

EGOSST를 이용한 이동 경로의 가중치를 반영한 효과적 연결

김 인 범[†]

요 약

본 논문에서는 EGOSST를 이용하여 가중치를 갖는 이동 경로들을 최소 비용으로 모두 연결하는 방법을 제안한다. 이동 경로는 가중치 선분으로 변환될 수 있는데, 이것은 통신선, 도로 및 철도망에서의 동적 케적뿐 만 아니라, 가중치인 이동 량이나 통행 빈도를 포함한다. 제안되는 방법은 단순한 위치 정보만을 고려하여 처리하는 방법에 비해 더 광범위하고 유용한 분야에 응용이 가능할 것이다. 입력 선분의 수, 각 선분 가중치의 최대 크기, 그리고 그리드 정밀도를 입력 인자로 설정한 실험에서, 본 논문에서 제안된 방법은 가중치 최소 신장 트리를 이용한 방법과 비교할 때, 연결 비용은 평균 1.07%, 가중치 스타이너 최소 트리 방법에 비해서는 평균 0.43% 감소하였다. 또한 그리드 정밀도를 0.1과 0.001로 했을 경우, 가중치 최소 신장 트리 방법에 비해 실행 시간이 각각 평균 97.02%, 2843.87% 증가했으나, 연결 비용은 각각 평균 0.86%, 1.13% 감소되었다. 이는 제안된 방법이 가중치를 반영한 이동 경로의 효과적 연결 뿐 아니라, 그리드 정밀도를 조절하여 생성 시간과 비용 절감을 응용 분야에 맞추어 사용될 수 있음을 보인다.

키워드 : EGOSST(Edge Grade Of Services Steiner Minimum Tree), 선분 최소 스타이너 트리, 선분 최소 신장 트리, 가중치 입력 선분, 연결선

Efficient Connection of Migration Routes with Their Weights Using EGOSST

Inbum Kim[†]

ABSTRACT

In this paper, a mechanism connecting all weighted migration routes with minimum cost with EGOSST is proposed. Weighted migration routes may be converted to weighted input edges considered as not only traces but also traffics or trip frequencies of moving object on communication lines, roads or railroads. Proposed mechanism can be used in more wide and practical area than mechanisms considering only moving object traces. In our experiments, edge number, maximum weight for input edges, and detail level for grid are used as input parameters. The mechanism made connection cost decrease average 1.07% and 0.43% comparing with the method using weight minimum spanning tree and weight steiner minimum tree respectively. When grid detail level is 0.1 and 0.001, while each execution time for a connecting solution increases average 97.02% and 2843.87% comparing with the method using weight minimum spanning tree, connecting cost decreases 0.86% and 1.13% respectively. This shows that by adjusting grid detail level, proposed mechanism might be well applied to the applications where designer must grant priority to reducing connecting cost or shortening execution time as well as that it can provide good solutions of connecting migration routes with weights.

Keywords : EGOSST(Edge Grade Of Services Steiner Minimum Tree), Edge Steiner Minimum Steiner Tree, Edge Minimum Spanning Tree, Weighted Input Edge, Connection Line

1. 서 론

가중치를 갖는 이동 경로들에 대하여 이들을 최소 비용으로 모두 연결하는 문제는, 가중치를 갖는 입력 선분으로 변환가능하다. 가중치 선분(Weighted Edge)은 크기나 양을 갖는 연속적으로 인접한 노드들의 집합으로 고려될 수 있다. 이것은 교통량을 고려한 이동 경로뿐만 아니라, 통신 네트

[†] 정 회 원 : 김포대학 IT학부 부교수
논문접수 : 2011년 3월 7일
수 정 일 : 1차 2011년 6월 3일
심사완료 : 2011년 6월 3일

워크에서의 대역폭, 트래픽 양을 반영한 통신선등으로 적용할 수 있으며, 동적으로 공간 또는 평면상에 위치한 특정 물체의 예측 가능한 이동궤적의 통행 양이나 빈도가 될 수 있다. 연결선(Connection Line)은 이러한 가중치 선분을 서로 연결하는 선으로, 연결하려는 선분들의 가중치 역량을 충분히 활용할 수 있도록 연결선의 가중치가 결정되어야 한다. 이러한 가중치 입력 선분(Weighed Input Edge)들의 연결은 이동 경로의 연결은 물론, 동적 라우팅, ad-hoc 네트워크, 회로 설계, 항로 결정, 도로 연결 등의 분야에 가중치와 중요도를 고려한 실제적인 방법을 제시할 수 있다. 또한 응용분야에 따라 연결 방법을 찾는 시간의 단축이 중요하거나 또는 연결비용의 절감이 상대적으로 더 중요한 요소일 수 있다. 본 논문에서는 이러한 경우를 고려하여 GOSST (Grade Of Service Steiner Minimum Tree)[1] 기법을 확대 적용한, 입력 가중치 선분들을 최소 비용으로 연결하는 방법인 EGOSST(Edge Grade Of Service Steiner Minimum Tree)를 제안한다. EGOSST에서 입력 가중치 선분은 가중치 없는 일반 노드로 변환하고, 이 노드들을 완전 연결하여 근사 스타이너 포인트들을 생성한 후, 외부 입력 파라미터인 그리드 정밀도(Grid Detail Level)에 따라 생성된 스케일된 그리드(Scaled Grid)상의 가상 노드를 활용하여 스타이너 포인트를 GOSST 포인트로 변환하여 가중치 고려 연결을 생성한다. 최종적으로 각 노드들은 입력 가중치 선분으로, GOSST 포인트들은 브리지로, 연결선은 입력 선분들에 대한 가중치 고려 연결선으로 변형한 결과물을 생성한다.

