

■ 論 文 ■

고속도로 경로통행시간 산출을 위한 전진반복 전후방탐색법(PIFAB)의 개발

Progressive Iterative Forward and Backward (PIFAB) Search Method to
Estimate Path-Travel Time on Freeways Using Toll Collection System Data

남궁 성

(한국도로공사 도로교통기술원 수석연구원)

목 차

- | | |
|----------------------|------------------------|
| I. 서론 | 2. 전진반복 전후방탐색법 (PIFAB) |
| II. 관련연구 고찰 | V. 알고리즘 적용 및 분석 |
| III. TCS 통행시간의 문제점 | VI. 결론 |
| IV. 경로통행시간 산출알고리즘 개발 | 참고문헌 |
| 1. 단순 전방탐색법 (SF) | |

Key Words : 경로통행시간, 통행시간 추정(산출), 교통정보, 고속도로 통행료수납시스템 (TCS), 탐색법

요 약

총연장 3,000km를 바로 눈앞에 두고 있는 우리나라 고속도로는 과거와는 달리 출발/도착시간에 여러 개의 경로가 존재하는 다중경로시대를 맞이하게 되었다. 이에 효과적인 교통량분산을 위한 통행시간정보는 그 어느 때보다도 중요해지고 있다. 본 연구는 고속도로 통행료수납시스템(TCS)으로부터 수집되는 통행시간 자료(즉, TCS통행시간 자료)를 이용해서 고속도로 톨게이트간 경로 통행시간을 산출하는 방법에 관한 것이다.

TCS통행시간 자료는 고속도로를 통행한 모든 차량이 경험한 통행시간의 전수자료로서 정확도나 활용도가 매우 높을 뿐만 아니라, 별도의 설비투자가 필요치 않다는 장점을 가지고 있다. 그러나 이는 톨게이트간 출발시각과 도착 시각만을 가지고 있어 해당 출발/도착시간에 2개 이상의 경로가 존재하는 경우 어느 경로를 통행한 시간인지 알 수가 없으며, 더욱이 본선 주행시간외에 휴게소 체류시간, 램프 통행시간, 통행요금 정산을 위한 대기시간 등이 포함 되어 있어 올바른 경로통행시간 산출이 어렵다는 문제를 가지고 있다.

본 연구에서는 고속도로 톨게이트간 경로통행시간을 산출할 수 있을 뿐만 아니라 각 톨게이트와 센터간 자료통신량을 대폭 줄여줄 수 있는 전진반복 전후방탐색법(PIFAB)을 개발하여 그 이론적인 구조와 실제 적용결과를 살펴 보도록 한다.

The purpose of this paper is to develop a method for estimation of reliable path-travel time using data obtained from the toll collection system on freeways. The toll collection system records departure and arrival time stamps as well as the identification numbers of arrival and destination tollgates for all the individual vehicles traveling between tollgates on freeways. Two major issues reduce accuracy when estimating path-travel time between an origin and destination tollgate using transaction data collected by the toll collection system. First, travel time calculated by subtracting departure time from arrival time does not explain path-travel time from origin tollgate to destination tollgate when a variety of available paths exist between tollgates. Second, travel time may include extra time spent in service and/or rest areas. Moreover, ramp driving time is included because tollgates are installed before on-ramps and after off-ramps. This paper describes an algorithm that searches for arrival time when departure time is given between tollgates by a Progressive Iterative Forward and Backward (PIFAB) search method. The algorithm eventually produces actual path-travel times that exclude any time spent in service and/or rest areas as well as ramp driving time based on a link-based procedure.

I. 서론

고속도로 3,000km시대를 눈앞에 두고 있는 현재(2005. 2월기준 총 24개노선 2,923km), 과거 그 어느 때 보다도 고속도로 교통정보의 중요성이 날로 높아지고 있다. 특히, 신설노선의 증가로 동일한 출발도착시간(고속도로 톨게이트간을 기준으로 하여)에 2개 이상의 경로가 존재하는 고속도로 다중경로환경에서 "통행시간" 정보에 대한 요구는 매우 높아지고 있다.

통행시간 자료는 출발지점에서 도착지점까지 "소요된" 시간으로서 출발시각으로부터 도착시각까지의 기간 동안 시공간상의 정체상태 변화를 반영하고 있다. 이 점이 바로 차량감지장치(루프검지기, 영상검지기 등)로부터 얻어진 통행속도자료를 근거로 한 통행시간과 구별되는 점이다. 즉, 차량감지장치로부터 얻어진 정보는 지점정보로서 해당구간의 차량감지 지점으로부터 얻어진 자료를 이용해서 어느 시간단면에서의 해당구간 전체의 정체상태를 나타내는 것이며, 통행시간정보는 구간정보라 하여 이와 구분하기도 한다.

일반적으로 통행시간 정보는 지점정보를 구간정보로 가공 변환하여 이용하는 간접적인 방법과 해당구간을 주행한 차량이 경험한 통행시간자료를 직접적으로 이용하는 것이 있는데, 후자의 경우가 전자보다 정확하고 가장 효과적인 방법으로 알려져 있다. 그러나 문제는 지점정보로부터의 산출하는 방법이 아닌 주행 차량으로부터 직접적으로 통행시간 자료를 얻어내기 위해서는 막대한 시간과 비용을 들여 차량위치추적이나 번호판 인식과 같은 AVL(automatic vehicle location) 또는 AVI(automatic vehicle identification)기술을 이용한 시스템이 구축되어야 한다는 점이다.

