

■ 論 文 ■

수정형 덩굴망 최단경로 탐색 알고리즘을 이용한 다경로 생성 알고리즘의 개발

Generating Multiple Paths by Using Multi-label Vine-building Shortest Path Algorithm

김 익 기

(한양대학교 건설교통공학부 교수)

목 차

- I. 서론
- II. 수정형 덩굴망 알고리즘의 개념
- III. 다경로 생성 알고리즘
- IV. 다경로 대안노선의 생성 예제
- V. 결론
- 참고문헌

Key Words : 다경로탐색, k-shortest path, 덩굴망 알고리즘, 노선 정보, 확률적 노선배정모형

요 약

통행자들에게 과거, 실시간 혹은 예측 노선 정보 제공을 위한 정보 생성과 교통정보에 의한 교통영향 분석이 가능한 시뮬레이션 프로그램 개발에 관한 연구가 ITS 분야의 활발한 연구와 함께 과거 10여년에 걸쳐 진행되어 왔다. 또한 링크기반(link-based)이 아닌 경로기반(path-based)인 동적 확률적 노선배정 기법이 ITS 정책의 교통체계 상의 영향을 분석하는데 적합성이 높고, 또한 현실적 현상에 대한 설명력이 높다고 고려되어 이 분야에 대한 연구가 역시 활발해지고 있다. 이와 같은 현실적 노선정보의 생성, 교통체계분석을 위한 시뮬레이션 프로그램 개발 및 확률적 노선배정 기법 개발을 위한 핵심적 알고리즘 중의 하나가 합리적인 다경로 탐색 알고리즘인 것이다.

다경로 탐색 알고리즘에 대한 필요성과 요구가 커짐에 따라 현재까지 K-shortest path 알고리즘에 관하여 여러 학자들이 연구하여 왔다. 기존 연구 대부분의 알고리즘이 링크 제거 방법, 링크 저항가중 방법 및 시뮬레이션 방법에 의한 것이거나, 노선선택의 기준을 다양하게 변화시키는 방법 등을 적용하여 다경로를 탐색하는 방법을 제안하였다. 하지만 아직도 이 중에 어떠한 방법도 학자들 간에 공론화되어, 실무적으로 보편화되어 적용하는 방법은 없다고 고려된다. 본 연구에서는 기존의 K-shortest 탐색 알고리즘보다는 좀 더 행태적 접근방법을 접목시키는 다른 접근방법을 제안하였다. 즉 링크를 제거하는 등과 같이 교통 네트워크를 변경하지 않으며, 링크의 속성을 임의적으로 변경하지도 않으면서, 노선 선택의 동일한 기준을 일관되게 적용하면서도 다경로를 탐색하는 알고리즘을 제안하였다. 제안된 알고리즘의 핵심은 각 노드에 도달하는 각 방향별 경로에 대해 "이성적 한계범위(rational upper boundary)"라는 통행자들의 행태적 개념을 도입하여, 합리적인 대안 경로의 집합을 선정하는 방법인 것이다. 기존에 개발된 수정형 덩굴망 최단경로 탐색 알고리즘을 기반으로 한 다경로 생성 알고리즘을 제안하였다. 또한 본 논문에서는 다경로 생성 알고리즘의 개념의 이해를 돕기 위해 단순 네트워크를 예제로 합리적 대안경로 집합을 역추적 생성하는 과정을 단계별로 추가 설명하였다.

본 연구는 한국과학재단 목적기초연구(과제번호 RD1-2003-000-10764-0) 지원으로 수행되었음.

1. 서론

1990년 전후하여 ITS 정책 응용 및 연구에 대한 관심이 높아지면서 노선 교통정보 제공에 대한 연구가 활발해지면서 현실적인 경로탐색 및 노선기반(path-based) 통행배정(Traffic assignment) 기법에 대한 연구 관심도 높아졌다. 이와 같은 관심과 함께 ITS 분야 중 하나인 ATIS(Advanced Traveler Information Systems), DRGS(Dynamic Route Guidance Systems), 및 ATMS(Advanced Traffic Management Systems) 분야에서 통행자들에게 과거, 실시간 혹은 예측 노선 정보 제공을 위한 정보 생성과 교통정보에 의한 교통영향 분석이 가능한 시뮬레이션 프로그램 개발에 관한 연구도 함께 활발하게 진행되어 왔으며 또한 현재도 진행되고 있다. 이러한 현실적 노선정보의 생성과 교통체계분석을 위한 시뮬레이션 프로그램 개발에 있어 핵심적 알고리즘 중의 하나가 현실적인 최단경로탐색 알고리즘이며, 더 바람직한 것은 합리적인 다경로 탐색 알고리즘인 것이다. 현실적으로 각 도로 구간의 실시간 통행시간 정보가 TMC(Traffic Management Center)에 수집되었을 경우 최단경로 탐색에 의한 하나의 경로를 운전자에 제공하는 것보다는 소요되는 통행시간이 유사한 몇 개의 대안 노선을 통행자들에게 제공하는 것이 통행자들의 경로 선택 범위도 넓어지고, 또한 차량을 일정 노선에 집중되게 하는 과반응 현상(over-reaction)도 줄일 수가 있어 더 바람직한 접근방법으로 고려된다. 이와 같은 노선정보를 제공하기 위해 다경로 탐색 알고리즘에 대한 연구는 절실히 필요한 연구 분야인 것이다.