본 논문의 구성은 다음과 같다. 2장은 본 논문과 관련된 연구 내용이고, 3장은 제안하는 연결 방법에 대한 기술이다. 4장은 제안된 방법에 대한 타당성을 보이기 위해, 이를 구현하고 입력 파라미터들을 변경하면서 입력 선분을 연결하는 실험 및 분석이고 5장은 결론이다.

2. 관련 연구

본 논문의 이동 경로의 연결과 관련된 연구는 통신 경로의 연결 또는 라우팅, 통신 QoS와 관련되어 많이 시도되었다. 무선 센서 네트워크에서, 정보 제공자인 센서들의 데이터를 수집하는 싱크(Sink)들의 예측 가능한 이동성을 고려하여 소스 노드인 센서로부터 이동하는 싱크까지, 또는 이동하는 싱크들 사이의 데이터를 효과적으로 전달할 수 있는, 싱크의 예측 가능한 이동성 기반 데이터 전달 프로토콜에 대한 연구가 시도되었다[2]. 이 연구에서는 기존 연구의 싱크의 이동성이 무작위라는 전제를 탈피하여, 실제적인 응용에서 싱크의 이동 경로는 미리 결정 또는 예측될 수 있으므로, 이를 고려하여 센서와 이동 싱크 또는 이동 싱크 사이를 연결하거나 통신할 수 있다면 효과적인 센서 네트워크가 될 것이라고 주장하였다. 예측 가능한 이동성 기반 프로토콜을 사용함으로써 무선 센서 네트워크에서의 주요 문제인 센서의 에너지소비를 감소시키고, 데이터 전송률을 높이고 데이터 지연율을 감소시킨다고 주장하였다. 또한 모바일 폰

네트워크에서 격리되어있거나 자주 사용되지 않는 영역에 있는 임의의 노드가 특정 영역에 위치할 때만 데이터 전송이 일어나는 모바일 인포스테이션 네트워크(Mobile Infostaion Network) 환경에서 두 노드의 이동에 관한 연구가 있었다[3]. 이러한 환경에서, 두 노드는 특정 거리내로 근접할 때만 서로 통신하며, 노드의 이동이 상대적 위치를 변경시키기 때문에, 이 노드들은 간헐적으로 연결된다. 이 연구에서는 고속도로 모바일 인포스테이션 네트워크상에서, 각 노드는 무작위 속도로 고속도로 구간에 Poisson 비율로 진입하고, 또한 각 고속도로 구간에서 속도를 변경하는 것을 전제로 하였다. 같은 방향으로 이동하는 두 노드가 거리 내에 위치할 때, 일시적으로 서로 연결되는 순방향 트래픽(Forward Traffic)과, 두개의 노드가 반대 방향으로 이동하다가 거리 내에 위치할 때 일시적으로 서로 연결하는 역방향 트래픽(Reverse Traffic)을 고려하였다. 균일 분포의 노드 속도에 대해, 기대 연결 확률은 역방향 트래픽의 경우 관찰 노드의 속도에 독립적인 것과, 순방향 트래픽에서 관찰자의 속도가 증가함에 따라 연결 시간 비율은 증가함을 발표했다. 무선 통신 기반시설을 공유하여, 다양한 데이터 타입과 전송율의 효과적인 멀티미디어 통신의 설계를 위한 연구로 다중 경로 연결(MPC, Multiple Path Connections) 기법을 적용한, 안테나 열(Antenna Array)의 WCDMA(Wide-band Code Division Multiple Access)에 관한 연구가 발표되었다[4]. 이러한 다중 경로 연결은 무선 네트워크에서 전체 자원의 효과적 공유를 위한 것으로 분리된 여러 연결의 활용과 무선 자원의 가용성을 이용하여 호출 거절(Call Rejection)을 감소시킬 수 있다. 이 연구에서 제안된 베이스 스테이션의 4개 요소의 안테나 열(4-element antenna array)은 혼합 트래픽 시나리오에서 2.2의 용량 이득을 보였으며, 다중 경로 연결 기능을 추가한 경우, 멀티-빔 시스템에서 22~36%의 용량 개선이 가능함을 발표하였다. ATM(Asynchronous Transfer Mode), Mutiprotocol Label Switching, IntServ의 Internet Protocol 등과 같은 연결지향 패킷 스위칭 네트워크(Connection Oriented Packet Switching Networks)에서의 가상 트래픽 경로(VTP, Virtual Traffic Path)를 활용한 트래픽 컨트롤에 관한 연구가 시행되었다[5]. 이 연구에서는 기존의 flow assignment model 기반과는 다른, 비선형 목적 함수와 함께 연결지향 서비스(Connection-oriented Service)에 초점을 맞춘 확장된 flow assignment 모델을 제안하였다. 이 모델은 모든 가용 가상 트래픽 경로에 트래픽을 균형 분산시키고, 중복된 capacity를 고려한다. 또한 생성된 트래픽이 확률론적으로 변경되었을 때, 확률론적인 프로그래밍 방법(stochastic programming methodology)을 적용하여, 전체적인 네트워크 성능을 의미 있게 개선할 수 있다는 것을 보였다. 물리적 네트워크상에 완전 meshed logical 네트워크를 가상 경로 연결(VPCs, Virtual Path Connections)로 설계한 논리적 네트워크인 가상 경로 연결 네트워크(VPCN, Virtual Path Connection Network)에 관한 연구가 있었다[6]. 이 연구에서 다중 서비스 트래픽(Multi Service Traffics)의 가상 경로 연결 네트워크들의 최적 구성을 위해,