그러나 고속도로를 통행하는 모든 차량의 출발, 도착 톨게이트간 통행시간 정보를 담고있는 고속도로 통행료수납시스템(TCS: toll collection system) 자료(이하, TCS자료)를 활용하는 경우, 별도의 통행시간 수집 인프라 구축을 위한 비용과 시간을 들이지 않아도 되는 장점이 있다.

한편, TCS자료를 이용한 통행시간 산출은 크게 단위시간 간격마다의 톨게이트간 통행시간 대푯값을 결정하는 전처리 과정과 그 대푯값을 바탕으로 경로통행시간을 산출하는 두 과정으로 나뉜다. 본 연구는 두 번째 과정에 초점을 맞춘 것으로 TCS자료가 고속도로 톨게이트간 "경로통행시간" 정보로서 활용되기 위해서 극복

되어야 할 문제를 적시하고, 이를 해결한 "반복전진 전후탐색법"을 제안한다. 이 방법은 TCS자료를 이용한 경로통행시간뿐만 아니라 고속도로 통행시간에 포함되어 있을 수 있는 휴게소 체류시간 등의 노상외 통행시간(고속도로 본선외의 통행시간)을 제거하는 단순하면서도 매우 효과적인 방법이다.

II. 관련연구 고찰

우리나라 유료 고속도로는 수도권 등 개방식 운영방식의 일부구간을 제외하고 대부분의 구간에서 출구 톨게이트에서 통행료를 정산하는 폐쇄식 운영방식을 채택하고 있다. 이와 유사한 방식으로 유료고속도로를 운영하고 있는 대표적인 국가로는 일본, 이태리, 스페인 등이 있으나, 폐쇄식 운영방식으로 얻어지는 TCS자료를 통행시간 정보로 활용하려는 시도로서 국제학술계에 발표된 논문으로는 우리나라와 일본의 경우를 들 수 있다. 우리나라의 경우, Yun(1999) 등은 우리나라 고속도로의 당시 156개 톨게이트를 대상으로 TCS자료를 이용한 교통정보서비스시스템(TISS: traffic information services system)을 처음으로 소개하였다. 1998년에 구축된 TISS에서는 15분단위(당시 대부분의 톨게이트 통행료정산서비가 노후하여 속도와 용량 그리고 통신속도 문제로 15분단위 이하로 하기가 어려웠음) TCS통행시간 자료를 수집하고, 이를 각 단위구간별로 속도로 변환한 다음, 루프검지기로부터 얻어진 속도데이터와 융합하고 이를 다시 통행시간으로 변환하여 활용하였다. 이와 같은 방법은 통행시간 산출 측면에서 볼 때, 여러 가지 문제점을 가지고 있는데, 이는 당시 TISS는 TCS자료를 이용해서 톨게이트간 통행시간을 얻어내는데 초점을 두었다기 보다도 당시 루프검지기가 설치되어 있지 않은 구간의 정체상태를 TCS자료를 통해 파악하기 위한 것이었기 때문이라 이해된다. 그 이후, 남궁성(2001) 등은 총 4단계의 TCS통행시간 자료의 전처리과정을 포함하여 TCS자료의 단위구간 합산법에 의한 통행시간 산출방법을 제안하였다. 본 연구는 남궁성(2001) 등이 제시한 알고리즘을 일부 개선하고, 단위구간 합산법에서의 통행시간 탐색구조를 이론적으로 밝힘과 아울러 실제 현장에 확대적용한 결과를 보여준 것이다. 그 외에 TCS 통행시간자료를 연구에 활용한 몇몇 사례가 있었으나, 모델정립을 위해 TCS자료를 주어진 그대로 이용했을 뿐, 경로통행시간 산출과정에

관한 것은 아니었다.

한편, 일본의 사례로서 Ohba(1999) 등은 TCS 자료를 이용한 통행시간 산출방법을 제안하였는데, 이는 빈도분포분석을 이용한 이상치 제거 및 이동평균법에 의한 데이터 결측처리 등 TCS 통행시간 자료의 전처리 과정에 초점을 맞추고 있을 뿐이다. 더욱이 동 연구에서는 전체 1일 데이터를 동시에 이용할 수 없는 실시간 운영환경을 고려하지 않았으며, 대상구간 또한 경로통행시간이 아닌 인접 톨게이트 1~2구간의 단거리 구간만을 대상으로 삼고 있다. 그리고 TCS 통행시간을 주로 루프검지기 등과 같은 차량감지장치가 설치되어 있지 않은 단거리 구간에서의 통행시간 산출을 위한 대안으로 고려하고 있는 등 본 연구와는 방향이 다소 다르다 하겠다. Ohba(2000, 2001) 등은 그 이후에 TCS 자료를 기반으로 통행시간 이력자료의 패턴이나 단순한 대기행렬 개념을 도입한 통행시간예측을 다룬 바 있으나, 통행시간산출은 그 이전의 연구에서 제시한 과정을 그대로 따르고 있다.

이 밖에 미국이나 유럽 등에서 고속도로의 주행차량이 경험하는 통행시간을 직접적으로 얻어내어 이를 통행시간정보로 활용하려는 노력이 있었으나 거의 대부분 전자통행료징수시스템에서 차량에 부착된 태그를 노변에 장치된 리더기에서 읽어들이는 방식을 둔 것으로 본 연구에서 다루고자 하는 내용과 다르다.

