

도시부 가로망에서의 링크 통행속도 기반 One-to-One 최단시간 경로탐색 알고리즘 개발

Development of One-to-One Shortest Path Algorithm Based on Link Flow Speeds on Urban Networks

김 태 형*
(Taehyeong Kim)

김 태 형**
(Taehyung Kim)

박 범 진***
(Bum-Jin Park)

김 형 수****
(Hyoungsoo Kim)

요 약

시간 종속적 가로망에 대한 최단경로 탐색은 ITS분야의 경로 일정계획과 실시간 내비게이션 시스템에서 중요한 부분을 차지한다. 본 연구에서는 매시간간격 변동적인 링크 통행속도를 고려하는 one-to-one 시간 종속적 최단시간 경로 알고리즘을 제시한다. 이를 위해, 먼저 기존의 일반적인 최단거리 경로 알고리즘 중에서 실제 도로망에서 비교적 빠르고 효율적인 알고리즘으로 알려져 있는 3가지의 알고리즘들, 즉, two queues 구조를 가진 Graph growth 알고리즘, approximate buckets 구조를 가진 Dijkstra 알고리즘, double buckets 구조를 가진 Dijkstra 알고리즘이 선택되었다. 이 알고리즘들은 모두 네트워크 내 하나의 노드에서 모든 노드(one-to-all)로의 최단거리 경로를 빠르게 탐색하기위해 개발되었다. 선택된 알고리즘들은 시간 종속적 도로망에 대해 하나의 출발노드에서 하나의 목적노드(one-to-one)로의 최단시간 경로 탐색이 가능하도록 확장된다. 또한, 제안된 3가지의 시간 종속적 최단시간 경로탐색 알고리즘들은 미국의 Anaheim, Baltimore, Chicago, Philadelphia 4개 도시의 실제 가로망에 적용하여 검증·평가된다. 결과적으로, 도시부 가로망을 대상으로 한 시간 종속적 최단시간 경로탐색 알고리즘으로 double buckets 구조를 가진 확장된 Dijkstra 알고리즘이 추천된다.

핵심어 : 교통약자, 정수계획법, 휴리스틱, 최적화, 다수차고지

Abstract

Finding shortest paths on time dependent networks is an important task for scheduling and routing plan and real-time navigation system in ITS. In this research, one-to-one time dependent shortest path algorithms based on link flow speeds on urban networks are proposed. For this work, first we select three general shortest path algorithms such as Graph growth algorithm with two queues, Dijkstra's algorithm with approximate buckets and Dijkstra's algorithm with double buckets. These algorithms were developed to compute shortest distance paths from one node to all nodes in a network and have proven to be fast and efficient algorithms in real networks. These algorithms are extended to compute a time dependent shortest path from an origin node to a destination node in real urban networks. Three extended algorithms are implemented on a data set from real urban networks to test and evaluate three algorithms. A data set consists of 4 urban street networks for Anaheim, CA, Baltimore, MD, Chicago, IL, and Philadelphia, PA. Based on the computational results, among the three algorithms for TDSP, the extended Dijkstra's algorithm with double buckets is recommended to solve one-to-one time dependent shortest path for urban street networks.

Key words : Time dependent shortest path problem, one-to-one shortest path, Dijkstra's algorithm, Graph growth algorithm

† 본 논문은 한국ITS학회 춘계학술대회(2012. 4. 27)에서 우수논문으로 선정된 논문에 기반합니다.

† 본 연구는 한국건설기술연구원 연구과제 “미시적 교통류 분석을 위한 ITS 모의실험 환경 개발”의 연구비 지원에 의해 수행되었습니다.