XFG로 불리는 알고리즘을 제시하였고, 이것을 표준 비선형 프로그래밍 해법과 비교하였다. 이 연구에서 XFG 알고리즘은 각 busy period 동안 최적의 VPCN을 설계 할 수 있으며, DAR(Dynamic Alternative Routing), Call Queuing과의 적용 및 성능 분석을 통해 해당 알고리즘의 우수함을 주장하였다. WMNs(Wireless Mesh Networks)에서 발생하는 일시적인 통신의 혼잡(Congestion)을 완화시키려는 연구가 있었다[7]. TCP 연결의 신뢰성을 위해 PART(Path-alternation-based throughput regulator)는 이 연구에서 제안된 대체-경로 접근 방법이다. PART에서 다중 경로의 구축을 위한 Region-established M-RMAODV 메커니즘과 연결의 혼잡함을 발견하는 Rate-based Congestion Detection, 혼잡 시 신뢰할 만한 경로의 선택 및 경로의 변경을 하는 Mesh path-alternation 메커니즘으로 구성되었다. 이 연구에서 제안된 PATR는 WMNs에서 TCP 연결의 견고함을 효과적으로 개선과 함께, loss, delay, jitter를 감소시켰다고 발표하였다.

본 논문에서 제안하는 방법인 가중치 입력 선분을 EGOSST를 이용하여 효과적으로 연결하는 것은 입력 노드를 대상으로 최소 신장 트리(Minimum Spanning Tree)와 스타이너 최소 트리(Steiner Minimum Tree)의 생성과도 밀접한 관련이 있다. 다항식 적 시간 내에서 입력 노드 및 이들 간의 연결에 대해 최적의 최소 신장 트리는 Kruskal이나 Prim의 알고리즘으로 생성가능하다[8]. 반면에 입력 노드 이외의 추가적인 가상의 노드를 생성하여 활용한다면, 최소 신장 트리보다 더 최적화된 트리인 스타이너 최소 트리를 생성할 수 있다[9]. 그러나 이러한 트리를 생성하는 것은 비-다항적 문제에 속하므로, 실제 적용 가능한 스타이너 근사 최소 트리를 얻기 위해서는 적절한 휴리스틱이 필요하다. 연속적인 두 연결선을 구성하는 세 노드로 형성된 정삼각형의 외접원을 활용하여 스타이너 포인트를 구하여 근사 스타이너 최소 트리를 구하는 방법이 대표적인 휴리스틱이다[10]. 최소 신장 트리나, 스타이너 최소 트리와는 달리, 가중치를 고려하여 트리의 생성 또는 연결비용을 최소화하려는 것이 GOSST이다[1]. 여기서 비용이란 거리와 가중치의 곱의 합을 의미한다. GOSST는 스타이너 최소 트리 문제의 확장인 비-다항적 문제이므로, 이것의 실제 응용의 적용을 위해서는 문제에 부합하는 적절한 휴리스틱이 필요하다. 일반적인 최소 신장 트리, 스타이너 최소트리, GOSST는 입력 노드들의 효과적 연결 문제이지만, 본 논문에서는 입력 노드가 아닌 이동 경로인 선분을 입력으로 하여 이들을 연결하는 효과적인 네트워크를 구성하고자 하는 것이다. 또한 이 연결은 단순히 길이만의 최적화가 아닌 각 선분의 가중치를 고려한 비용의 최적화가 목표이다. 현실 세계의 여러 응용 내에서 존재하는 가중치를 고려한 이동 경로의 효과적 연결은 가중치 선분의 연결로 변환될 수 있다. 이것은 유선 통신선이나 도로, 철도와 같은 정적인 형태, 항로, 이동 물체의 경로뿐만 아니라, 그것의 이동의 중요도나 이동 물체의 교통량, 이동 빈도 등에 대한 것도 고려되어 처리될 수 있다. 다항 적 시간 내에서 선분들을 최소 신장 트리를 이용하여 연

결하는 방법이 연구되었다[11] 이 연구에서는 가중치를 고려되지 않았고 단순히 연결 길이를 최소화하는 것이 목적이었다. 즉 입력 선분 연결 문제를 해결하기 위해, Prim의 최소 신장 트리 알고리즘을 변형하였고 또한 입력 선분 상의 임의의 위치에 존재하는 가상 중개 노드인 포탈(Portal)을 활용하여 입력 선분들을 응용 시스템의 요구에 부합하도록 연결하는 방법을 찾으려고 하였다. 이 연구에서 선분 연결 방법을 찾는 시간과 선분 연결 길이를 조절할 수 있게 하여, 응용 시스템에서 원하는 것과 부합하는 연결방법을 찾을 수 있게 하였다. J.Kim 등은 GOSST문제에 최소 스타이너 트리를 이용하여 $1+\epsilon$ 의 근사 도를 갖는 트리를 다항 적 시간 내에 얻을 수 있음을 증명하였다[12]. 센서 네트워크에서의 효율적인 라우팅을 위한 문제를 평면상의 주어진 여러 개의 노드를 최적으로 상호 연결하는 문제로 모델링하여, 이를 스타이너 트리문제로 해결하는 연구가 시도되었다[13]. 그러나 스타이너 최소 트리 문제는 대상 노드들의 상호 거리에 대한 제약이 없지만, 센서 네트워크에서는 이것과는 달리 센서들이 일정한 거리만큼 떨어져 있다는 것을 가정할 수 있으므로, 비 다항 적 문제에 속하는 스타이너 최소 트리 문제의 복잡 도를 감소시킬 수 있고, 따라서 개발된 근사 알고리즘은 적합한 실행시간을 보장하며, 적절한 최적치에 대한 비율의 근사 비율을 가짐을 발표하였다.