III. TCS 통행시간의 문제점

TCS 자료가 통행시간정보로서 활용되기 위해서는 다음 2가지의 특성이 반드시 고려되어야 한다. 첫째, 다중경로환경에서 TCS 통행시간은 경로통행시간이 아니다. TCS 통행시간은 고속도로 이용차량의 톨게이트 진입시각 및 진출시각의 차를 통해 계산된다. 그러나 진출입 톨게이트간에 2개 이상의 경로가 이용가능 할 때에는 해당 TCS 통행시간이 어느 경로에 대한 것인지 알 수가 없다. 둘째, TCS 통행시간에는 고속도로 본선을 주행한 시간외에 갓길 정차시간, 휴게소 방문에 따른 체류시간 등이 포함되어 있다. 이는 톨게이트간 거리가 멀수록 심해지는 경향이 있다. 이와 같은 2가지 특성은 TCS 자료를 경로 통행시간정보로 활용하는 것을 어렵게 만드는 요인이 되어왔다.

일반적으로 지적된 2가지 문제를 해결하는 방법으로 링크기반의 접근방법이 있다. 즉, 주어진 톨게이트간에

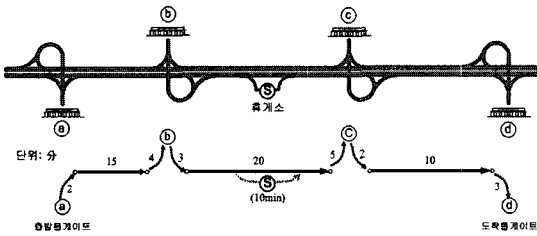
존재하는 여러경로 중 한 경로의 통행시간은 그 경로를 구성하고 있는 인접 톨게이트간의 구간(또는 단위링크), 즉, 링크들의 통행시간을 합산함으로써 얻어질 수 있다는 것이다. 또한 약 20~30km 이내의 짧은 거리를 갖는 단위링크의 통행시간에는 휴게소체류시간과 같은 본선 주행시간외의 시간이 포함되어 있지 않다는 가정을 전제할 수 있다면, 상기한 두 번째 문제도 해결될 수 있다. 즉, 고속도로에서 본선을 주행한 시간외로서 대부분을 차지하고 있는 것은 휴게소 체류시간이며, 실제로 인접한 톨게이트간의 평균거리가 20km내외(한편, 휴게소 설치간격은 30~50km임)의 짧은 거리라는 점을 감안하면, 타당한 가정이라 할 수 있다. 설사 단위링크의 TCS 통행시간에 휴게소 체류시간을 포함한 자료가 일부 반영되었다 하더라도 전체 경로통행시간에 미치는 영향은 미미하다고 봐도 무방할 것이다.

결론적으로 TCS 통행시간에서 경로통행시간(휴게소 체류시간 등이 포함되어 않은)을 산출하는 방법은 인접한 톨게이트간의 구간을 하나의 단위링크로 하여 링크 통행시간을 합산하면 될 것이다.

그러나 여기에도 해결해야할 두 가지 문제가 있다. 첫째, 각 링크 통행시간에는 본선 주행 시간 뿐만 아니라 톨게이트와 본선간의 진출입로(이하 램프)에서 소요된 시간까지 포함되어 있다는 것이다. 이것이 경로통행시간을 얻기 위해 링크통행시간을 합산해 가는 과정에서 누적되어 실제보다 경로통행시간이 크게 나오게 되는 것이다. 이는 경로의 거리가 길수록, 그리고 특히 램프에서의 정체가 심해지는 출퇴근시간대에 심해진다.

둘째, 고속도로 IC 분합류부에 정체가 있는 경우, 램프이용시간 뿐만 아니라 링크의 본선 구간을 통과하는데 소요되는 통행시간과 링크에 진입하거나 진출하는 차량이 경험하는 동일한 본선구간의 통행시간에 차이가 발생한다는 것이다. 즉, 링크단위의 TCS 통행시간을 이용해서 경로통행시간을 산출 시에는 램프이용시간뿐만 아니라 분합류부에서의 정체상태에 따라서도 결과가 달라질 수 있다.

이와 같은 문제를 예를 들어 설명하면 다음 <그림 1>과 <그림 2>와 같다. 우선 <그림 1>과 관련해서 몇 가지 가정을 두기로 한다. 첫째, 중간에 휴게소를 포함하고 있는 3개 구간(또는 링크)으로 이루어진 경로가 있다. 둘째, 각 링크(본선 및 램프구간 포함)에서의 통행시간은 변화하지 않고 고정되어 있다. 셋째, 1~2개 구간의 단거리를 통행하는 차량은 휴게소를 이용하지



실제 경로통행시간 = 2 + 15 + 20 + 10 + 3 = 50분
 TCS 경로통행시간 이용시 = (2 + 15 + 20 + 10 + 3) + 10 = 60분 (X)
 TCS 링크통행시간 이용시 = (2 + 15 + 4) + (3 + 20 + 5) + (2 + 10 + 3) = 19 + 28 + 15 = 64분 (X)

〈그림 1〉 TCS 링크통행시간을 이용한 경로통행시간 산출 문제점(1)

않으나, 3개이상의 구간을 통행하는 차량(즉, 경로 a-d)은 휴게소에서 10분의 체류시간을 갖는다.