* 주저자 및 교신저자 : 한국건설기술연구원 첨단교통연구실 박사후연구원

** 공저자 : 한국교통연구원 교통시스템통합기술개발실 연구위원

*** 공저자 : 한국건설기술연구원 첨단교통연구실 수석연구원

**** 공저자 : 한국건설기술연구원 첨단교통연구실 수석연구원

† 논문접수일 : 2012년 8월 10일

† 논문심사일 : 2012년 8월 28일

† 게재확정일 : 2012년 9월 6일

I. 연구의 배경 및 목적

전통적인 최단경로 문제(Shortest Path Problem)는 최적화 문제의 중요한 부분을 차지하며 비즈니스, 물류, 교통, 지리, 그리고 컴퓨터 분야에서 수십 년 동안 폭 넓게 연구되어 왔다. 최단경로 문제는 크게 하나의 출발노드에서 다른 모든 노드까지의(one-to-all) 최단경로를 구성하는 방법과 모든 노드 사이의(all-to-all) 최단경로를 구성하는 방법으로 구분한다. 또한, 전통적인 최단경로문제에서는 링크의 통행속도와 통행시간은 일정한 것으로 가정한다. 그러나 도시지역 내에서는 가로망의 일정부분에서의 혼잡에 의해 하루 중에도 수시로 링크의 통행속도와 통행시간은 변하는 것이 사실이다. 따라서 매시간간격 변동적인 링크 통행 속도와 통행시간을 고려하기 위해서 최단경로 문제는 시간 종속적 최단경로 문제(Time Dependent Shortest Path Problem)로 확장되었다. 또한, AVL(Automatic Vehicle Location), 디지털 통신, 컴퓨터, GPS(Global Positioning System) 그리고 GIS(Geographic Information System)와 같은 첨단정보기술의 발달로 인하여 실생활의 폭넓은 분야에 시간 종속적 최단경로 문제의 적용이 가능해지고 있다. 또한, 시간 종속적 가로망에 대한 보다 빠르고 효율적인 최단경로 탐색 알고리즘의 개발은 교통, 물류, 그리고 ITS 분야에서 중요한 과제가 될 것이다. 특히, ITS 분야의 경로 일정 계획과 실시간 내비게이션 시스템에 적용하기 위해서는 실시간으로 빠른 one-to-one 최단경로 탐색이 가능해야만 한다. 시간 종속적 최단경로 문제에서는 모형이 first-in first-out(FIFO) 특성을 만족시켜야 하는데 FIFO특성은 똑 같은 링크를 운행하는 두 차량이 있을 경우, 링크내의 혼잡과 상관없이 출발시점에서의 순서대로 목적지점에 도착해야만 한다는 것을 의미한다.

따라서 본 연구는 FIFO특성을 만족함과 동시에 실제 도시 가로망에 적용 가능한 시간간격에 따라 변동적인 링크 통행속도를 고려하는 빠르며 효율적인 one-to-one 최단시간 경로 알고리즘을 개발하는데 목적이 있다. 이를 위해, 먼저 기존의 일반적인 최단

거리 경로 알고리즘 중에서 실제 도로망에서 비교적 빠르고 효율적인 알고리즘으로 알려져 있는 3가지의 알고리즘들, 즉, two queues 구조를 가진 Graph growth 알고리즘(TWO_Q), approximate buckets 구조를 가진 Dijkstra 알고리즘(DIKBA), double buckets 구조를 가진 Dijkstra 알고리즘(DIKBD)이 선택된다. 선택된 알고리즘들은 FIFO특성을 만족하면서 시간 종속적 가로망에 대해 하나의 출발노드에서 하나의 도착노드로의(one-to-one) 최단시간 경로탐색이 가능하도록 제안된다. 또한, 제안된 3가지의 시간 종속적 최단경로 알고리즘들은 미국의 Anaheim, Baltimore, Chicago, Philadelphia 4개 도시의 실제 가로망에 적용하여 검증·평가된다.

이 논문의 구성은 다음과 같다. 2장에서는 기존의 개발된 전통적인 최단경로 알고리즘과 시간 종속적 최단경로에 대한 연구를 검토한다. 3장에서는 전통적인 최단경로 알고리즘을 기초로 해서 개발된 3가지의 시간 종속적 최단경로 알고리즘들을 제안한다. 또한, 4장에서는 제안된 시간 종속적 최단경로 알고리즘들을 실제 도시부 가로망에 적용하여 그 결과를 비교한다. 마지막으로 5장에서 본 연구에 대한 결론과 앞으로의 연구방향을 제시한다.

II. 관련 연구

1. 전통적인 최단경로 알고리즘

One-to-all 최단경로 문제를 풀기 위한 알고리즘은 크게 표지고정(label-setting)과 표지수정(label-correcting) 기법으로 나누어진다.

전통적인 최단경로 문제에 대한 모형과 알고리즘들은 Bellman(1958)와 Dijkstra(1959)의 연구에서 시작되었다[1,2]. 이후에 Dial 외(1979)는 리스트(list) 구조체와 표지(labeling) 기술의 최단경로 알고리즘 성능에 대한 영향을 평가하였다[3]. Pallottino(1984)는 표지수정(label-correcting)기법의 하나인 graph growth 알고리즘을 개발하였으며, Gallo와 Pallottino(1988)는 다양한 자료구조를 적용한 8가지의 최단경로 알고리즘을 소개하고 컴퓨터 실험을 통해서

비교 평가하였다[4,5].