네트워크 분석 차원에서 링크기반(link-based)이 아닌 경로기반(path-based)으로 동적 노선배정 기법을 적용하는 것이 노선정보와 관련된 ITS 정책을 분석하는데 적합성이 높고, 또한 현실적 현상에 대한 설명력도 높다고 고려된다. 이와 같은 이유로 동적 확률노선배정 기법에 관한 연구가 ITS연구 활성화와 함께 활발해지고 있다. 하지만 대부분의 기존 확률적 노선배정(Stochastic Traffic Assignment) 기법들은 특정 O-D에 대한 대안 노선의 집합이 이미 주어졌다는 가정 하에서 분석되고 있는 것이 일반적이다. 그러므로 확률적 노선배정 분석에 앞서 대안노선 생성(Path Enumeration)이 우선 수행되어야 만이 분석이 가능해지는 것이다. 예외적인 경우로는 Dial 알고

리즘처럼 합리적 노선(reasonable path)에 대한 정의가 주어진 상태에서 노선배정 알고리즘이 수행될 때 동시에 대안노선 생성이 함께 계산되는 경우도 있기는 하다. 하지만 Dial 알고리즘에서 합리적 노선을 정하는 논리는 계산상의 용이성을 고려하여 그 논리가 단순하게 처리되어, 현실적 통행패턴 현상과는 다르게 대안 노선이 설정될 가능성이 높다. 따라서 현실적인 대안 노선 경로를 찾아내는 알고리즘의 개발은 확률적 노선배정 모형을 실무적 교통정책 분석 혹은 ITS 정책 분석에 적용하는 획기적인 변화를 줄 수 있는 핵심적 연구분야 중에 하나이다.

이상에서 서술한 것과 같이 ITS 분야에 대한 관심이 커지면서 다경로 탐색 알고리즘의 필요성이 함께 더욱 커져가고는 있으나, 효과적인 대안경로 생성 알고리즘 개발이 미진한 것이 현재 이 분야 연구의 현실이다. 기존 연구에서 K-shortest path 알고리즘의 명칭으로 여러 학자들에 의해 다양한 알고리즘이 제안되어 왔다. 기존 연구에서 제공된 알고리즘을 방법에 따라 구분하면 1)링크 제거 방법(link elimination approach), 2)링크 저항가중 방법(link penalty approach), 3)시뮬레이션 방법(simulation approach)으로 구분할 수가 있다. 이와 같은 기존에 개발된 다양한 K-shortest path 알고리즘에 대해서는 Park(1998), Ramming(2002)이 종합적으로 잘 요약 설명하였다. 이와 같은 기존의 방법 중에 세계 학자들 사이에 공론을 얻어 현실적 분석 응용에 보편적으로 사용되고 있는 방법은 아직 없다고 여겨진다.

기존의 연구에서는 앞에서 분류한 것과 같이 링크 속성의 변형, 링크 제거, 확률 변수의 적용 혹은 통행자 취향 등 다양한 경로선택 기준 적용 등과 같이 네트워크의 변경 또는 속성 변경, 그리고 경로설정 기준의 다양화(multi-criteria)를 통해 다경로를 탐색하는 방법들이었다. 하지만 본 연구에서는 경로를 선택하는 기준도 동일하게 유지하면서, Network의 변경이나 속성 변경 없이 경로의 일반화 비용의 차이로만 다경로를 탐색하는 알고리즘을 개발하고자 하였다. 네트워크의 변경 혹은 속성 변경 없이 동일한 경로선택 기준으로 다경로를 탐색하므로 임의적 수학적 속성변경 없이 계산이 이루어지며, 또한 통행자들의 행태를 기반으로 하여 좀 더 현실적인 다경로를 탐색할 것으로 고려된다. 즉 기존의 방법과는 다른 접근방법으로써 다노드표식(multiple node labeling)과 이

성적 한계범위(rational upper boundary)의 개념을 적용하여 합리적인 다경로 생성 알고리즘을 개발함으로써 현실적 이용의 가능성을 높여 보고자 하는 것이 본 논문의 연구 목적이다.

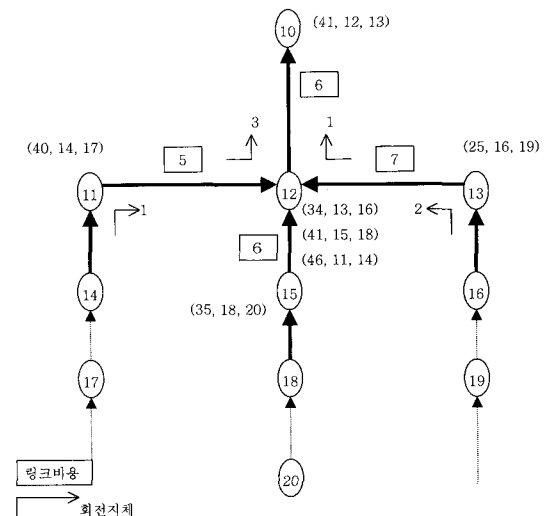
II. 수정형 덩굴망 알고리즘의 개념

수정형 덩굴망 알고리즘은 김익기(1997), Kim(2001)이 제안한 알고리즘이다. 이 알고리즘은 Ziliaskopoulos와 Mahmassani(1996)가 제안한 알고리즘과 실질적인 계산방법에서는 차이가 있지만 개념적으로는 유사한 알고리즘이다. 이 알고리즘은 Caldwell(1961), Kirby와 Potts(1969), Easa(1985), Thomas(1991), 노정현과 남궁성(1995), 최기주(1995), 이승환, 최기주와 김원길(1996) 등이 제시한 최단경로 탐색 알고리즘과 같이 교차로에서의 회전지체 및 회전금지를 반영하여 현실적인 최단경로를 찾아내는 데에 초점을 두었다. 다만 수정형 덩굴망 최단경로 탐색 알고리즘의 차이점은 네트워크 코딩 변경에 의한 방법이나 link labeling에 의한 방법이 아닌 node labeling 방법을 유지하며 multiple-label 방법에 의해 회전지체와 회전금지를 반영함과 동시에 U-turn 및 P-turn과 같이 동일한 node을 두 번 이상 방문하는 순환경로도 찾을 수 있도록 제안한 알고리즘이다.