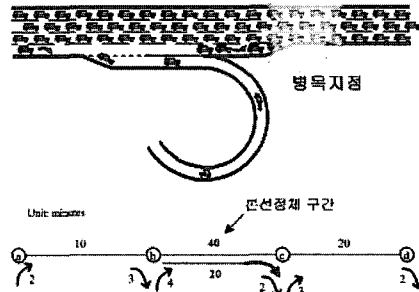
이와 같은 가정을 바탕으로 경로 a-d 간의 통행시간을 구하면 〈그림 1〉에서 보느바와 같다. 즉, 목표로 하는 경로 a-d의 통행시간은 3개 링크의 본선 주행시간 45분 (=15+20+10)과 출발 톨게이트와 도착 톨게이트에서의 램프이용시간 5분(=2+3)을 포함한 "50분"이어야 한다. 그러나 TCS 통행시간은 앞서의 가정에 따라 10분의 휴게소체류시간을 포함한 "60분"이 된다. 따라서 휴게소 체류시간을 포함하지 않은 링크통행시간을 합산하는 방법(다음 절에서 설명되는 SF탐색법임)으로 하면, "64분"이 나온다. 이는 각 링크의 TCS 통행시간이 포함하고 있는 램프이용시간(출발 톨게이트와 도착톨게이트에서의 램프이용시간은 제외) 14분(=4+3+5+2)이 합산되었기 때문이다.

〈그림 2〉는 앞서 지적한 두번째 문제점에 관한 것으로 단위링크를 통행한 차량이 경험하는 통행시간과 실제 우리가 얻고자 하는 단위링크의 본선구간 통행시간(즉, 해당구간을 지나쳐가는 차량이 경험하는 본선구간 통행시간)이 분합류부에서의 정체로 다를 수 있음을 보여주고 있다. 여기서 램프이용시간 뿐만 아니라 분합류부에서의 정체도 실제 경로통행시간과의 차이를 발생시키는 원인이 될 수 있음을 알 수 있다.

이러한 문제점들이 어떻게 해결될 수 있는지는 제4장2절에서 다시 〈그림 2〉를 토대로 다루어질 것이다.

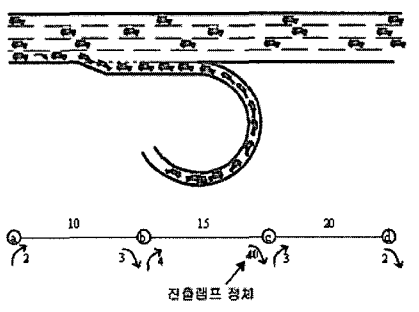
결론적으로 TCS 통행시간은 어느 경로를 이용한 것인지에 대한 정보가 없이 단지 출발/목적지 톨게이트간 통행시간만을 나타내어 다중경로환경에서 경로통행시간으로 활용할 수가 없고, 이를 해결하기 위하여 TCS 링크통행시간을 이용하는 경우, 단순히 링크통행시간을 합산해 가는 방식으로는 고속도로 폐쇄식 운영구간의

(a) 경우 1 : 본선 정체



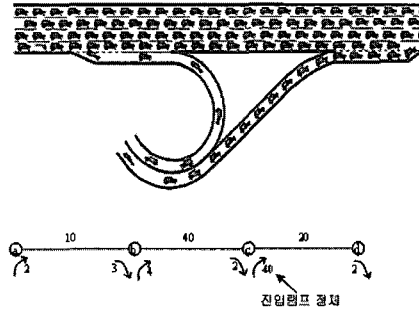
- 실제 경로통행시간 = 2 + 10 + 40 + 20 + 2 = 74 min
- TCS 링크통행시간 이용시 = (2 + 10 + 3) + (4 + 20 + 2) + (3 + 20 + 2) = 66 min (X)

(b) 경우 2 : 진출램프 정체



- 실제 경로통행시간 = 2 + 10 + 15 + 20 + 2 = 49 min
- TCS 링크통행시간 이용시 = (2 + 10 + 3) + (4 + 15 + 40) + (3 + 20 + 2) = 99 min (X)

(c) 경우 3 : 진입램프 정체



- 실제 경로통행시간 = 2 + 10 + 40 + 20 + 2 = 74 min
- TCS 링크통행시간 이용시 = (2 + 10 + 3) + (4 + 40 + 2) + (40 + 20 + 2) = 123 min (X)

〈그림 2〉 TCS 링크통행시간을 이용한 경로통행시간 산출 문제점 (2)

기하구조 특성상 램프구간 및 분합류부의 정체에 기인한 문제로 실제 경로통행시간을 산출하기가 어렵다는 것이다.

IV. 경로통행시간 산출알고리즘 개발

전 장에서 지적한대로 출발/도착 톨게이트간 TCS 통행시간을 그대로 경로통행시간으로 이용할 수가 없다. 대신 그 경로를 구성하고 있는 링크통행시간을 이용하게 되는데, 여기에는 각각의 링크통행시간에 포함되어 있는 램프이용시간을 어떻게 제거할 것인가 하는 문제와 분합류부 정체시 발생하는 본선구간 통행시간에 기인한 차이를 어떻게 해결할 것인가가 남는다. 본 연구에서는 우선 램프이용시간 등을 고려하지 않고, 단순히 링크통행시간을 순차적으로 합산해 가는 단순전방탐색법(SF: simple forward search, 이하 SF탐색법)을 위한 알고리즘을 제시하고, 이어서 램프이용시간을 제거할 수 있도록 SF탐색법을 개량한 전진반복 전후방탐색법(PIFAB: progressive iterative forward and backward search, 이하 PIFAB탐색법)을 살펴보기로 한다.

