차에 따라 중간 포맷으로 변환된 후, 시뮬레이터의 입력 교통망으로 변환되었다. Centroid와 Centroid connector가 제거된 후, 총 8,382개의 링크와 6,321개의 노드가 시뮬레이션 네트워크로 입력되었다. 시공간적인 택시 수요를 발생시키기 위하여 동일 모델에 존재하는 추정된 택시 통행 분포를 이용하였다. 서울시내에 총 560개의 통행 발생 존(Zone)을 구성하였으며, 공간적인 범위를 위해 각 존의 반경은 500m~1km로 제한하고 해당 범위 내에서 승객 기종점의 위치를 랜덤하게 발생하였다.

승객의 실시간 서비스 요청을 위해 택시 수요 데이터에 근거한 음지수분포가 가정되었다. 그림 5는 시뮬레이션을 위해 발생된 택시 수요의 통행거리 분포를 보여주고 있다. 대부분의 발생된 택시 수요 거리분포는 1km~10km로 분포되어 있으며 평균 문전 통행거리는 6.3km를 보여준다.

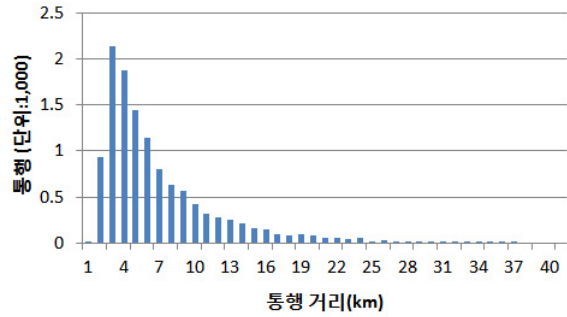


그림 5. 택시 수요의 통행거리 분포

2011년 기준 서울시 법인택시의 하루 운행대수의 3.5%에 해당하는 총 600대의 택시가 시뮬레이션에 적용되었으며, 합승 차량 최대 탑승 인원은 4명으로 가정되었다. 시뮬레이션 중 택시의 운행 속도는 도로 종별에 따라 법정 최고 속도의 60~90% 수준으로 가정되었으며 승객의 승하차시간은 표준 정규분포(평균 1분)에 근거하여 랜덤 설정되었다.

시뮬레이션은 처음 30분의 Warm-up 시간을 포함하여 4시간으로 이루어지고 다음과 같이 총 4가지 택시 서비스 수요 수준을 고려하였다, (1) 9,000 (2) 12,000 (3) 15,000 (4) 18,000 (승객 서비스 요청 건수/4시간). Shared-ride 특성을 파악하기 위하여 모든 승객 그룹의 구성인원들은 1명이고 알고리즘에 적용된 제약조건 내에서 합승 허용 의사가 있는 것으로 가정하였다. 또한 시스템의 성능 비교를 위하여 기존의 합승이 허용되지 않는 택시의 시뮬레이션 또한 병행되었다.

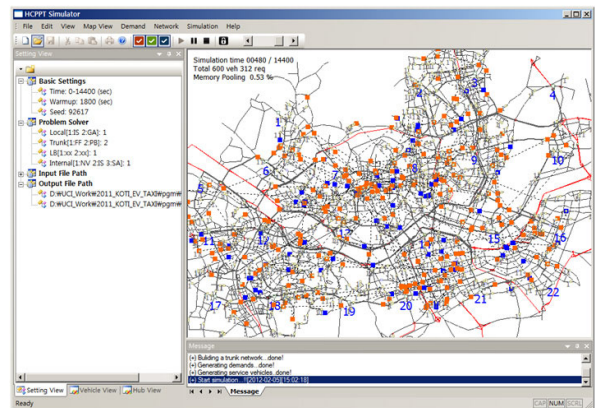


그림 6. 서울시 Shared-Taxi 시뮬레이션

## 5. 시나리오 결과 및 분석

시나리오 결과 분석을 위하여 Black(1995)의 연구에서 제안된 수요대응형 교통수단을 위한 3가지의 성능지표를 고려하였다. (1) 수송된 승객 수(명), (2) 승객의

평균 대기시간(분), (3) 승객의 평균 통행시간지표(Ride Time Index: 평균 통행시간/평균 문전 통행시간), (4) 서비스 품질지표(Level of Service: 평균 대기+통행시간 / 평균 문전 통행시간). 참고로, 승객의 평균 통행시간 지표와 서비스 품질 지표는 알고리즘의 제약조건 중 최대 우회 허용치와 동일 단위로 사용한다. 평균 문전 통행시간(average door-to-door travel time)은 승객이 승용차를 이용하여 통행하였을 때를 가정한 시간이다. 예를 들어, 승용차 통행시간이 20분으로 예상되는 승객의 기종점 통행이 Shared-Taxi로 이용되었을 때, 택시 대기시간과 통행시간의 총합계가 40분이라면 LOS(Level of Service) 지수는 2.0이다. 이는 상대적으로 더 낮은 값이 더 좋은 서비스 품질을 의미한다.