수정형 덩굴망 알고리즘의 기본적 개념은 특정 출발점으로부터 해당 node i에 이르는 경로 가운데 node i와 연결된 각 방향 별 link를 통해 도달하는 방향 별 최단경로를 찾는 알고리즘이라는데 특징이 있다. 다시 말해서 수정형 덩굴망 알고리즘은 node labeling 방법으로 출발점으로부터 해당 node i의 전노드(predecessor)인 node j를 거쳐 도달하는 노선 가운데 최소 비용이 되는 노선의 비용, 전노드, 전전노드를 덩굴망 탐색에 따라 node label을 표기함으로써 회전금지, 회전지체를 반영한 최단경로의 탐색을 가능하게 한다. 또한 한 node에 여러 개의 node label을 표시함으로써 해당 node i로의 각 방향 별 최단경로를 모두 탐색할 수 있는 알고리즘인 것이다. 수정형 덩굴망 알고리즘은 최종 결과의 node label를 이용해서 역추적하면서 최종적인 최단경로를 기록하고 저장하게 된다. 이때의 역추적 과정은 특정 node k의 전전노드와 k node의 전노드 m의 전노드 번호를 일치시키면서 여러 개의 전노드 중에 회전을 고려한 최단

경로 하나를 선택함으로써 역추적을 지속하는 것이다. 이와 같이 회전 node에서 한 링크를 중첩 확인하면서 경로를 추적하므로 최단 경로 외의 경로를 추적하는 오류를 방지할 있다. 또한 연속된 3개의 node를 확인하면 회전에 대한 지체나 금지가 확실하게 반영되므로 2개 이상의 전노드에 대해서는 확인할 필요가 없다. 이와 같은 역추적이 출발점에 도달하면 추적 경로를 기록하고 알고리즘을 완료하게 된다. 이와 같은 역추적 방법은 회전 penalty와 회전금지가 정확하게 반영된 경로를 역추적하게 되면 현실적으로 논리적인 경로 탐색을 보장하게 된다.

이상에서 설명한 수정형 덩굴망 알고리즘의 개념에 대한 이해를 돕기 위해 다음 <그림 1>과 같은 단순한 예제 네트워크를 이용해 추가적 설명을 하고자 한다. 예제에서 특정 출발점으로부터 node 11까지 현재까지 탐색한 최단 경로의 비용이 40이고, 현재까지 출발점에서 탐색된 최단경로는 node 11의 전노드인 node 14와, 전전노드인 node 17로 역추적 될 수 있음을 node 11의 label (40, 14, 17)이 의미하고 있는 것이다. 출발점으로부터 node 12에 이르는 최단경로를 탐색하는 과정에서 node label을 붙일 때 수정형 덩굴망 알고리즘은 각 방향 별로 별도의 label을 붙이게 된다. 이 예제의 경우 3개의 label이 node 12에 붙게 된다. 그 하나는 node 11을 거쳐서 node 12에 도착하는 경로에 대한 label이고, 다른 두 개는 node 13과 node 15를 거쳐서 도착하는 경로이다. 각 방향



Node label : (경로비용, 전노드, 전전노드)

<그림 1> 수정형 덩굴망 알고리즘의 node label 예제

별로 각각의 최단경로에 대한 정보만 label에 포함되게 된다. 따라서 노드 12의 label 가운데 (46, 11, 14)가 의미하는 것은 전노드가 11인 방향에서 접근하는 경로 중 전전노드가 14인 경로가 최소비용을 나타내며, 이 경로를 통해 노드 12에 도달하는 비용은 노드 11까지의 비용 40과 링크 11-12의 비용 5 그리고 14-11-12 방향의 회전지체 1을 모두 더해 $40+5+1=46$ 의 경로 비용이 계산된 것을 의미하는 것이다. 마찬가지로 다른 2개의 노드 12의 label(34, 13, 16)과 (41, 15, 18)도 노드 13의 방향과 노드 15의 방향에서 접근하는 경로 중 최소 비용에 대한 정보를 기록한 것으로 노드 16과 18이 각 방향의 전전노드로 할 때 최소비용이 되며, 그 경로 비용은 각각 $25+7+2=34$, $35+6+0=41$ 이 됨을 의미한다.

예제 네트워크에서 최종 목적지인 노드 10까지의 최단경로 탐색은 오직 노드 12 방향에서 접근하는 경로 밖에 없다. 이와 같은 경우는 노드 label이 단 하나만 있게 된다. 하지만 노드 12를 거쳐 노드 10에 접근하는 경로는 3가지가 존재한다. 즉 노드 11에서 12를 거쳐 노드 10에 오는 경로 (경로비용 : $46+6+3=55$), 노드 13에서 12를 거쳐오는 경로 (경로비용 : $34+6+1=41$), 그리고 노드 15에서 12를 거쳐 오는 경로 (경로비용: $41+6+0=47$)가 있다. 수정형 덩굴망 최단경로 알고리즘은 이 3경로 중 가장 최소비용을 나타내는 경로에 대한 정보만 노드 10의 label에 기록하게 된다. 따라서 가능한 3경로 중 최소비용을 나타내는 41이 되는 경로인 노드 13에서 노드 12를 거쳐 노드 10으로 오는 경로에 대한 정보인 (경로비용, 전노드, 전전노드)=(41, 12, 13)가 노드 10의 최종 label이 되는 것이다.

위에서 설명한 것과 같이 label correcting 방법에 의해 최단경로의 탐색이 끝나고 최종 노드 label이 결정되면 각 노드에서 출발점까지의 최단경로를 역추적하는 것이 다음 과정이다. 역추적 방법을 예제 단순 네트워크를 이용하여 설명하면 최종 목적지 노드 10의 label (41, 12, 13)에서 노드 12가 전노드라는 것을 알 수 있으며, 노드 12에 이르는 여러 경로 중에서 노드 13 방향에서 접근하는 경로가 최단경로라는 것을 알 수 있다. 따라서 전노드인 노드 12에서 다시 역추적을 하게 되는데 노드 12의 3개 label 가운데 label (34, 13, 16)이 전노드가 13으로 노드 10의 전전노드 13과 동일하므로 최단경로의 label로

선정한다. 따라서 노드 12의 전노드인 노드 13에서 다시 역추적을 계속하게 되며, 이와 같은 역추적은 출발노드에 도달할 때까지 반복하게 된다. 최종 추적된 label은 목적지에서 출발점까지 역방향으로 차례로 기록 저장되어 출발점까지 역추적이 완료되면 최단경로의 노드가 순차적으로 저장(예 : 출발점--16-13-12-10)하면 된다. 수정형 덩굴망 최단경로 알고리즘에서는 위에서 설명한 것과 같이 multi-label 가운데 최단경로만 역추적하는 것으로 끝났으나, 다음 장에서는 역추적하는 과정에서 이성적 한계범위 (rational upper boundary) 개념을 적용하여 다경로 대안노선을 역추적 생성하는 알고리즘이 제안되었다.