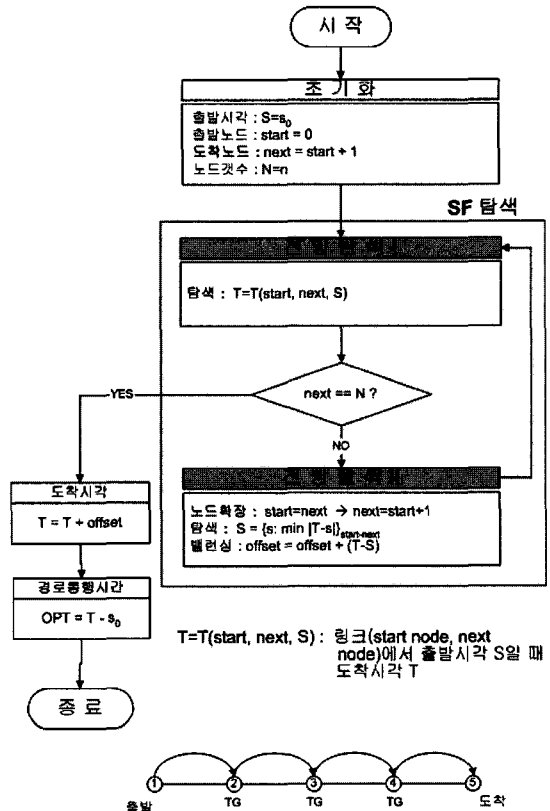
1. 단순 전방탐색법 (SF)

TCS 자료는 출발 톨게이트에서의 출발시각, 도착 톨게이트에서의 도착시각 정보를 가지고 있다. 따라서 경로를 구성하는 링크의 통행시간을 합산해 가는 것은 결국, 주어진 출발시각에 대응하는 도착시각을 탐색하는 것과 같다.

SF탐색법은 각 링크에서의 출발시각을 기준으로 도착시각을 출발 톨게이트로부터 시작해서 도착 톨게이트에 도달할 때까지 각 링크에서의 도착시각을 전방향으로 차례대로 탐색해가는 방법이다.

즉, <그림 3>에서 보는 바와 같이 우선 출발시각 S가 주어지면 첫 번째 링크의 TCS자료에서 주어진 출발시각 S를 갖는 레코드를 검색하고, 그 레코드에서 도착시각 T를 읽어 이를 그 다음 링크에서의 출발시각으로 하여 다시 도착시각을 읽는 과정을 경로의 마지막 링크에 도달할 때까지 반복한다. 이때, 마지막링크에서 읽혀진 도착시각(T)과 당초 주어진 출발시각(S)과의 차이가 경로통행시간(=T-S)이 된다.

그러나 TCS 통행시간 자료에서 출발시각은 자료의 집



<그림 3> 단순 전방탐색법(SF) 알고리즘

계단위(여기서는 5분)로 정렬되고, 도착시각은 분단위로 저장됨에 따라 직전 링크에서 읽혀진 도착시각을 그 다음 링크에서의 출발시각으로 하여 탐색할 때, 문제가 발생하게 된다. 즉, 도착시각이 우연히도 5분단위인 경우에는 문제가 없으나, 그렇지 않은 경우, TCS통행시간은 출발시각기준 단위시각(5분)단위로 정렬되어 있기에 일치하는 출발시각을 찾을 수가 없다. 이 때문에 <그림 3>에서 보듯이 offset이란 보정 변수를 두어 직전 링크의 도착시각과 가장 가까운 현재 링크의 출발시각과의 차이를 누적 합산하여 저장하고, 이를 나중에 통행시간에 가산해준다. 결국 경로통행시간은 "offset + (T-S)"가 된다.

2. 전진반복 전후방탐색법 (PIFAB)

전 장에서 지적한대로 TCS 통행시간에는 진출 톨게이트에서 발권하고 본선에 진입하기까지의 램프이용시간과 본선에서 벗어나 진출 톨게이트에서 정산하고 나가기까지의 램프이용시간이 포함되어 있다. 이 때문에 SF탐색법은 실제보다 과도한 경로통행시간을 산출하게

된다. PIFAB탐색법은 SF탐색법을 개량한 것으로 각 링크에서의 램프이용시간을 효과적으로 제외시켜주는 방법이다.

PIFAB탐색법을 구체적으로 살펴보기 전에 이해를 돕기위해 우선 앞서 살펴본 <그림 1>의 예시에 PIFAB 방법을 적용하면, 다음과 같다.

우선 a-c간의 통행시간 42분(=2+15+20+5)을 구하고, 여기에 b-c간의 통행시간 28분(=3+20+5)을 뺀 다음, 마지막으로 b-d간의 통행시간 36분(=3+20+10+3)을 더하면 결국, "50분"(=42-28+36)이 된다. 이는 실제 경로통행시간과 정확하게 일치한다. 한편, <그림 2>에서 예시한 3개의 경우에 대하여 PIFAB에 의한 계산 결과를 살펴보면 이 역시 실제 통행시간과 정확히 일치함을 볼 수 있다.