### 5.1. 알고리즘 성능 비교

그림 7은 Shared-Taxi 시스템에 적용된 알고리즘의 성능을 보여준다. 첫째로 IS 알고리즘은 NVD 알고리즘에 비해서 전체적으로 5~10% 증가된 승객 처리수(수송된 승객수: 단위 천명)를 보여주고 있다. 참고로, 승객 처리수(혹은 요청완료 건수)는 시뮬레이션 시간 내에 목적지 하차가 완료된 승객수를 의미한다(Warm-up 시간제외). 반면, 평균택시 대기시간은 이전에 언급된 것처럼 NVD 알고리즘에서 더 나은 성능을 보여준다.

수요가 낮은 경우(9k), 평균 대기시간은 6.5분(NVD)과 8.7분(IH)의 큰 차이를 보여주며, 수요가 점차 증가함에 따라 그 차이가 줄어든다. 평균 통행시간지표(Ride Time Index)의 경우, IH 알고리즘은 전체 승객의 통행시간과 대기시간을 동시에 고려하기 때문에 NVD 보다 더 나은 성능을 보여준다. 참고로, 통행 시간 지표 2.0은 본 알고리즘에서 제약조건으로 설정한 최대 우회값이다. 하지만 시뮬레이션의 특성상 승객의 승하차 시간이 랜덤하게 설정되었기 때문에 최대 우회값을 넘을 수 있다.

그림 7의 LOS(Level of Service) 지표는 평균 대기시간과 통행시간지표를 결합한 형태의 성능지표로 전체적으로 볼 때, IS 알고리즘이 승객의 전체 통행시간 측면에서 더 우수한 것으로 나타났다. 결론으로, 제안된 Shared-Taxi 알고리즘 중 IH는 NVD에 전체적으로 더 나은 이용자 대기 및 통행시간은 물론, 더 높은 시스템 성능(승객 수송량)을 보여준다. 하지만 승객 수요가 증가하면서, 그 차이는 점차 줄어든다. 이것은 실시간 수요 대응 시스템의 서비스 품질 대비 제공 가능한 서비스 용량을 파악하기 위한 주요한 지표가 될 수 있다.

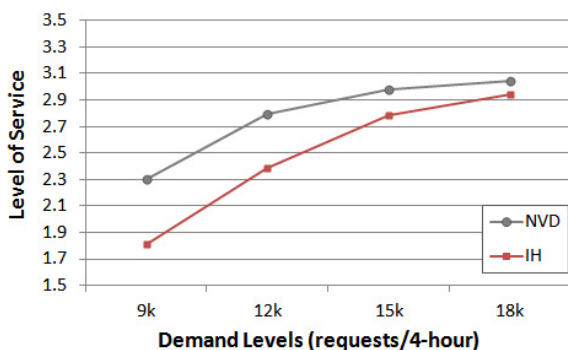
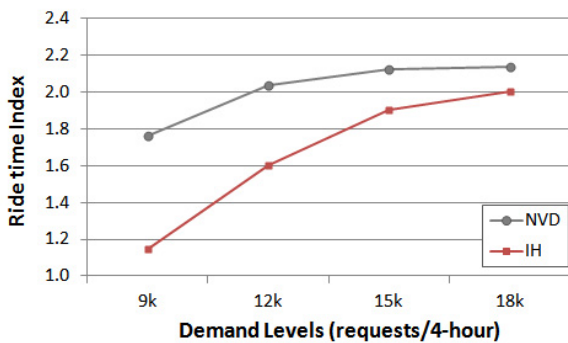
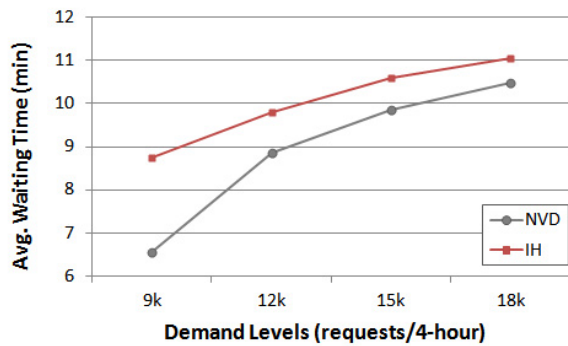
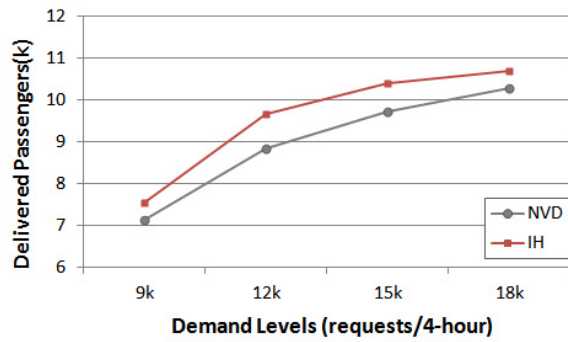