본 논문에서 제안하는 방법은 최소 신장 트리와 근사 스타이너 최소 트리를 이용한 가중치 선분 연결하는 방법과 비교된다. 또한 이 방법들은 제안된 EGOSST 방법의 초기 단계에서 부분적으로 사용된다. 최소 신장 트리 이용방법은 Prim의 알고리즘을 변형하였는데, Prim의 알고리즘은 다항식 적 문제 영역에서 naive 최소 신장 최적의 해를 얻을 수 있기 때문이다. 가중치 고려 최소 신장 트리 방법은 이러한 naive 최소 신장트리의 임의의 두 노드 사이의 가중치를 비교하여 최솟값을 결정하고 이 값 이상이 되는 가중치를 갖는 경로를 하나 이상 갖도록 각 노드의 가중치를 변경시켜준 것이다. 입력 노드 이외에 스타이너 포인트라는 가상적인 노드 추가하여 구성하는 naive 스타이너 최소 트리는 비-다항식 적 문제 영역에 속하여 많은 실행시간을 요구하지만, 최소 신장 트리보다 더 최적화된 결과를 얻을 수 있으므로, 문제에 적합한 휴리스틱을 개발 또는 적용을 통해서 더 바람직한 결과를 얻을 수 있다. 가중치 고려 스타이너 최소 트리 방법도 역시 naive 스타이너 트리 방법을 변형한 것으로 휴리스틱으로 생성된 근사 스타이너 최소 트리에서 임의의 두 노드를 선택하여 이들의 가중치를 비교하여 최솟값을 경로 가중치로 결정하고, 이 값 이상이 되는 가중치를 갖는 경로가 하나 이상 존재하도록 각 노드의 가중치를 변경시켜준 것이다. 이들 방법들은 일반적인 노드들에 대한 방법이므로, 본 논문에서 다루는 이동 경로의 연결, 즉 입력 선분의 연결 문제에 적용하기 위해서는 제안하는 방법의 초기 단계에서 입력 선분을 입력 노드로 변환해야 하고, 최종 결과를 제출하기 전에 생성된 연결체에서의 노드를 입력 선분으로 변환하고 최적화하는 과정이 추가되었다.

3. 본 론

<표 1>에는 본 논문에서 제안하는 EGOSST를 이용한 가중치 입력 선분을 연결하여 이동 경로의 연결을 생성하는 과정이 표현되어있다. 먼저, 가중치 입력 선분을 크기를 갖는 노드로 변환한 후, 이들에 대한 완전 연결선을 생성한다. 노드와 생성된 연결선에 대하여 Prim의 알고리즘을 응용한 최소 신장 트리 T_{nSP} 를 구성한다[8]. 최소 신장 트리 T_{nSP} 에 존재하는 모든 연속적인 두 연결선에 대하여, 이들을 구성하는 각각의 세 개의 노드를 활용하는 휴리스틱을 이용하여 스타이너 포인트를 생성하고 이를 세 개의 노드들과 연결하는 스타이너 그래프 G_{nST} 를 생성한다[10]. G_{nST} 에 존재하는 노드와 연결선들을 대상으로 하여 Prim의 알고리즘을 재실행시킨 최소 신장 트리 T_{gnSP} 를 생성한 후에, 필요 없는 연결선을 제거하여 근사 스타이너 최소 트리 T_{nST} 를 생성한다. T_{nST} 에서 존재하는 임의의 한 쌍인 두 노드의 가중치의 최솟값을 조사하여 두 노드 사이에 존재하는 경로를 구성하는 노드의 가중치를 비교, 변경하여, 해당 연결선의 가중치를 결정된 새로운 트리인 T_{nSTW} 을 생성한다. T_{nSTW} 에 존재하는 모든 스타이너 포인트들 중의 임의의 한 포인트인 P_{nST} 의 생성에 관여되는 노드 a, b, c로 구성된 영역을 외부 입력 파라미터인 그리드 정밀도로 스케일된 그리드(Scaled Grid)를 생성하고, 이 그리드 상의 모든 가상 노드들을 대상으로 a, b, c와의 연결비용이 최소가 되는 GOSST 포인트 P_{GOSST} 를 탐색한다. T_{nSTW} 의 모든 스타이너 포인트들에 대하여 GOSST 포인트가 결정된 노드 GOSST인 T_{nGOSST} 를 생성한다. T_{nGOSST} 에서 각 노드는 초기의 입력 선분으로, 연