(경우 1)

$$(2+10+20+2)-(4+20+2)+(4+40+20+2)=74분$$

(경우 2)

$$(2+10+15+40)-(4+15+40)+(4+15+20+2)=49분$$

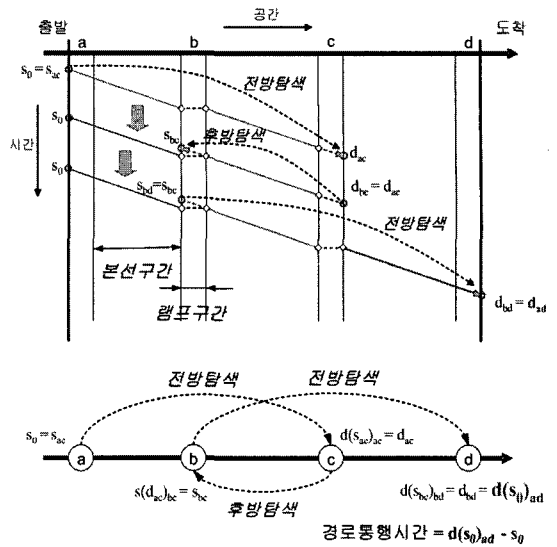
(경우 3)

$$(2+10+40+2)-(4+40+2)+(4+40+20+2)=74분$$

이는 각 링크에서의 통행시간이 불변이라는 가정 하에 계산된 것이고, 실제로 링크의 통행시간은 정체상태에 따라 변하며, 따라서 출발시각에 따라 통행시간은 다르다고 봐야 한다. 따라서 TCS통행시간 자료에서 이와 같은 계산을 하려면 단순히 각 통행시간을 더하거나 빼는 과정이 아니라 출발 톨게이트로부터 앞으로 전진하면서 출발시각 또는 도착시각을 탐색해 나가는 과정을 따르게 된다. 또한 이들 과정은 탐색이 도착 톨게이트에 도달할 때까지 전진하면서 반복적으로 탐색하되, 때로는 도착시각 탐색(전방탐색) 또는 출발시각 탐색(후방탐색)을 수행하므로 이를 "전진반복 전후방탐색법(또는 PIFAB탐색법)"이라 명명한 것이다.

PIFAB탐색법에 의한 탐색과정을 시공간상에 도해하면 <그림 4>와 같다. 이를 통해 앞서 <그림 1>과 <그림 2>의 예시를 바탕으로 PIFAB에 의해 계산된 통행시간이 실제 우리가 얻어야 할 경로통행시간과 어떻게 하여 정확하게 일치할 수 있었는지 설명된다.

우선 <그림 4>에서 보는 바와 같이 3개의 링크로 구성된 경로가 있다고 하자. 여기서 a-d구간의 경로통행시간에는 b와 c 톨게이트에서의 램프이용시간이 포함되지 않



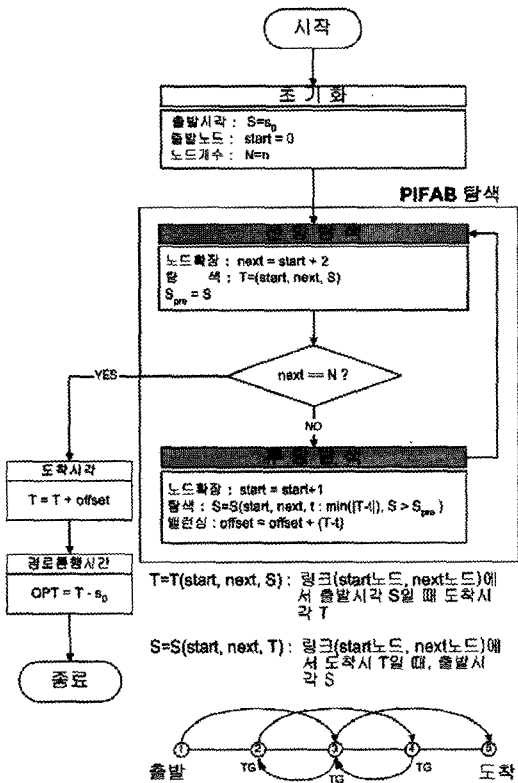
<그림 4> PIFAB탐색법의 시공간도해

아야 한다. 따라서 시공간에서의 차량 궤적은 이를 반영하여 b와 c의 램프구간에서 수평으로 나타나야 하는 것이다. 그러나 TCS 통행시간으로는 램프구간에서 수평이 되는, 즉, 램프구간에서의 통행시간이 없는 그러한 궤적을 얻어낼 수 없다. 그 이유는 본 연구에서 대상으로 하고 있는 고속도로 폐쇄식 운영구간에서 램프구간을 이용하지 않고 톨게이트에 진출입할 수 없기 때문이다.

하지만, PIFAB탐색법에 따르면 이와 같은 문제를 해결할 수 있다. 즉, 출발시각 s_0 가 주어지면, 그것을 a-c구간에서의 출발시각으로 하는 도착시각 d_{ac} 를 탐색한다. 다음으로 b-c구간에서 d_{ac} 를 도착시각 d_{bc} 로 갖는 출발시각 s_{bc} 를 찾는다. 마지막으로 b-d구간에서 s_{bc} 를 출발시각 s_{bd} 로 하는 도착시각 d_{bd} 를 탐색한다. 여기서 d_{bd} 는 결국 s_0 를 출발시각으로 할 때, a-d구간의 도착시각 d_{ad} 가 되며, 통행시간은 도착시각과 출발시각의 차이 $d_{bd}-s_0$ 가 된다.

이와 같은 PIFAB탐색법은 연이은 3개링크를 하나의 단위로 하여 도착 톨게이트에 도달할 때까지 반복된다. <그림 5>는 PIFAB탐색법 구현을 위한 알고리즘을 보여주고 있다.