Cherkassky 외(1993)의 연구는 최단거리 경로 알고리즘들에 대한 가장 종합적인 평가 중의 하나로서 무작위로 만들어진 다수의 네트워크에서 17가지 one-to-all 최단거리 경로 알고리즘들을 평가했다[6]. 결과적으로 비음(nonnegative)의 아크길이를 가진 네트워크에 대해 표지고정(label-setting)기법의 하나인 DIKBD 알고리즘을 최적의 알고리즘으로 제안했다.

Zhan과 Noon(1998)은 앞서의 연구에서 평가된 17가지 알고리즘들 중에서 15가지의 알고리즘들을 실제 네트워크를 통해서 평가했다[7]. 결과적으로 one-to-all 최단거리 경로탐색을 위해서 표지수정(label-correcting)기법의 하나인 TWO_Q가 추천되었고, 표지고정(label-setting)기법인 DIKBA와 DIKBD는 one-to-one 또는 one-to-some 최단거리 경로탐색을 위해서 추천되었다.

또한, Zhan과 Noon(2000)은 앞서 Zhan과 Noon(1998)의 연구에 기초해서 one-to-one 최단거리 경로 탐색에 대해 DIKBA와 TWO_Q를 비교하였다[8].

최근에, 이미영 외(2008)는 통행시간, 환승횟수, 환승시간과 같은 복수의 제약 또는 목적의 고려가 요구되는 대중교통망을 대상으로 한 경로 탐색 기법을 제안했다[9].

2. 시간 종속적 최단경로 알고리즘

시간 종속적 최단경로 문제를 위한 모형과 알고리즘에 대한 연구는 Cooke와 Halsey(1966)에 의해 시작되었다[10]. 이후에 Orda와 Rom(1990)은 링크 지연이 시간에 대한 함수로 정의되는 네트워크에서의 최단거리 문제를 제안했다[11]. Ziliaskopoulos와 Mahmassani(1993)는 bellman의 최적화원리에 기초하여 표지수정(label correction)의 횟수를 줄이기 위해서 double-ended queue를 이용한 시간 종속적 최단거리 알고리즘을 개발했다[12].

Sung 외(2000)와 Ichoua 외(2003)는 기존의 불연속적인 통행시간과 비용함수에 기초한 모형들의 단점으로 first-in first-out(FIFO)특성을 만족하지 못함을 지적하고 통행시간 모형대신에 통행속도 모형을 제안했다[13,14].

3. 시사점

기존의 연구검토 결과, 비교적 큰 규모의 실제 가로망에 대한 one-to-all 최단거리 경로문제에서 표지고정 기법의 알고리즘 중에서는 DIKBA와 DIKBD, 표지수정 기법의 알고리즘 중에서는 TWO_Q가 비교적 계산시간이 빠르며 메모리 사용도 효율적인 것으로 나타났다. 차량의 경로일정계획, 긴급구조출동, 배송문제, 실시간 내비게이션 등을 포함하는 교통과 ITS분야에서는 one-to-one 최단거리 탐색이 필수적이며 one-to-all 최단거리 알고리즘은 one-to-one 최단거리 알고리즘으로 쉽게 변형이 가능하다.

시간 종속적 최단경로에 대한 연구에서는 개발된 알고리즘의 FIFO 특성 만족이 중요시되고 있으며 그동안 실제 가로망에 적용된 사례가 많지 않았다. 따라서 본 연구에서는 one-to-all 최단거리 알고리즘인 DIKBA, DIKBD, TWO_Q를 이용하여 one-to-one 시간 종속적 최단거리 경로 알고리즘을 개발하고 실제 가로망에 적용하여 검증·평가하고자 한다.