III. 다경로 생성 알고리즘

수정형 덩굴망 최단경로 탐색 알고리즘에 의해 최종적으로 모든 노드에 대해 다노드표식(multi-labeling)이 완료되면, 각 노드의 label을 이용하여 하나의 출발점에서 다른 하나의 목적지까지 이르는 합리적인 여러 개의 대안 경로를 찾는 역추적 알고리즘을 제안하고자 한다. 제안된 역추적 알고리즘의 기본 개념은 목적지 노드에 도달하는 여러 방향의 접근 경로의 통행비용을 비교하여 그 중 최소비용과 다른 대안경로의 통행비용의 차이가 통행자의 대안노선 집합 허용 범위 안에 들어오면 대안집합(alternative set)안에 포함시키고, 그 범위를 벗어나면 대안집합에 포함시키지 않는 판단을 하는 것이 알고리즘의 첫번째 단계이다. 그 다음 단계로는 전노드를 거쳐 목적지에 도달하는 여러 개의 경로에 대해 최소비용의 경로와 비용 차이를 비교하며 합리적 대안노선 집합에 포함 여부를 앞에서와 같이 비교 판단한다. 즉 목적지의 전노드까지의 경로비용과 전노드 지점에서의 회전 벌점 (penalty) 그리고 전노드에서 목적지까지의 링크비용을 합한 비용을 각 방향 별로 계산 후에 그 비용을 최소비용과 비교한다. 만일 목적지의 전노드에서 합리적인 대안경로가 추가 설정될 경우 이 전노드를 "경로 가지노드(branch node)"로 기록한 후 나중에 이 지점부터 또 다른 대안경로를 역추적하게 된다. 이와 같은 방법으로 출발점에 도달할 때까지 계속 역추적하여 하나의 경로 추적을 마무리 짓게 된다. 한 경로의 추적이 완료되면 그 동안 설정된 경로 가지노드에 대해 하나하나씩 역추적하여 모든 합리적 대안경로에

대해 탐색이 완료되면 알고리즘의 계산 과정은 끝나게 된다.

제안된 다경로 생성 알고리즘에서는 대안경로의 통행비용이 최단 경로비용에 비해 한계범위 내에 포함되면 대안경로 집합의 하나로 통행자가 인식하게 되는 현상을 기본 개념으로 적용하였으며, 이와 같은 개념을 본 논문에서는 "이성적 한계범위(rational upper boundary)"라고 정의하였다. 이와 같은 개념은 알고리즘에 적용하는 방법에 있어서는 차이가 있지만, 통행자들의 대안노선 집합의 선택에 대한 심리적 배경 측면에서는 Mahmassani와 Peeta(1993)의 "rational boundary"와 유사한 개념이다. 구체적으로 다경로 생성 알고리즘을 정리하면 다음과 같다.

[표식 정의]

- $C_{rm}^{d(m)}$: 출발점 r에서 노드 m을 거쳐 노드 n에 이르는 최단 경로의 비용
- cost(p) : 경로 p의 출발점 r에서 도착점 s까지 경로 총 비용
- ϵ^{node} : 출발점 r에서 특정 node까지의 경로비용을 비교하여 합리적 대안경로로 포함 여부를 결정하는 이성적 한계범위의 상한 값
- ϵ^{path} : 출발점 r에서 도착점 s까지의 총 경로비용을 비교하여 합리적 대안경로로 포함 여부를 결정하는 이성적 한계범위의 상한 값
- flag(p) : 0 만일 경로 p가 현재까지 역추적되지 않으면,
: 1 만일 경로 p가 합리적 대안경로로 역추적 판단되었으면
- path(p, n) : 경로 p상에 있는 노드로써 목적지로부터 역방향으로 n번째 노드번호

[다경로 생성 알고리즘]

- (단계 1) 출발점 r과 목적지 s를 선정
- (단계 2) 초기화
 $path(p, n) = 0 \quad \forall p, n$
 $flag(p) = 0, \quad \forall p$
 $ppath = 1, np = 1$
- (단계 3) 출발점 r에 대한 각 노드의 최종 multi-label 값을 읽음

(단계 4) 목적지 s에서 대안 경로의 비용을 비교

- (1) 최단 경로를 제외한 s의 전노드 b_{rs}^d 하나를 선정
- (2) 선정된 경로비용 C_{rs}^d 과 최단 경로비용 $C_{rs}^{d(min)}$ 을 비교

만일 $C_{rs}^d - C_{rs}^{d(min)} \leq \epsilon^{path}$ 이면,

$$np = np + 1$$

$$cost(np) = C_{rs}^d$$

$$path(np, 1) = s, \quad path(np, 2) = b_{rs}^d$$

$$at(np) = 2$$

만일 아니면 (4)로 감

- (3) 최단 경로를 제외하고 처리하지 않은 경로가 있으면 (1)로 가고, 아니면 (4)로 감
- (4) 최단 경로에 해당되는 s의 전노드 b_{rs}^d 와 전전노드 bb_{rs}^d 을 읽고 기록

$$path(1, 1) = s, \quad path(1, 2) = b_{rs}^d,$$

$$path(1, 3) = bb_{rs}^d$$

$$p_{fw} = s, \quad pat = b_{rs}^d, \quad pbw = bb_{rs}^d, \quad m = 3$$

(단계 5) 노드 pat를 거쳐 노드 pfw에 이르는 대안 경로의 비용을 비교

- (1) pat의 전노드가 pbw이 아닌 전노드 B_{rs}^d 하나를 선정
- (2) B_{rs}^d 에서 pat을 거쳐 pfw로 가는 경로 비용과 최단 경로 비용을 비교

$$\Delta C_{p_{fw}} = C_{r, pat}^{d(B)} + t_{pat, p_{fw}} + p_{B, pat, p_{fw}} - C_{r, p_{fw}}^{d(pat)}$$

$$\Delta C_{rs}^{bpath} = cost(p_{bpath}) + \Delta C_{p_{fw}} - C_{rs}^{d(min)}$$