PIFAB탐색 알고리즘은 크게 초기화(Initialization)와 전방탐색(Forward Search) 그리고 후방탐색(Backward Search)로 구분된다. SF탐색법과 다른 점은 후방탐색이 있다는 것과 노드확장(node expansion)과정이다. 즉, SF탐색법에서의 노드확장이 한발 한발 앞으로 나아가는 것이었다면 PIFAB탐색법에서는 두발 앞으로 나가



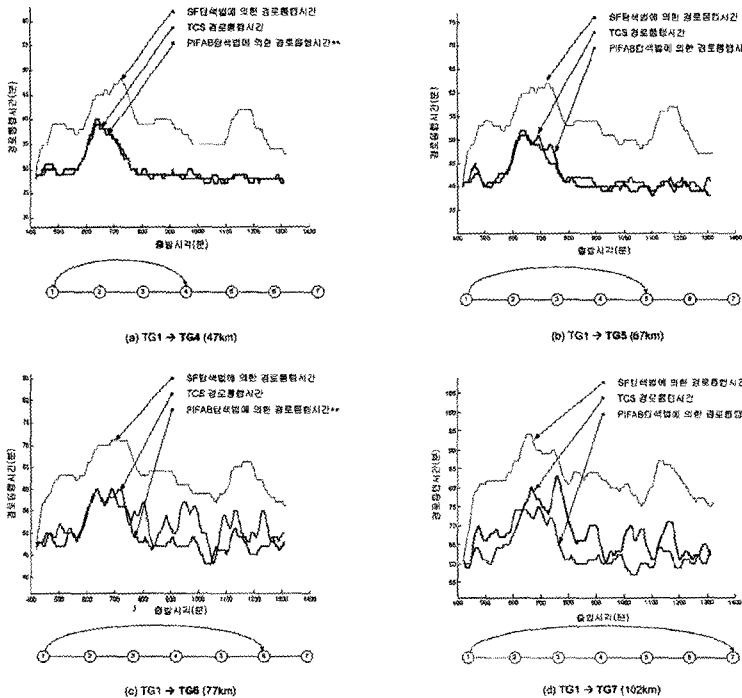
〈그림 5〉 전진반복 전후방탐색(PIFAB) 알고리즘

고 한발 뒤로 가고, 다시 두발 나가고 한발 뒤로 가는 식이다.

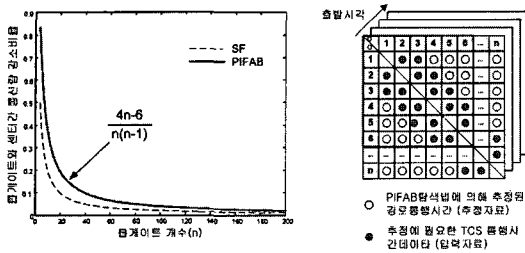
한편, 후방탐색과정에서 단순 SF탐색법 알고리즘에 서처럼 offset이라는 변수를 볼 수 있다. SF탐색법에서 offset변수는 출발시각(5분단위)과 도착시각(1분단위)간의 단위차이에서 발생하는 오차를 보정해 주기 위한 것이었다면, PIFAB탐색법에서의 offset변수는 도착시각 탐색시 완전히 일치하는 것이 없을 때, 가장 근접한 도착시각으로 하는데, 이때 발생하는 차이를 보정해 주기 위한 것이다.

V. 알고리즘 적용 및 분석

〈그림 6〉은 경부고속도로 서울~청주구간을 대상으로 각 톨게이트간 경로통행시간 산출을 위해 PIFAB 탐색법을 적용한 결과이다. 우선 PIFAB 탐색법의 최소 기본구간인 3개구간으로부터 1구간씩 늘려가면서 TCS 경로통행시간(TCS자료로부터 직접 얻어진 해당 톨게이트간 통행시간), SF탐색법에 의한 경로통행시간 그리고 PIFAB탐색법에 의한 경로통행시간을 비교하여 살펴보았다. 그 결과, 첫째, SF탐색법에 의한 결과는



〈그림 6〉 PIFAB 탐색법을 이용한 경로통행시간 산출결과 (경부선 서울-청주구간)



〈그림 7〉 PIFAB탐색법의 센터와 톨게이트간 통신량 감소비율

예상대로 램프이용시간 등의 누적합산 등으로 인해 과대하게 계산되고 있음을 볼 수 있다. 둘째, 3구간 및 4구간의 단거리 구간(거리로는 47km, 67km)에서는 TCS 경로통행시간과 PIFAB에 의한 경로통행시간과 큰 차이를 보이지 않고 있다가 5개구간을 넘어가면서 TCS 경로통행시간과 같거나 작은 값을 보이고 있다. TCS 경로통행시간은 휴게소 체류등과 같은 본선을 주행한 시간의 것을 포함하고 있을 수 있어 실제 경로통행시간은 이와 같거나 작게 나와야 함을 감안할 때, 타당한 결과라 할 수 있겠다. 참고로 통행거리가 길어질 수록 휴식 등을 위해 휴게소를 방문하는 차량의 비율이 늘어나고, 그에 따라 TCS 경로통행시간과 실제 경로통행시간의 차이가 커지게 되며, 상대적으로 적은 구간을 통행하는 경우, 그 차이는 반대로 작아야 한다. 〈그림 6〉의 결과는 그러한 것을 잘 나타내주고 있어, 본 연구에서 제안한 방법이 실제 현장에서도 잘 적용될 수 있음을 보여주고 있다.

한편, 단위시간마다의 고속도로의 모든 톨게이트간 경로통행시간을 산출은 막대한 계산량(특히 탐색회수)을 요구하는데, PIFAB 탐색법과 같이 링크기반의 접근방법은 그 계산량을 대폭 줄여줄 수 있는 장점이 있다.