Ⅲ. 시간 종속적 최단거리 알고리즘

1. 링크 통행속도 모형

본 연구에서 통행시간은 시간대에 따라 변동하는 것으로 가정한다. 어느 한 지점과 다른 지점간의 통행시간은 양방향에 항상 같지는 않다. 링크 통행속도는 교통 센터로부터 실시간으로 매 10분마다 주어진다고 가정하고 시간 종속적 최단거리 알고리즘을 이용하여 출발지에서 목적지까지의 통행시간을 계산할 수 있다. FIFO특성을 만족시키기 위해서, Sung 등(2000)에 의해 개발된 링크 통행속도 모형과 도착시간함수를 다음과 같이 본 연구에 적용했다. 여기서, $Arrival_Time_TD(s_t, arc_{ij})$ 는 노드 i 에서 시간 s_t 에 출발했을 때 노드 j 에 도착하는 시간, $Link_Speed_Function(k, arc_{ij})$ 은 time interval이 k 일 때 링크 ij 의 링크통행속도, $IntervalLength$ 는 시간간격의 길이, $source$ 는 출발노드, $ending$ 은 도착노드, f_node 는 링크의 시작노드, t_node 는 링크의 끝노드를 말한다.

Function Arrival_Time_TD(s_t, arc_ij)

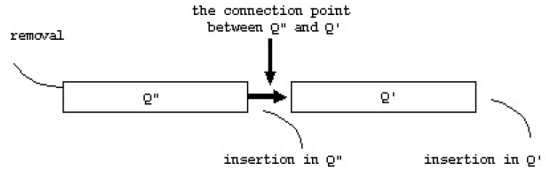
```

temp_ArrivalTime = s_t *1.0/60;
k = temp_ArrivalTime / IntervalLength;
Res_length = length of arc_ij;
temp_speed=Link_Speed_Function(k, arc_ij);
Res_length=Res_length-temp_speed/60
*((k+1)*IntervalLength-temp_ArrivalTime);
while(Res_length>0) do
begin
k = k +1;
temp_speed = Link_Speed_Function(k, arc_ij);
Res_length=Res_length-temp_speed/60
*IntervalLength;
end;
Arrival_Time=(k+1)*IntervalLength+Res_length/speed
*3600;
Return Arrival_Time;
    
```

실제 환경에서는 교통 센터로부터 매시간간격 링크 통행속도를 실시간으로 수집이 가능하나 본 연구에서는 매시간간격으로 변동하는 링크 통행속도를 다음과 같이 가정한다. 먼저 링크를 그들의 기능에 따라서 3가지 유형으로 나눈다. 즉, 첫 번째 유형인 고속도로는 60mph, 두 번째 유형인 간선도로는 40mph, 그리고 세 번째 유형인 집산도로는 30mph의 링크속도를 가진다. 첨두시간의 교통상황에 덜 영향을 받는 집산도로를 제외한 고속도로와 간선도로는 시간대마다 통행시간이 변동하는 것으로 간주한다. 매 시간간격은 10분으로, 하루는 144개의 시간간격으로 나누어진다.

2. E-TWO_Q 알고리즘

앞에서 설명된 링크 통행속도 모형과 도착시간 함수를 이용하여 two queues 자료구조를 가진 one-to-all Graph growth 알고리즘(TWO_Q)은 one-to-one 시간 종속적 최단경로 탐색 알고리즘(E-TWO_Q)으로 확장되었다. Two queues 자료구조를 가진 Graph growth 알고리즘(TWO_Q)은 Pallottino (1984)에 의해 최초로 소개되었다[3]. <그림 1>는 two-queue 자료구조를 예시하고 있으며 Zhan(1997)에서 참조하였다[15]. Two-queue 자료구조에서는 Q'와 Q''로 표



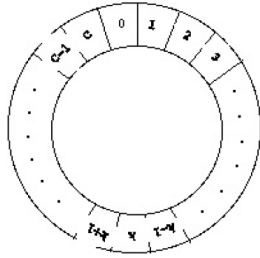
<그림 1> Two-queue 자료구조(Zhan, 1997)
<Fig. 1> Data structure of two-queue

현되는 두 개의 queue를 가지고 있어서 Q'와 Q''의 end에서 노드삽입, head에서 노드제거가 가능하다.

기존의 표지수정(label-correcting) 기법의 하나인 Graph growth 알고리즘은 one-to-all 방식으로 시작노드에서 다른 모든 노드까지의 최단경로를 탐색한다. 따라서 목적노드가 시작노드와 가깝다 하더라도 최단경로를 찾아내기 위해서 모든 노드를 탐색해야 하는 단점이 있다. 본 연구에서는 one-to-one 알고리즘을 위해 심층섭과 김진석(2002)의 연구에서와 같은 방법으로 시작노드에서 목적노드까지의 경로 값을 threshold 값으로 정해 이 값 이상으로의 노드 탐색을 방지하여 경로탐색 시간을 줄였다[16].