만일 $\Delta C_{p_{fw}} \leq \epsilon^{node}$ 이고, $\Delta C_{rs}^{bpath} \leq \epsilon^{path}$ 이면,

$$np = np + 1$$

$$cost(np) = cost(ppath) + \Delta C_{p_{fw}}$$

$$path(np, m) = B_{rs}^d,$$

$$path(np, i) = path(ppath, i), \quad i = 1, m-1$$

$$at(np) = m$$

만일 아니면, (3)으로 감

- (3) 만일 전노드가 pbw이 아닌 전노드 모두를 탐색하였으면 (4)로 가고,

만일 아니면, (1)로 가서 반복 계산

- (4) pat까지의 최소비용 경로를 역추적하여 pbw의 전노드 bb_{rs}^d 을 읽고 (단계 6)으로 감

$$m = m + 1, \quad path(ppath, m) = bb_{rs}^d$$

$$p_{fw} = pat, \quad pat = pbw, \quad pbw = path(ppath, m)$$

(단계 6) ppath에 대한 경로 역추적이 출발점 r에 도달 여부 검토

- (1) 만일 pbw=r 이면, flag(ppath)=1
만일 아니면, (단계 5)로 가서 반복 계산
- (2) 다음 역추적 대상의 경로를 선택
만일 flag(p)=0, p=np, 2 이 존재하며, 그 중 역순서로 첫번째 p를 선택하고,
ppath=p
만일 아니면, (단계 7)로 감
- (3) 새로운 ppath 경로 상의 역추적 시작 노드를 선택
m=at(ppath), pfw=path(ppath, m-1),
pat=path(ppath, m)
- (4) pat를 거쳐 pfw까지의 최단 경로가 되는 pat의 전노드 bb_s^d 을 기록하고, (단계 5)로 간다.
pbw = bb_s^d
m=m+1, path(ppath, m)=pbw

(단계 7) flag(p)=1 인 모든 경로 p를 대안 경로 집합에 순차적으로 보관 정리

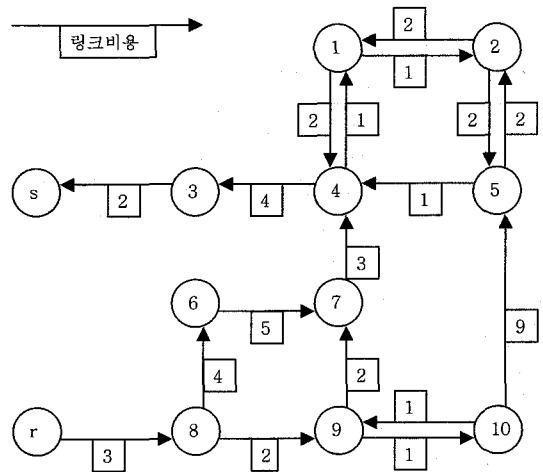
이 논문에서 제시한 합리적 다경로 생성 알고리즘에서 통행자들이 어느 한 노선을 합리적인 대안 경로로써 인식하는가 아닌가를 판단하는 규칙이 통행자의 행태적 기준에 근거한 이성적 한계범위의 개념이다. 최단경로와 기타 다른 경로와의 경로비용의 차이가 일정 규모 이상으로 크게 차이가 나면 통행자가 대안 경로로 고려하지 않고 선택 노선 대상에서 제외한다는 개념이다. 이때 이성적 한계범위의 상한 점은 최단경로 비용과의 차이 절대값으로 적용할 수도 있으며, 최단 경로비용에 대한 차이의 비율로써 상대적 값을 적용할 수도 있다. 또한 경로비용은 통행시간일 수도 있으며, 기타 다른 비용을 포함하는 일반화 비용(generalized cost)를 적용할 수도 있다. 절대값의 한계범위를 적용한다는 의미는 경로의 길이로부터 독립적으로 합리적 경로 여부를 결정한다는 가정이 내포되어 있는 것이며, 상대적 값인 비율을 적용한다는 의미는 경로 길이에 따라 허용 통행비용 차이가 변한다는 가정이 내포되어 있는 것이다. 만일 상대적 값인 비율을 적용할 때 경로의 길이가 짧은 경우 한계범위 상한 값이 비합리적으로 너무 적은 값이 될 수 있으므로, 이때에는 분석자가 임의로 적절한 크기의 값을 설정하여 적용하여도 될 것이다. 이와 같은

이상적 한계범위의 상한 값은 현실적 자료의 조사, 분석을 통한 심도 있는 경험적 실험연구 결과에 의해 설정되어야 할 것이다.

N. 다경로 대안노선의 생성 예제

다경로 생성 알고리즘의 내용과 기능을 좀 더 쉽게 이해하기 위해 <그림 2>의 단순 네트워크와 수정형 덩굴망 알고리즘 계산의 결과를 예제로 하여 설명하고자 한다. <표 1>은 예제 네트워크의 교차로에서의 회전지체와 회전금지를 요약 정리한 것이며, <표 2>는 수정형 덩굴망 최단경로 탐색 알고리즘으로 계산한 최종 결과인 각 노드의 multi-label 이다.

<그림 2>와 <표 1>의 예제 네트워크와 수정형 덩굴망 알고리즘의 최종 결과 <표 2>를 이용하여, 앞에서 설명한 다경로 생성 알고리즘을 각 단계별로 적용하고자 한다. 알고리즘의 (1 단계)에서 (3 단계)까지는 기계적인 작업으로 설명이 필요하지 않아,



<그림 2> 다경로 생성 알고리즘의 예제 네트워크

<표 1> 예제 네트워크의 회전지체 및 회전금지

From-node	Turn-node	To-node	회전지체 혹은 회전금지
r	8	6	6
2	5	2	900 (U-turn 금지)
7	4	3	6
8	9	7	900 (left-turn 금지)
9	10	5	3
10	5	4	3