예를 들어, 200개의 톨게이트를 가정했을 때, 총 OD쌍은 39,800(=200×199)가 되며, 이것은 결국 단위시간마다 각 톨게이트에서 센터로 보내는 TCS 통행시간 자료의 송신횟수이기도 하다. 그러나 PIFAB탐색법에 의한 경우, 794개(=4×200-6)의 OD쌍만 있으면, 이를 이용하여 전체 39,800개 OD간의 경로통행시간 계산이 가능하여(그림 7 참조), 톨게이트와 센터간 통신량을 대폭 줄일 수 있는 장점이 있다.

VI. 결론

본 논문은 우리나라 고속도로 폐쇄식 운영구간(진입

톨게이트에서 발권하고 진출톨게이트에서 정산하는 운영 방식)에서 수집되는 TCS 통행시간 자료를 이용하여 톨게이트간 경로통행시간을 산출하는 방법에 관한 것이다. 이를 위해 살펴본 SF탐색법은 링크기반으로 경로통행시간 산출할 때 이용되는 가장 일반적인 방법이나 램프이용시간 등으로 인하여 실제 경로통행시간에 비해서 과대하고 왜곡된 결과를 가져온다. 그러나 본 연구에서 제안한 PIFAB탐색법은 이론적으로 실제 경로통행시간과 거의 완벽하게 일치하는 결과를 가져다 줄 뿐만 아니라 실시간으로 각 톨게이트로부터 센터로 TCS자료를 전송하는 통신횟수를 대폭 줄여줄 수 있는 장점도 가지고 있다.

그러나 제안한 방법을 실제 적용한 결과를 살펴볼 때, PIFAB에 의한 결과가 얼마나 실제 경로통행시간과 일치할 수 있는지 검증하는 연구가 필요하다. 본 논문에서는 실제 경로통행시간을 얻을 수 없어, TCS 경로통행시간과의 논리적인 비교(적어도 실제 경로통행시간은 TCS 경로통행시간보다 같거나 작아야 한다 등의 논리)를 통하여 간접적으로 PIFAB탐색법에 의한 결과의 타당성만을 살펴보았다. 따라서 통제된 환경에서 결과를 살펴볼 수 있는 시뮬레이션에 의한 검증이나 실제 Probe차량을 이용한 검증방법을 고려해 볼 필요가 있다.

또한, PIFAB탐색법이 실제 현장에 확대·적용되기 위해서는 도착교통량이 적은 톨게이트를 포함한 단위링크의 TCS통행시간 결측처리가 관건이 될 수 있다. 특히, 그러한 구간에서 지정체가 일어나 전체 경로통행시간에 미치는 영향이 클수록 그 문제는 심각해진다. 이 문제는 최근 한국도로공사에서 2008년 전국확대를 목표로 추진되고 있는 전자통행료징수시스템인 하이패스(Hi-Pass)가 도입되는 경우, 교통정보수집 용도의 노변장치를 통해 얻어지는 하이패스 교통정보(즉, 하이패스 단말기를 장착한 차량들의 위치정보)를 이용한다면 충분히 해결될 수 있을 것으로 판단된다.

한편, 방법의 특성상 TCS 통행시간 자료에서 단위시간간격마다 출발 및 도착시각을 탐색하는 횟수가 많아(200개 노드 기준 약 80,000회), 이를 감안한 고성능의 시스템의 설계가 요구된다. 그러나 최근 서버컴퓨터 성능향상의 기술적 진보를 감안할 때, 문제가 없을 것이라 판단된다.

결론적으로 PIFAB탐색법에 의한 고속도로 경로통행시간 산출방법은 그 구조가 간단하면서도 링크기반의 경로통행시간 산출방법이 갖는 모든 단점을 해결하는 매우 유용한 방법이라 할 수 있다. 아울러 PIFAB 탐

색법은 우리나라와 같이 유료도로에서 폐쇄식 운영방식을 채택하고 있는 일본, 스페인, 이태리와 같은 나라에서도 유용하게 적용될 수 있을 것으로 판단된다.

참고문헌

1. 한국도로공사(2000), ITS기술개발연구(IV): 고속도로 통행시간 예측시스템 개발, 한국도로공사.
2. 남궁성·윤일수(2001), "나선형 단위구간 합산법에 의한 고속도로 순통행시간 산출기법", 대한교통학회 제40회 학술대회 논문집.
3. I. S. Yun, S. Namkoong, J. U. Choi, G. Y. Seong(1999), "Traffic information services system based on toll collection system data", Proceedings of 6th World Congress on Intelligent Transport Systems.
4. Y. Ohba, H. Ueno, M. Kuwahara(1999), "Travel time information calculation method for expressway using toll collection system data", Proceedings of 6th World Congress on Intelligent Transport Systems.
5. Y. Ohba, H. Ueno, M. Kuwahara(2000), "Travel time prediction method for expressway using toll collection system data", Proceedings of 7th World Congress on Intelligent Transport Systems.
6. Y. Ohba, H. Ueno and M. Kuwahara(2001), "Travel Time Prediction Method From Toll Collection System Data Based On Deterministic TIC Queueing Theory", Proceedings of 8th World Congress on Intelligent Transport Systems.

✉ 주 작 성 자 : 남궁성
 ✉ 논문투고일 : 2005. 6. 18
 논문심사일 : 2005. 7. 15 (1차)
 2005. 8. 17 (2차)
 심사판정일 : 2005. 8. 17
 ✉ 반론접수기한 : 2005. 12. 31