3. E-DIKBA 알고리즘

Approximate buckets 자료구조를 가진 Dijkstra 알고리즘(DIKBA)도 앞에서 설명된 링크통행속도 모형과 도착시간함수를 이용하여 시간 종속적 최단경로 탐색 알고리즘(E-DIKBA)으로 확장되었다. Dial (1969)은 처음으로 Dijkstra 알고리즘에 bucket 자료구조를 적용했다[3]. 원래의 Dijkstra 알고리즘에서는 표지된 노드들(labeled nodes)은 정렬되지 않은 상태의 목록(list)으로 관리되어 경로탐색의 반복과정 중에 minimum distance label를 가진 노드를 선택하기 위해 모든 표지된 노드들을 확인해야 하는 비효율적인 과정을 거쳐야 한다. 이러한 문제점을 개선하기 위해 bucket 자료구조를 적용하여 표지된 노드들을 관리함으로써 표지된 노드들을 distance label에 따라 정확하게 또는 근사적으로 정렬할 수 있다[8]. Approximate buckets 자료구조를 적용할 경우, 한 bucket내 distance label들의 값들은 정확하게 동일하지 않고 대신에 일정한 범위 안에 속하게 된다.



〈그림 2〉 Bucket의 자료구조(Zhan, 1997)
 〈Fig. 2〉 Data structure of bucket

〈그림 2〉는 bucket 자료구조를 예시하고 있다[15].

표지고정(label-setting) 기법의 하나인 DIKBA에서는 one-to-one 최단경로를 구할 경우, 일단 목적노드가 찾아지면 알고리즘은 종료되며 네트워크내의 모든 노드들을 검색할 필요가 없게 된다.

4. E-DIKBD 알고리즘

시간 종속적 최단경로 탐색을 위해 double buckets 자료구조를 가진 확장된 Dijkstra 알고리즘(E-DIKBD)은 E-DIKBA와 똑 같은 처리과정을 가지며 단지 다른 형태의 자료구조를 가진다. 즉, 상위레벨과 하위레벨로 이루어진 2개 레벨의 buckets를 적용한다. 검색과정에서 먼저 하위레벨의 bucket들내 모든 노드들이 검색된 후 상위레벨의 비워있지 않는 bucket들내 노드들을 해당하는 하위레벨의 bucket들로 이동시키고 다음 검색과정이 다시 시작된다.

IV. 제안된 알고리즘들의 비교

1. 실험대상 네트워크

본 연구에서 제안된 3가지의 시간 종속적 최단시간 경로탐색 알고리즘들은 실제 도시부 가로망에 적용하여 검증 평가된다. 네트워크 자료는 미국의 Anaheim, Baltimore, Chicago, Philadelphia 4개 도시의 실제 가로망 자료를 이용했으며, Anaheim, Chicago, Philadelphia의 자료는 Barger(2011)로부터 Baltimore의 자료는 상용패키지인 ArcLogistics Route의 GIS data를 이용했다[17].

〈표 1〉 실험대상 네트워크의 주요특성들
 〈Table 1〉 Characteristics of the test networks

네트워크	Anaheim	Baltimore	Chicago	Philadelphia	
노드수	416	63,356	12,982	13,389	
링크수	914	142,483	39,018	40,003	
링크/노드 비율	2.20	2.25	3.01	2.99	
링크 길이	Max.	1.786239	0.094554	9.99	7.25
	Min.	0.049896	0.02395	0.02	0.01
	Mean	0.50867	0.06198	0.69328	0.45776
	S.D.	0.33	0.05	0.64	0.45

*링크길이는 지리좌표시스템의 소수정확도를 따름(unit: mile)

실험대상 네트워크의 주요특성들은 <표 1>과 같으며 대상 네트워크 중에서 Anaheim은 노드와 링크의 수가 상대적으로 가장 적으며, Baltimore는 노드와 링크의 수가 가장 많다.

Zhan(1997)의 연구에서 실제 도로 네트워크의 중요한 특성중의 하나로서 링크/노드 비율로 측정되는 연결정도를 언급하고 있다[16]. 최단경로 트리를 구성하는데 필요한 검색 수는 링크/노드 비율과 직접적으로 관련이 있기 때문에 실제 도로 네트워크와 가상 네트워크 간 이러한 링크/노드 비율의 차이를 확인하는 것이 중요하다. 본 연구의 실험대상 네트워크에서는 링크/노드 비율은 2.20에서 3.01의 범위를 보이고 있다.