〈표 2〉 수정형 덩굴망 알고리즘에 의한 예제 네트워크 노드의 Multi-label 최종 결과

노드 번호	노드 Label ($C_{rk}^d, b_{rk}^d, bb_{rk}^d$)	노드 번호	노드 Label ($C_{rk}^d, b_{rk}^d, bb_{rk}^d$)
1	(13, 4, 7) (22, 2, 5)	7	(18, 6, 8) (9, 9, 10)
2	(20, 5, 10) (14, 1, 4)	8	(3, r, 0)
3	(21, 4, 5)	9	(5, 8, r) (7, 10, 9)
4	(12, 7, 9) (17, 5, 2) (24, 1, 2)	10	(6, 9, 8)
5	(18, 10, 9) (16, 2, 1)	r (출발점)	(0, 0, 0)
6	(13, 8, r)	s(도착지)	(23, 3, 4)
최단 경로 : r - 8 - 9 - 10 - 9 - 7 - 4 - 1 - 2 - 5 - 4 - 3 - s			
최단 경로비용 : 23			

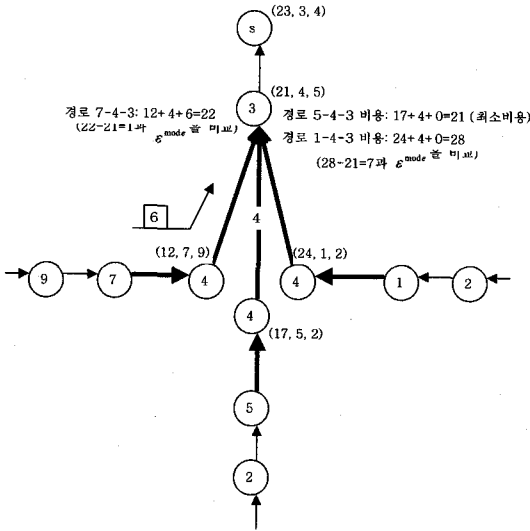
(단계 4)부터 예제에 적용하고자 한다. (단계 4)에서는 목적지 s에 접근하는 모든 방향 별 경로비용을 비교하여 합리적인 경로 대안의 집합에 포함 여부를 판단하는 과정으로 만일 대안경로로 적합하면 나중에 역추적할 수 있도록 표시해 두는 단계이다. 하지만 본 예제에서는 경로가 단 한 개밖에 없으므로 (단계 4)의 (4)만 수행하면 된다. 즉 〈표 2〉의 도착지 노드의 label에서 전노드와 전전노드를 읽어와 path(1,1)=s, path(1,2)=3, pfw=s, pat=3, pbw=4으로 표시한 후 목적지 s까지 경로에 대한 검토를 마무리하고 (단계 5)로 가면 된다.

(단계 5)는 노드 pat를 거쳐 노드 pfw에 이르는 여러 경로를 최단 경로와 비교하여 대안 경로집합에 포함시킬 것인가를 검토하는 단계이다. 현재까지 예제에서 pfw=s이므로 pat=3을 거쳐 s에 도달하는 최단 경로가 pbw=4으로 부터인 4-3-s가 유일한 경로이므로 노드 3까지의 대안경로 탐색을 마무리 짓고 단계 5의 (4)로 가서 m=m+1에 의해 m=4로 수정한다. 그리고 노드 3의 label에서 전전노드를 읽어 path(ppath, m)에 path(1,4)=5로 기록한 후, pfw=3, pat=4, pbw=5로 수정한다. 그 다음 (단계 6)의 (1)을 거쳐 다시 (단계 5)로 와서 pfw=3에 이르는 대안 경로에 대해 검토하는 과정을 반복하게 된다.

pat=4로 노드 4를 거쳐서 pfw인 노드 3에 이르는 경로는 앞의 경우와는 달리 최단경로를 제외하고

2개의 경로가 더 존재한다. 노드 4의 label 가운데 노드 3의 전전노드 5와 같은 노드 4의 전노드 값이 5인 label (17, 5, 2)은 일단 제외하고, 노드 4의 label (12, 7, 9)와 (24, 1, 2)에 대해 (단계 4)의 (1) (2) (3)을 먼저 수행하게 된다. 즉 이 단계에서 최단경로 외에 노드 4에서 가지 치는 다른 경로가 합리적인 대안경로의 집합에 포함될 수 있는가 여부를 “이성적 한계범위”의 개념에 의해 분석하게 된다. label (12, 7, 9)에 대해 우선 설명하면, (단계 5)의 (1)에서 현재 pat=4이므로 노드 4의 전노드 7를 읽어 온다. (단계 5)의 (2)에서 현재 pfw=3이므로 노드 7에서 노드 4를 거쳐 노드 3에 이르는 비용인 22으로부터 노드 3의 최소비용인 21을 뺀 값인 22-21=1과 “이성적 한계범위” 상한 값 ϵ^{node} 을 비교한다. 또한 동시에 이와 같이 새로이 탐색된 r에서 s까지의 경로 비용을 23+1=24와 같이 계산하여, 이 값에서 최단 경로 비용을 뺀 24-23=1을 “이성적 한계범위” 상한 값 ϵ^{path} 과 비교한다. 이와 같이 노드 3까지의 새로운 경로의 비용과 이 경로를 목적지까지 연장했을 경우의 경로 비용을 각각 최단경로와 비교하여 그 차이가 동시에 한계값 보다 적으면 합리적 대안경로 집합에 포함시킨다. 합리적인 경로로 인정될 경우 np=np+1에 의해 np=2로 수정되고, 현재 m=4이므로 path(np, m) 값이 7이 되어 path(2, 4)=7이 된다. 그리고 앞에서 s로부터 역추적한 경로를 가져와 path(2,1)=s, path(2,2)=3, path(2,3)=4를 기록하고 저장하고, (단계 5)의 (3)으로 간다. 만일 비용 차이가 한계범위보다 클 경우는 대안경로로 인식되지 않게 되므로 아무런 표시 없이 (단계 5)의 (3)으로 가면 된다.