결선은 가중치가 표시된 연결선으로 변환하고, GOSST 포인트와 입력 선분과의 연결, 또는 입력 선분과 다른 입력 선분의 연결선이 최단길이가 되는 작업을 하여 이동 경로의 가중치 반영 연결의 최종 결과물인 T_{EGOSST} 를 완성하게 된다. <표 1>의 실행시간을 분석하면 다음과 같다. 단계 1에서 노드로 변환된 전체 입력 선분의 수를 E 로 했을 때, 유클리드 최소 신장트리를 피보나치 힙(Fibonacci heap)으로 구현한 Prim의 최소 신장 생성 알고리즘을 이용하면 $O(E^2 + E \log E) = O(E^2)$ 시간 내에, 입력 선분을 노드로 변환한 최소 신장 트리 T_{nSP} 의 생성 가능하다. 단계 2에서 T_{nSP} 는 $E-1$ 개의 연결선으로 구성되어 있으므로, 스타이너 포인트 P_{nST} 들을 구하기 위해서는 $E-1 C_2 = O(E^2)$ 의 시간이 필요하며, 생성된 스타이너 그래프 G_{nST} 에는 $O(E^2)$ 개의 스타이너 포인트 P_{nST} 가 존재하게 된다. 단계 3에서 입력되는 노드의 수는 $O(E^2)$ 이므로, 이것에 대한 최소 신장 트리 T_{gnSP} 는 $O(E^2 \log E^2) = O(E^2 \log E)$ 시간 내에 생성된다. 단계 4의 T_{gnSP} 에서 연결 차수가 1인 스타이너 포인트 P_{nST1} 를 찾아 처리하기 위해서 $O(E^2)$ 개의 P_{nST} 를 탐색해야 하므로, T_{nST} 를 생성하기 위해서 $O(E^2)$ 시간이 필요하고, 단계 5의 T_{nST} 의 임의의 두 노드 사이의 경로 가중치를 조사하고, 적절히 조정하여 가중치가 반영된 트리 T_{nSTW} 를 생성하기 위해서는 $O(E^2)$ 의 시간이 필요하다. 단계 6에서 한 개의 스타이너 포인트 P_{nST} 를 GOSST 포인트 P_{GOSST} 로 대체하기 위해서는, $X_y \times Y_y$ 그리드 상에 위치한 가상노드, 즉 후보 GOSST 포인트들을 탐색해야 한다. 즉 $O(X_y \times Y_y) = O(\frac{D_x}{P_s} \times \frac{D_y}{P_s})$ 개의 후보 GOSST 포인트들 중에

<표 1> EGOSST를 이용한 가중치 반영 이동 경로의 연결 알고리즘

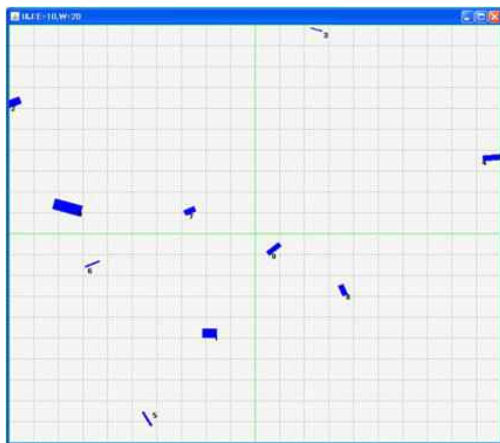
단계	내 용
1	입력 선분은 노드로, 입력 가중치는 해당 노드의 크기로 변환하고, 노드들을 모두 연결하는 완전 연결선을 생성한 후, 노드와 연결선으로 최소 신장 트리 T_{nSP} 를 생성한다[8].
2	T_{nSP} 에 존재하는 연속적인 두 연결선 e_1, e_2 를 구성하는 세 노드 a, b, c에 대하여 적절한 휴리스틱[10]을 실행하여 스타이너 포인트 P_{nST} 를 생성하고, a, b, c와 P_{nST} 를 연결한다. 모든 연속 연결선에 대하여 스타이너 포인트가 생성될 때까지 단계 2를 반복하여 스타이너 그래프 G_{nST} 를 생성한다.
3	G_{nST} 에서 스타이너 포인트 P_{nST} 들을 포함하는 모든 노드들과 연결선에 대한 새로운 최소 신장 트리 T_{gnSP} 를 생성한다.
4	T_{gnSP} 에서 스타이너 포인트 P_{nST} 들을 탐색하여, 연결 차수가 1인 스타이너 포인트 P_{nST1} 을 찾고, 이것의 연결선과 P_{nST1} 를 모두 제거한다. 연결 차수가 1인 스타이너 포인트가 T_{gnSP} 에 존재하지 않을 때까지 단계 4를 반복하여 T_{nST} 를 생성한다.
5	T_{nST} 에 존재하는 두 노드 X, Y의 가중치 W_x, W_y 에 대하여, X, Y사이의 경로의 가중치가 $\min(W_x, W_y)$ 인 경로가 최소 1개 이상 존재하도록, X와 Y사이의 존재하는 모든 연결선의 가중치를 결정한다. T_{nST} 에 존재하는 모든 두 개의 노드들에 대하여 단계 5를 반복하여 T_{nSTW} 를 생성한다.
6	T_{nSTW} 에 존재하는 한 개의 스타이너 포인트 P_{nST} 에 대하여, 이 포인트를 생성하는데 관여한 노드 a, b, c로 생성되는 영역에 대하여 입력 파라미터인 그리드 정밀도로 스케일된 그리드를 생성하고 이 그리드 상에 존재하는 모든 노드 P_g 중에서 a, b, c와의 연결비용이 최소가 되는 GOSST 포인트인 P_{GOSST} 를 탐색한 후, P_{GOSST} 와 P_{nST} 를 대체하고, P_{nST} 와 관련된 연결선 또한 P_{GOSST} 의 연결선으로 대체한다. T_{nSTW} 에 존재하는 모든 스타이너 포인트들에 대하여 단계 6을 반복하여 T_{nGOSST} 를 생성한다.
7	T_{nGOSST} 에서, GOSST 포인트인 P_{GOSST} 는 GOSST 연결지점으로, 그 밖의 노드들을 원래의 입력 선분으로 변환한다. 노드 간의 연결선을 입력 선분간의 연결로 변환한다. 또한 GOSST 포인트인 P_{GOSST} 와 노드간의 연결선은 GOSST 연결지점을 노드가 대표하는 입력 선분과 가중치를 반영한 최단 연결하고, 그 밖의 두 노드 사이의 연결선은 이들이 대표하는 두 입력 선분을 가중치를 반영하여 최소비용으로 연결한다. T_{nGOSST} 에 존재하는 모든 노드들과 연결선에 대하여 입력 선분과 가중치를 반영한 연결선으로 변환하여 T_{EGOSST} 를 생성한다.
8	단계 7에서 생성된 T_{EGOSST} 를 이동 경로의 가중치 반영 연결의 최종 결과로 출력한다.