3가지의 알고리즘들은 C++ 프로그램 언어로 코딩되었으며 Cherkassky 외(1993)의 연구에서 제공된 one-to-all 최단경로탐색을 위한 C 소스코드를 이용하였다[6]. 입력 네트워크 자료는 forward star 형태로 표현되었으며 모든 컴퓨터 작업은 Window XP환경 내에서 2.0GHZ Intel Core 2 Duo CPU와 3GB 메모리를 가진 컴퓨터에서 실행되었다.

제안된 3가지 시간 종속적 최단시간 경로 알고리즘들의 성능비교를 위해서 실험대상 네트워크에서 무작위로 100쌍의 출발노드와 도착노드를 선택했다. 또한, 각 쌍의 출발노드와 도착노드에 대해 2개씩의 출발시간이 무작위로 선택되는데, 하나는 비침두시간대에, 다른 하나는 침두시간대에 속한다.

2. 실험결과

일반적으로 표지고정 기법인 DIKBA, DIKBD 와 표지수정 기법인 TWO_Q 간의 계산시간 차이는 검

색횟수와 검색을 위한 노드들을 선택하는데 필요한 시간에 좌우된다. DIKBA와 DIKBD는 TWO_Q에 비해서 상대적으로 노드 당 검색 수는 적은 편이지만 검색에 맞는 표지된 노드를 포함하는 하나의 bucket을 찾을 때까지 모든 bucket들을 연속적으로 확인해야 함으로 과도한 수의 빈 bucket들을 확인하는데 많은 시간을 소모할 수도 있다. 이에 비해, TWO_Q에서는 검색을 위해 한 노드를 선택하는데 필요한 시간은 최소화된다.

<표 2>는 실험대상 네트워크에 대한 제안된 3가지 시간 종속적 최단시간 경로 알고리즘들의 평균 계산시간에 대한 비교를 보여준다. 비교적 Anaheim과 Chicago 네트워크에서는 E-TWO_Q이 E-DIKBA와 E-DIKBD보다 4%~20% 더 빠르게 최단시간 경로를 탐색했고 E-DIKBD가 E-DIKBA보다 약간 더 빠른 것으로 나타났다. 반면에 비교적 규모(노드와 링크 수)가 더 큰 Baltimore와 Philadelphia 네트워크에서는 E-DIKBD가 E-TWO_Q와 E-DIKBA보다 6%~29% 빠른 것으로 나타났다. 즉, 소규모의 네트워크에서는 E-TWO_Q가 좀 더 빠르며, 대규모의 네트워크에서 E-DIKBD가 좀 더 빠른 것으로 판단할 수 있다. 이는 네트워크 규모가 커짐에 따라 E-DIKBD의 경우 상대적으로 적은 노드 당 검색수로 인하여 빠른 계산시간의 결과를 보이는 것으로 해석된다.

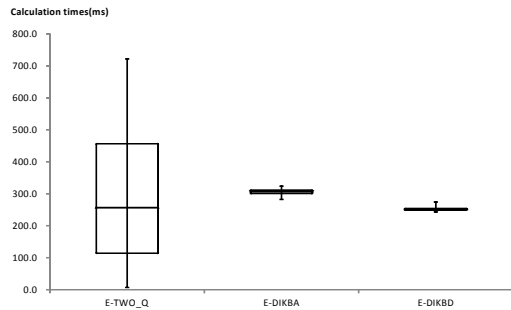
계산시간의 max/mean 값을 비교했을 때, E-TWO_Q의 max/mean 값이 E-DIKBA와 E-DIKBD의 값들보다 2배~3배 정도 큰 것으로 나타났다. E-TWO_Q가 E-DIKBA와 E-DIKBD에 비해서 통행시간의 길이에 따른 계산시간의 변동이 큰 것으로 해석되며 이는 E-TWO_Q 알고리즘의 경우, 출발노드와 목적노드 간의 통행시간의 길이에 따라 계산시간이 좌우됨을 보여주는 것이다. 또한, max/mean 값은 알고리즘의 예측가능성에 대한 척도로서 max/mean 값이 클수록 해당 알고리즘은 평균적인 노드 당 소요시간과 비교했을 때 어떤 출발노드들에서 상당히 긴 시간을 필요로 할 수 있음을 의미한다. 따라서 E-DIKBA와 E-DIKBD는 출발노드에 상관없이 일률적인 속도성을 유지하는 것으로 나타났다.