그림 7. Shared-Taxi 알고리즘 비교

### 5.2. Ride-Sharing(합승) 기대 효과

합승을 이용한 택시 시스템의 생산성과 이용자의 편의효과를 분석하기 위하여, Shared-Taxi 서비스에 해당하는 다수의 승객을 허용하는(Multiple Customer Policy: MCP) 운영기법과 전통적인 택시 서비스로 여겨지는 단일 승객만을 허용하는(Single Customer

Policy: SCP) 운영기법을 비교하였다. 두 시뮬레이션 모두 시스템 성능을 최대화할 수 있는 IH 알고리즘을 적용하였다(그림 8 참조).

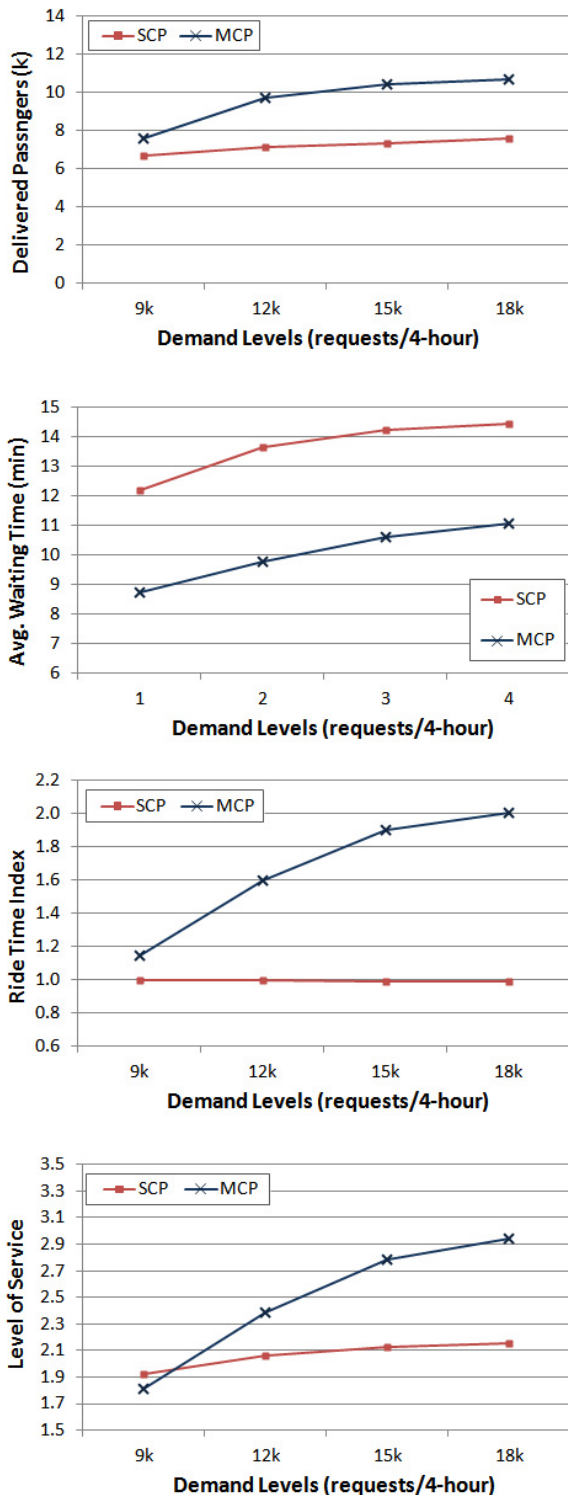


그림 8. SCP와 MCP 비교

수송된 승객의 처리 수(단위:천명)에서 볼 수 있듯이 합승을 허용하는 경우, 전통적인 택시 운영 기법에 비해

최대 40%의 승객을 추가로 처리할 수 있다. 승객의 평균 대기시간은 합승을 허용하는 경우에 더 낮은 것으로 분석되었다. 이것은 단일 승객만 허용하는 경우(SCP), 최소한의 한계 비용을 만족하는 새로운 서비스 요청이 들어오더라도 차량은 즉시 승객을 태울 수 없으며 이미 탑승했거나 예정된 승객의 서비스를 완전히 끝내고 나서야 다음 승객을 처리할 수 있기 때문이다. 결국, 합승은 실시간 수요에 적절하고 기민하게 대응할 수 있는 운영요소로 부합하는 것을 보여준다. SCP의 이용자 대기 시간이 수요가 늘어남에 따라 점차 증가하는 현상은 차량의 운행 스케줄이 없는 대기(Idling) 상태뿐만 아니라 운행 중에도 다음 승차 승객에 대한 새로운 운행 스케줄을 미리 계획할 수 있기 때문이다. 만약 차량의 대기 상태인 경우에만 새로운 운행 스케줄을 허용한다면, 대기 시간을 줄일 수 있으나 승객 처리 용량이 현저하게 줄어드는 것으로 나타났다.

합승이 허용되지 않은 경우, 수요가 증가함에도 불구하고 요청완료 건수는 크게 증가하지 않으나, 합승을 허용하는 경우(MCP)에는 수요가 증가됨에 따라 요청완료 건수도 점차적으로 증가하는 것을 보여준다. 이것은 합승 자체가 낮은 수요는 물론, 높은 수요에서도 시스템의 처리 잠재 능력을 크게 증가시킨다는 것을 의미한다.