서, P_{nST} 생성과 관련된 a, b, c와의 연결 비용이 최소가 되는 것을 최종 GOSST 포인트 P_{GOSST} 로 결정하게 된다. 여기서 D_x, D_y 는 적용하려는 문제를 2차원 평면 영역으로 변환했을 때의 X축, Y축의 길이이고, P_s 는 외부 입력인 그리드 정밀도이다. 이러한 작업을 스타이너 포인트의 수인 $O(E^2)$ 만큼 해야 하므로 단계 6의 실행시간은 $O(\frac{D_x}{P_s} \times \frac{D_y}{P_s}) \times O(E^2)$ 이다. $O(\frac{D_x}{P_s} \times \frac{D_y}{P_s})$ 는 그리드 정밀도인 P_s 에 따라 조절가능한 유한한 값이므로, $O(\frac{D_x}{P_s} \times \frac{D_y}{P_s}) \times O(E^2) = O(E^2)$ 이다. 단계 7에서 T_{nGOSST} 에 존재하는 $O(E^2)$ 의 노드와 $O(E^2)$ 개의 연결선에 대한 변환 및 처리 작업을 시행하여 $O(E^2)$ 의 시간 후에 최종 T_{EGOSST} 가 생성된다. 따라서 본 논문에서 제안하는 EGOSST 방법에 의한 경로의 가중치 연결방법을 찾아내기 위한 전체 실행시간은 $O(E^2 \log E)$ 이 된다.

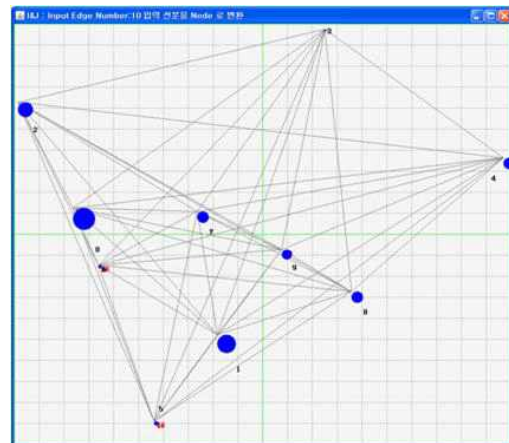
(그림 1)은 최대 가중치가 20인 10개의 입력 선분이 나타나 있다. 이 입력 선분은 가중치가 반영된 이동 경로이다. 각 입력 선분은 자체의 가중치를 가지고 있는데, 이것은 선분의 굵기로 표현되어 있다. 이러한 입력 선분의 길이는 이동 경로의 궤적 길이로, 입력 선분의 굵기는 이동 경로의 중요도 또는 이동 량으로 변환될 수 있다. (그림 1)의 입력 정보에서 가중치는 20가지 단계로 부여해야 하는 환경, 즉 최대 가중치가 20인 입력을 나타낸다. (그림 2)는 (그림 1)의 입력정보를 노드 정보로 변환하여 각 입력 선분의 중점 위치에 노드의 위치를, 가중치는 노드의 크기로 변환한 것을 보인다. (그림 1)에서 입력 선분들 간의 연결 정보가 없으므로, (그림 2)에서 입력 선분이 변환된 입력 노드들은 유클리드 트리를 구하기 위해 완전 연결선을 생성한다. 이 연결선은 가중치를 반영하지 않은 순수한 연결 정보만을 의미한다. (그림 2)의 노드와 연결선에 대하여 최소 신장 트리를 생성한 후, 연결선에 가중치를 적절히 배분하고, 노드와 연결선을 입력 선분과 이것들의 연결선으로 변환하여 생성된