<표 2> 실험대상 네트워크에 대한 계산시간
(Table 2) Calculation times for the test networks

네트워크	E-TWO_Q		E-DIKBA		E-DIKBD	
	계산시간 (ms)	max/mean	계산시간 (ms)	max/mean	계산시간 (ms)	max/mean
Anaheim	1.52	1.98	1.84	1.63	1.80	1.11
Baltimore	289.30	2.50	305.51	1.06	252.27	1.09
Chicago	71.57	3.41	78.50	1.06	74.35	1.25
Philadelphia	96.97	2.52	79.67	1.05	75.41	1.09

첨두시간과 비첨두시간으로 구분하여 3가지 알고리즘의 계산시간을 비교하였을 때 첨두시간과 비첨두시간간의 계산시간의 차이는 거의 없었으며 <표 2>에서와 같은 결과를 보였다.

<그림 3>은 Baltimore 네트워크에 대한 계산시간의 Box Plot을 보여 준다. 역시, E-TWO_Q 알고리즘이 E-DIKBA와 E-DIKBD보다 통행시간의 길이에 따른 계산시간의 변동이 큰 것을 알 수 있다. 따라서 전체적인 결과를 통해서 비교적 큰 규모의 도시부 가로망을 대상으로 한 one-to-one 시간 종속적 최단경로 알고리즘으로 계산시간이 빠르며 출발노드에 상관없이 일률적인 속도성을 보이는 E-DIKBD 알고리즘이 가장 적합한 것으로 판단된다.



<그림 3> Baltimore 네트워크에 대한 계산시간의 Box Plot
(Fig. 3) Box Plot of calculation times for Baltimore, MD

V. 결 론

본 연구에서 시간간격에 따라 변동하는 링크 통행속도를 고려한 3가지의 one-to-one 시간 종속적 최단시간 경로 알고리즘들, 즉, two queues 자료구조를 가진 확장된 Graph growth 알고리즘, App-

roximate buckets 자료구조를 가진 확장된 Dijkstra 알고리즘, double buckets 자료구조를 가진 확장된 Dijkstra 알고리즘이 제안되었고 4개의 실제 도시부 가로망에 적용하여 검증 평가되었다. 실험결과, 소규모의 네트워크에서는 E-TWO_Q가 좀 더 빠르며, 대규모의 네트워크에서 E-DIKBD가 좀 더 빠른 것으로 나타났다. 또한, E-DIKBD는 E-TWO_Q에 비해서 출발노드에 상관없이 일률적인 속도성능을 유지하는 것으로 나타났다. 따라서 전체적인 결과를 통해서 도시부 가로망을 대상으로 한 one-to-one 시간 종속적 최단경로 알고리즘으로 비교적 계산시간이 빠르며 출발노드에 상관없이 일률적인 속도성능을 보이는 E-DIKBD 알고리즘이 추천된다.

본 연구에서 제안된 시간 종속적 최단경로 알고리즘은 교통, 물류, 그리고 ITS분야의 경로 일정계획과 실시간 내비게이션 시스템에 적용가능하다. 실제로 E-DIKBD는 Kim 외(2011)에 의해 제안된 다수 차고지를 가진 dial-a-ride problem(교통약자 차량의 경로와 일정문제)의 시간 종속적 최단경로 탐색에 적용되어 시스템 전체 경로와 일정계획에 대한 총 계산시간을 절약할 수 있었다[18,19].