노드 4의 label (24, 1, 2)에서 노드 3까지의 또 다른 경로가 존재함을 할 수 있으므로 이에 대해 앞에서의 방법과 동일하게 (단계 5)의 (1), (2), (3)의 과정을 다시 거치게 된다. 이 과정이 끝나면 노드 3까지의 최단 경로로써 노드 4의 label (17, 5, 2)에 대해 (단계 5)의 (4)를 앞에서와 같은 방법으로 수행하게 된다. (그림 3)은 이와 같이 노드 3에 도착하는 3개의 다경로를 역추적 탐색하고, 합리적 대안경로 집합에 포함 여부를 결정하는 과정을 도식적 개념으로 요약하여 설명한 것이다. 즉 노드 7에서 노드 4를 거쳐 노드 3에 도달하는 경로 7-4-3과 노드 1에서 노드 4를 거쳐 노드 3에 도달하는 경로 1-4-3을 최단 경로인 경로 5-4-3과 그 경로비용을 비교하여 그

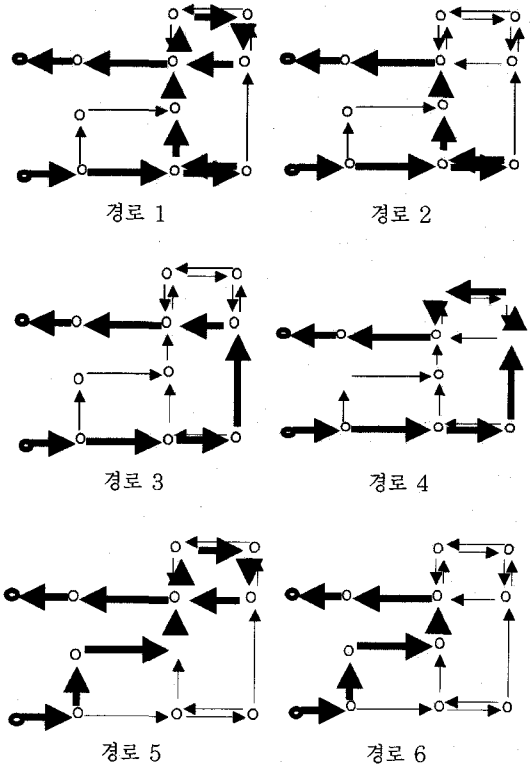


〈그림 3〉 예제 네트워크 중 노드 3에 이르는 경로 역추적 과정

차이가 이성적 한계범위 안에 포함되는지 여부를 검증하여 합리적 대안경로 집합 안에 포함 여부를 결정하는 것을 도식적으로 보여 준 것이다.

이상에서 설명한 것과 같은 방법으로 목적지에서부터 출발점까지의 경로에 포함된 노드를 뒤로 하나씩 역추적하여 경로 $ppath=1$ 이 출발점에 이르기까지 역추적을 반복 수행하게 된다. 또한 이와 같이 한 경로에 대한 역추적이 완료되면, 아직 역추적이 안되어 있는 $path(np, i)$, np 에 대해서도 $ppath=1$ 에서 수행한 방법과 같이 역추적을 하며, 더 이상 역추적할 대안 경로가 없으면 계산을 멈추고 알고리즘을 완료하면 된다.

“이성적 한계범위” ϵ^{node} 과 ϵ^{path} 값을 충분히 크게 하고(예를 들면 $\epsilon^{node}=9$ 이상, $\epsilon^{path}=10$ 이상), 예제 네트워크를 이용하여 다경로 생성 알고리즘을 수행하면 가능한 모든 가능한 경로가 다 추적되어, 〈그림 4〉



〈그림 4〉 알고리즘에 의해 탐색된 대안 경로

와 같은 6개의 대안노선이 추적된다. 〈표 3〉은 이와 같은 6개 대안 노선의 경로와 경로비용을 요약 정리한 것이다. 〈표 3〉에서 보는 것과 같이 최소비용 23에서 33까지 6개의 경로비용이 분포되어 있어 이성적 한계범위를 어떻게 설정하는가에 따라 합리적 대안경로 집합의 수가 바뀌는 것을 쉽게 알 수 있다. 이와 같은 이성적 한계범위를 $\epsilon^{node}=9$, $\epsilon^{path}=10$ 와 같이 절대값을 적용할 수도 있지만 [(대안 경로비용 - 최소 경로비용)/최소 경로비용]과 같이 상대적 값 즉 비율로 하여 $\epsilon^{node}=0.75$, $\epsilon^{path}=0.44$ 와 같이 적용

〈표 3〉 예제 네트워크의 대안 경로와 경로비용

대안 경로	출발점에서 도착점까지 대안경로의 순차적 노드 번호													비용 경로
	출발점	8	9	10	9	7	4	1	2	5	4	3	s	
Path 1	r	8	9	10	9	7	4	1	2	5	4	3	s	23
Path 2	r	8	9	10	9	7	4	3	s					24
Path 3	r	8	9	10	5	4	3	s						25
Path 4	r	8	9	10	5	2	1	4	3	s				30
Path 5	r	8	6	7	4	1	2	5	4	3	s			32
Path 6	r	8	6	7	4	3	s							33

할 수도 있다.

만일 $\epsilon^{node}=3$, $\epsilon^{path}=3$ 와 같이 앞의 예보다 적은 절대값 기준(혹은 $\epsilon^{node}=0.15$, $\epsilon^{path}=0.15$ 와 같이 적은 비율 기준)을 적용하여 다경로 생성 알고리즘을 수행하면, <표 3>에 있는 가능한 모든 6개 경로 중에서 오직 경로 1, 2, 3 만이 합리적 대안경로 집합에 포함되는 결과가 나온다. 따라서 현실적인 통행자들이 노선선택에 있어 대안 노선으로 고려하게 되는 “이성적 한계범위”를 경험적 실험 자료의 분석을 통해 현실적 값을 구하여야 할 것이며, 그래야만이 현실과 유사한 합리적인 대안경로의 수가 알고리즘에 의해 생성될 것이다.