SCP의 통행시간지표(Ride Time Index)는 경로 우회가 없는 1.0으로 일정하게 나타났다. 반면에, MCP는 수요가 낮은 경우 1.15로 낮은 수준을 보여주었으나 수요가 높은 경우 제약조건으로 설정된 최대값 2.0을 보여주었다. 택시의 운행거리 측면에서, SCP는 평균

표 1. SCP와 MCP 비교

9,000 requests/4-hour	SCP	MCP
요청완료 건수(단위:천명)	6,709	7,551
요청거부 건수(단위:천명)	1,007	6
평균주행거리(km/veh)	98.2	92.8
12,000 requests/4-hour	SCP	MCP
요청완료 건수(단위:천명)	7,100	9,679
요청거부 건수(단위:천명)	3,368	100
평균주행거리(km/veh)	98.6	88.1
15,000 requests/4-hour	SCP	MCP
요청완료 건수(단위:천명)	7,279	10,392
요청거부 건수(단위:천명)	5,910	1,837
평균주행거리(km/veh)	98.0	86.0
18,000 requests/4-hour	SCP	MCP
요청완료 건수(단위:천명)	7,553	10,689
요청거부 건수(단위:천명)	8,335	4,377
평균주행거리(km/veh)	97.5	85.2



98km 주행거리를 보여주었고, MCP는 평균 88km 주행거리를 보여주었다(표 1). 이것은 합승을 허용하는 경우, 허용하지 않는 경우보다 동일 운행 시간 기준으로 승객 승하차에 필요한 더 많은 지점을 경유하기 때문에 타당한 결과로 보여진다.

서비스 품질(LOS) 측면에서 볼 때, 수요가 낮은 경우엔 차이가 없지만, 수요가 높은 경우 합승을 자주 허용함으로써 이용자는 경로 우회로 인한 통행시간 증가를 감수하여야 한다. 그러나 이러한 단점은 시간대별 잠재 수요의 기종점 파악과 적절한 종류의 차종을 투입함으로써, 이용자 요금 할인이라는 측면에서 볼 때 충분히 극복 가능하리라고 본다. 또한, 본 연구에서 제안된 시뮬레이션은 모든 이용자가 합승에 참여할 의사가 있는 것으로 가정되었으나, 실제 적용에서는 승객의 일부만 참여할 의사가 있는 것을 감안할 때, 알고리즘을 일부 수정하여 승객에 따라 대기시간과 우회 제약조건을 개별적으로 부여할 수 있다.