가중치 최소 신장트리의 결과가 (그림 3)에 있다. 마찬가지로 (그림 2)의 노드와 연결선에 대하여, 근사 스타이너 최소 트리를 생성한 후, 연결선에 가중치를 적절히 배분하고, 노드와 연결선을 입력 선분과 이것들의 연결선으로 변환하여 생성된 가중치 스타이너 최소 트리의 결과가 (그림 4)에 있다. (그림 5)는 GOSST 포인트 탐색을 위한 그리드 정밀도를 0.001로 하여, 본 논문에서 제안하는 EGOSST 방법에 의해, 가중치 입력 선분을 연결하는 방법을 표시하는데, 이것의 연결비용은 421.9342이다. 이것은 (그림 3)의 가중치 최소 신장 트리를 이용한 방법과 (그림 4)의 가중치 스타이너 최소 트리를 이용한 방법보다 각각 9.68%, 3.63% 절감된 것이다. (그림 6)은 (그림 4)의 스타이너 최소 트리에서의 생성되는 스타이너 포인트인 s1, s2, s3, 그리고 (그림 5)의 EGOSST에서 생성된 GOSST 포인트인 g1, g2, g3의 위치를 문제 영역에서 비교한 것이다. 본 논문에서 제안하는 EGOSST는 스타이너 포인트를 입력 선분들의 가중치의 값에 따라 최소 연결 비용을 얻을 수 있는, 외부 파라미터로 스케일된 그리드 상의 위치로 이동하여 GOSST 포인트를 결정된다.

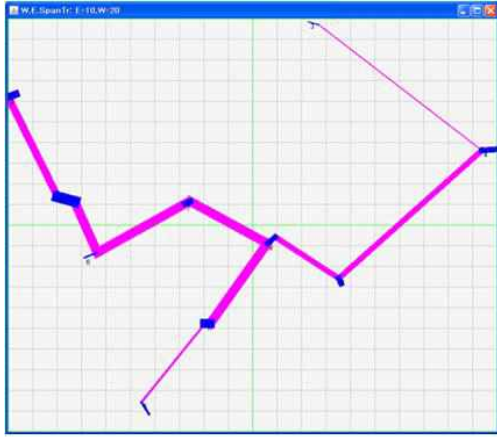
(그림 7)과 (그림 8)은 입력 선분의 수가 10이고 각 선분의 최대 가중치가 5, 즉 가중치의 종류가 5개인 환경에서, GOSST 포인트를 찾기 위한 그리드 정밀도를 0.1과 0.001로 하여 생성된 EGOSST인데 정밀도가 0.001인 경우는 0.1인 경우에 비해, 0.0138인 약 0.008%인 의 비용절감을 얻는다. (그림 9)과 (그림 10)은 입력 선분의 수가 10이고 각 선분의 최대 가중치가 20, 즉 가중치의 종류가 20개인 환경에서, 그리드 정밀도를 0.1과 0.001로 하여 생성된 EGOSST인데 정밀도가 0.001인 경우는 0.1인 경우에 비해, 약 0.006%인 0.0253의 비용절감을 얻는다. (그림 11)은 (그림 7), (그림 8), (그림 9), (그림 10)의 각기 다른 조건에 의해 생성된 EGOSST들의 GOSST 포인트들의 위치 정보를 보이고 있다. 이것에 대한 구체적인 GOSST 포인트들의 좌표 정보와 연결 비용이 <표 2>에 나타나 있다.



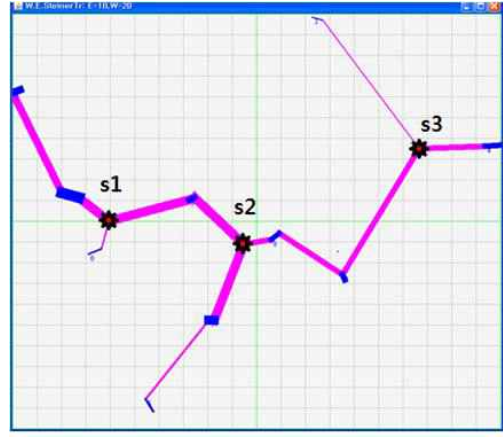
(그림 1) 최대 가중치가 20인 10개의 가중치 입력 선분



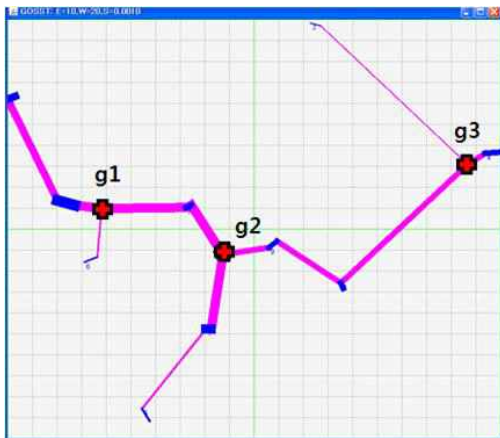
(그림 2) 가중치 입력 선분의 노드 변환 및 완전 연결 생성



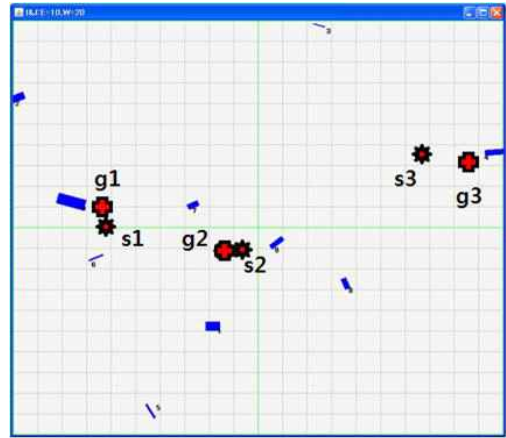
(그림 3) 최소 신장 트리를 이용한 가중치 입력 선분 연결 (비용=467.1679)