V. 결론

ITS(Intelligent Transportation Systems) 분야의 관심이 높아지고, 점차 현실 교통정책에의 응용이 많아지면서, ATIS, DRGS 및 ATMS를 통한 노선정보 제공 및 통행시간 예측과 효과 분석에 대한 연구가 활발하게 진행되고 있다. 경로 정보 제공을 위해 필수적인 분석 기법의 하나가 최단경로 탐색 알고리즘이다. 특히 통행자들의 욕구를 만족시키기 위해서는 하나의 수학적 최단경로에 대한 정보뿐 아니라 유사한 비용의 여러 개의 대안 경로에 대한 정보 제공이 요구되고 있다. ITS 응용 분야 뿐 아니라, 네트워크 분석 연구 분야에서도 확률적 노선배정 분석에 대한 필요성이 ATIS 분야 분석과 함께 많이 요구되고 있다. 최근에 이와 같은 확률적 노선배정 기법에 관한 연구는 활발하나 대부분의 연구가 대안경로 집합이 주어졌다는 가정을 기본으로 하고 분석을 하고 있다. 이와 같은 필요에 의해 K-shortest path 알고리즘에 대한 연구들이 여러 학자들에 의해 이루어졌으며 대부분의 연구는 대부분 링크 제거 방법, 링크 저항가중 방법 및 시뮬레이션 방법에 의한 것이거나, 노선선택 기준을 다양하게 변화시키는 방법 등에 의해 다경로를 탐색하고 있다. 하지만 이와 같이 제시된 방법 중에 아직 학자들 사이에 공론화되어 실무적으로 보편화하여 적용되는 방법은 없다고 고려된다. 본 연구에서는 기존의 K-shortest 탐색 알고리즘과는 달리 통행자의 행태를 집목시킨 방법을 제안하였다. 즉 링크를 제거하는 등과 같이 교통 네트워크를 변경하지 않고, 또한 링크의 속성을 임의적으로

변경하지도 않으며, 동일한 노선 선택의 기준을 일관되게 적용하면서도 다경로를 탐색하는 알고리즘을 제안하였다. 알고리즘의 핵심은 각 노드에 도달하는 각 방향별 경로에 대해 “이성적 한계범위(rational upper boundary)”라는 개념을 도입하여 합리적인 대안 경로의 집합을 선정하는 것이다.

제안된 다경로 생성 알고리즘은 수정형 덩굴망 알고리즘을 기반으로 하였기 때문에 교차로에서의 회전지체 및 회전 금지를 반영하며 최단경로를 탐색할 수 있으며, 또한 일반적 최단경로 알고리즘에서 동일한 노드를 두 번 이상 방문할 수 없다는 제약 또한 극복함으로써 순환 경로인 U-turn 및 P-turn에 의한 현실적 최단경로 대안을 탐색할 수가 있다. 그리고 이성적 한계범위의 상한 값을 어떻게 설정하는가에 따라 대안경로의 수를 많게 또는 적게 탐색할 수 있어 통행자들의 노선 대안에 대한 인식을 현실적으로 묘사할 수 있는 알고리즘 상의 특성을 갖고 있다. 본 연구에서 제안한 다경로 생성 알고리즘은 현실의 노선정보 관련 교통정책이나 교통 네트워크 분석 관련 연구에 다양하게 응용될 수 있다고 고려된다.

이 논문에서는 다경로탐색 알고리즘의 방법론에 초점을 두고, “이성적 한계범위”와 같은 입력 자료의 값은 일반화하여, 현실의 실험적 연구에서 현실적 값이 조사 분석될 경우 항상 적용할 수 있도록 일반적 방법론을 제시하였다. 향후 연구의 방향으로 크고 복잡한 교통 네트워크에서 제안된 알고리즘이 컴퓨터 계산 속도 측면에서 현실성을 갖고 있는지에 대한 실험적 연구와 계산된 경로가 현실적 경로와 얼마나 유사할 것인가에 대한 연구를 진행할 예정이다. 또한 통행자들이 실제로 선택 가능한 대안 경로로 인식하게 되는 이성적 한계 범위가 어떠한 값을 가질 것인가에 대해 연구가 필요하다. 따라서 경험적 조사 자료를 통해 이성적 한계 범위의 값을 추정하는 연구를 향후 연구로 진행한 후 그 값을 이용하여 알고리즘이 얼마나 현실적인 대안 경로를 생성할 수 있는가를 검증하는 연구를 진행하고자 한다.

참고문헌

1. Caldwell, T.(1961), On finding minimum routes in a network with turn penalties, Communications of the ACM, Vol.4, No.2, Feb, pp.

- 107~108.
2. Easa, S. M.(1985), Shortest route algorithm with movement prohibitions, Transportation Research B, Vol.19B, No.3, pp.197~208.
 3. Kirby and Potts(1969), The minimum route problem for networks with turn penalties and prohibitions, Transportation Research, Vol.3, pp.397~408.
 4. Kim, Ikki(2001), A Multi-label Vine-building Algorithm for the Shortest Path Problem with Turn Penalties and Prohibitions, Transportation Research Board 80th Annual Meeting, Proceeding CD-ROM.
 5. Park, Dongjoo(1998), Multiple path based vehicle routing in dynamic and stochastic transportation networks, Texas A&M University, Ph.D thesis
 6. Ramming, Michael Scott(2002), Network knowledge and route choice, MIT, Ph.D. thesis.
 7. Thomas, R.(1991), Traffic Assignment Techniques, Avebury Technical
 8. Ziliaskopoulos and Mahmassani(1996), A note on least time path computation considering delays and prohibitions for intersection movements, Transportation Research B, Vol.30, No.5, pp.359~367.
 9. 노정현·남궁성(1995), "도시가로망에 적합한 최단경로탐색 기법의 개발", 대한국토·도시계획학회 지 국토계획, 제30권 제5호.
 10. 김익기(1998), "ATIS를 위한 수정형 덩굴망 최단경로 탐색 알고리즘의 개발", 대한교통학회지, 제16권 제2호, 대한교통학회, pp.157~167.
 11. 이승환·최기주·김원길(1996), "도시부 ATIS 효율적 적용을 위한 탐색영역기법 및 양방향 링크탐색 알고리즘의 구현", 대한교통학회지, 제14권 제3호, 대한교통학회, pp.45~59.
 12. 최기주(1995), "U-Turn을 포함한 가로망 표현 및 최단경로의 구현", 대한교통학회지, 제13권 제3호, 대한교통학회, pp.35~52.

✉ 주 작성자 : 김익기

✉ 논문투고일 : 2004. 1. 27

논문심사일 : 2004. 3. 23 (1차)

2004. 4. 1 (2차)

심사판정일 : 2004. 4. 1

✉ 반론접수기한 : 2004. 8. 31