## 6. 결론 및 향후 연구과제

본 연구에서는 수요대응형 교통수단에 연관된 알고리즘과 운영기법을 검증 및 평가할 수 있는 시뮬레이션 방안을 소개하고, 적합한 적용 예시로 Shared-Taxi 알고리즘을 제안하고 적용하였다. 본 연구에서 개발된 시뮬레이션 모델은 추가적인 네트워크 입력작업 없이 기존의 교통망을 시뮬레이션에 이용하는 수요대응형 교통수단의 시뮬레이션 기법 가능성을 제시하였다. 적용 예시를 위해 Shared-taxi(합승택시) 서비스를 이용하였으며 차량 운영을 위한 시뮬레이션 모형을 위한 알고리즘을 제시하였다. 시뮬레이션을 통하여 개발된 알고리즘 성능을 비교하였으며, 또한 합승을 고려하였을 때의 시스템 운영 효과를 생산성, 이용자의 대기시간과 통행시간의 관점에서 분석하였다. 그 결과로 택시는 대표적인 도심형 수요대응형 교통수단으로 합승(Shared-Ride)을 도모함으로써 효율성을 극대화할 수 있는 잠재성을 보여주었다. 제안된 시뮬레이션 방안은 대상지역의 교통 네트워크를 효율적으로 변환하여 적용할 수 있는 장점을 보여주었으나, 다른 한편으로 기존의 미시적 교통 시뮬레이션과 비교할 때, 교통 소통과 도로 신호체계를 고려한 차량 이동이 고려되지 않은 한계점 또한 내재하고 있다. 이를 위하여 추가적으로 동

적인 교통정보를 시뮬레이션 환경에 입력하는 방안이 고려되어야 한다.

향후 연구로는 제안된 시뮬레이션에 추가로 메타-휴리스틱 기법(meta-heuristic)을 적용하여 보다 나은 효율적인 시스템 운영을 도모하는 방안을 마련할 것이다. 아울러, 본 시뮬레이터를 이용하여 도심형 택시들이 전기차(Electric Vehicles: EV)로 전환되었을 경우, 전기차의 효율성 평가와 적절한 충전소 위치를 위한 연구로 사용될 수 있을 것으로 기대된다.

## 감사의 글

이 논문에 사용된 교통망 및 수요 데이터는 한국교통연구원 국가교통DB센터와 전기차연구센터의 연구결과물을 활용하였습니다.

## 참고 문헌

- 박상우, 장수은, 김혜자 (2008). 수요대응형 교통체계 평가 모형 구축: 복지교통서비스 평가지표 개발 및 적용을 중심으로. 한국교통연구원.
- Quadrifoglio, L., Dessouky, M., and Ordonez, F. (2008). A Simulation Study of Demand Responsive Transit System Design. *Transportation Research, Part A: Policy and Practice*, 42, pp. 718-737.
- Venglar, S. P., Fambro, D. B., and Bauer, T. (1995). Validation of Simulation Software for Modeling Light Rail Transit. In *Transportation Research Record 1494, TRB*, National Research Council, Washington, D.C., pp. 161-166.
- Abdulhai, B., Shalaby, A., and Georgi, A. (2002). Microsimulation Modeling and Impact Assessment of Streetcar Transit Priority Options: Toronto Experience. *Presented at 81st Annual Meeting of the Transportation Research Board*, Washington, D.C.
- Cortes, C. E., Pages, L., and Jayakrishnan, R. (2005). Microsimulation of Flexible Transit System Designs in Realistic Urban Networks. In *Transportation Research Record: Journal of the Transportation Research Board*, No. 1923, *Transportation Research Board of the National Academies*, Washington D.C., pp.153-163.
- Psaraftis, H. (1988). *Dynamic Vehicle Routing Problems. Vehicle Routing: Methods and Studies*. B. L. Golden and A. A. Assad (Editors). Elsevier Science Publishers B. V.
- Black, A. (1995). *Urban Mass Transportation Planning*. McGraw-Hill Series in Transportation. McGraw-Hill, New York.
- (접수일 : 2012. 4. 10 / 심사일 : 2012. 4. 12 / 심사완료일 : 2012. 5. 16)