참고문헌

- [1] R. E. Bellman, "On a Routing Problem", *Quart. Applied. Math.*, vol. 16, pp.87-90, 1958.
- [2] E. W. Dijkstra, "A Note on Two Problems in Connection with Graphs", *Numeriche Mathematik*, vol 1, pp.269-271, 1959.
- [3] R. B. Dial, F. Glover, D. Karney, and D. Klingman, "A Computational Analysis of Alternative Algorithms and Labeling Techniques for Finding Shortest Path Trees", *Networks*, vol. 9, pp.215-248, 1979.
- [4] S. Pallottino, "Shortest-Path Methods: Complexity, Interrelations and New Propositions", *Networks*, vol. 14, pp.257-267, 1984.
- [5] G. Gallo, and S. Pallottino, "Shortest Path Algorithms", *Annals of Operations Research*, vol. 13, pp.3-79, 1988.
- [6] B. V. Cherkassky, A. V. Goldberg, and T. Radzik, *Shortest Paths Algorithms: Theory and Experimental Evaluation*, Technical Report 93-1480, Computer Science Department, Stanford University, 1993.
- [7] F. B. Zhan, and C. E. Noon, "Shortest Path Algorithms: An Evaluation Using Real Road Networks", *Transportation Science*, vol. 32, no. 1, pp.65-73, 1998.
- [8] F. B. Zhan, and C. E. Noon, "A Comparison Between Label-Setting and Label-Correcting Algorithms for Computing One-to-One Shortest Paths", *Journal of Geographic Information and Decision Analysis*, vol. 4, no. 2, pp.1-11, 2000.
- [9] 이미영, 박제진, 정점례, 박동주, "사용자 맞춤형 대중교통 경로정보제공을 위한 다계층의 다목적 경로탐색기법 연구", *한국ITS학회논문지*, 제7권, 제3호, pp.1-4, 2008. 6.
- [10] K. L. Cooke, and E. Halsey, "The shortest route through a network with time-dependent inter-nodal transit times", *Journal of Mathematical Analysis and Applications*, vol. 14, pp.493-498, 1966.
- [11] A. Orda and R. Rom, "Shortest-Path and Minimum-Delay Algorithms in Networks with Time-Dependent Edge-Length", *Journal of the ACM*, vol. 37, pp.607-625, 1990.
- [12] A. K. Ziliaskopoulos, and H. S. Mahmassani, "Time-Dependent Shortest Path Algorithm for Real-Time Intelligent Vehicle Highway System Applications", *Transportation Research Record* no. 1408, pp.94-104, 1993.
- [13] K. Sung, M. G. H. Bell, M. Seong, and S. Park, "Shortest Paths in a Network with Time-Dependent Flow Speeds", *European Journal of Operational Research*, vol. 121, no. 1, pp.32-39, 2000.
- [14] S. Ichoua, M. Gendreau, and J.-Y. Potvin, "Vehicle Dispatching with Time-Dependent Travel Times", *European Journal of Operational Research*, vol. 144, no. 2, pp.379-396, 2003.

- [15] F. B. Zhan, "Three Fastest Shortest Path Algorithms on Real Road Networks: Data Structures and Procedures", *Journal of Geo-graphic Information and Decision Analysis*, vol. 1, no. 1, pp.69-82, 1997.
- [16] 심충섭, 김진석, "One-to-One 최단경로 알고리즘의 성능 평가", *정보과학회논문지: 시스템 및 이론*, 제29권 제11호, pp.634-639, 2002. 12.
- [17] Bar-gera, H., <http://www.bgu.ac.il/~bargera/tntp/>, Accessed, 2011년 8월 접속.
- [18] T. Kim and A. Haghani, "Model and Algorithm Considering Time-Varying Travel Times to Solve Static Multidepot Dial-a-Ride Problem", *Transportation Research Record*, no. 2218, pp.68-77, 2011.
- [19] T. Kim, *Model and Algorithm for Solving Real Time Dial-a-Ride Problem*, Ph. D. Dissertation, University of Maryland, 2011.

저자소개



김 태 형 (Kim, Taehyeong)

2012년 3월 ~ 현 재 : 한국건설기술연구원 첨단교통연구실 박사후연구원
 2002년 9월 ~ 2007년 7월 : Turner-Fairbank Highway Research Center(FHWA) 연구원
 2011년 8월 : University of Maryland 공학박사(교통공학전공)
 2000년 2월 : 경상대학교 대학원 도시공학 석사



김 태 형 (Kim, Taehyung)

2005년 11월 ~ 현 재 : 한국교통연구원 교통시스템통합기술개발실 연구위원
 1994년 4월 ~ 1998년 7월 : 서울시정개발연구원 도시교통연구부 연구원
 2005년 8월 : University of Maryland 공학박사(교통공학전공)
 1995년 2월 : 한양대학교 대학원 교통공학 석사



박 범 진 (Park, Bum-Jin)

2003년 3월 ~ 현 재 : 한국건설기술연구원 첨단교통연구실 수석연구원
 2010년 2월 : 연세대학교 대학원 도시공학 박사
 2003년 2월 : 연세대학교 대학원 도시공학 석사



김 형 수 (Kim, Hyungsoo)

2008년 2월 ~ 현 재 : 한국건설기술연구원 첨단교통연구실 수석연구원
 2007년 12월 : University of Maryland 공학박사(교통공학전공)
 1995년 2월 : 연세대학교 대학원 도시공학 석사