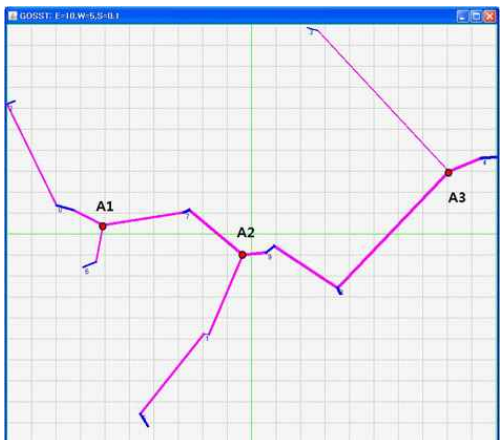
(그림 4) 근사 스타이너 최소 트리를 이용한 가중치 입력 선분 연결 (비용=437.8393)



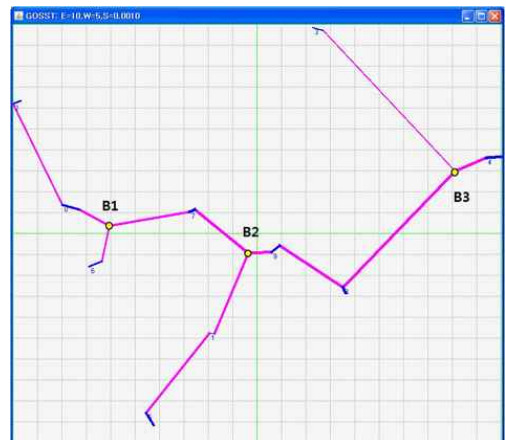
(그림 5) EGOSST를 이용한 가중치 입력 선분 연결(비용=421.9342)



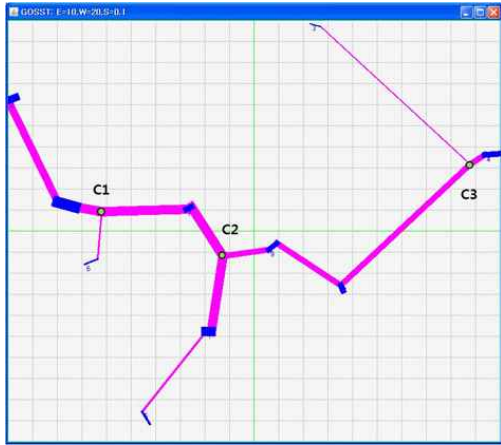
(그림 6) 스타이너 포인트와 GOSST 포인트의 위치 비교



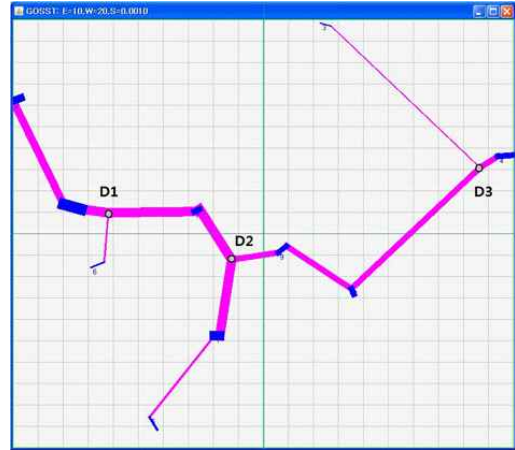
(그림 7) 입력 선분 10, 최대 가중치 5, 그리드 정밀도 0.1 환경에서의 EGOSST 생성 결과 (비용=171.7382)



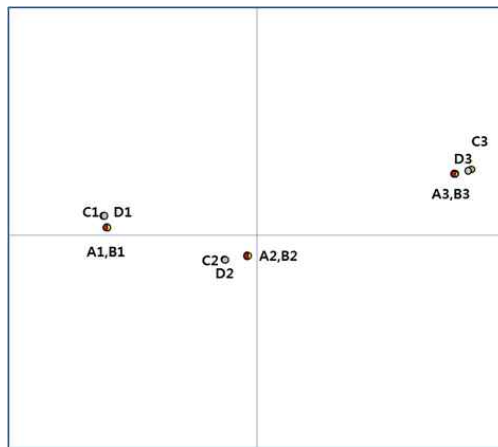
(그림 8) 입력 선분 10, 최대 가중치 5, 그리드 정밀도 0.001 환경에서의 EGOSST 생성 결과 (비용=171.7244)



(그림 9) 입력 선분 10, 최대 가중치 20, 그리드 정밀도 0.1 환경에서의 EGOSST 생성 결과 (비용=421.9595)



(그림 10) 입력 선분 10, 최대 가중치 20, 그리드 정밀도 0.001 환경에서의 EGOSST 생성 결과 (비용=421.9342)



(그림 11) 각 조건에 따라 생성된 연결 지점의 위치 비교

<표 2> 최대 가중치 및 그리드 정밀도에 따른 GOSST 포인트 좌표 및 입력 선분 연결 비용

조건 \ GOSST 포인트(i)	최대 가중치:5 정밀도:0.1 A _i	최대 가중치:5 정밀도:0.001 B _i	최대 가중치:20 정밀도:0.1 C _i	최대 가중치:20 정밀도:0.001 D _i
1	(-6.0700,0.4000)	(-6.0360,0.3450)	(-6.1700,0.9000)	(-6.1820,0.9470)
2	(-0.3400,-1.0000)	(-0.3360,-0.9510)	(-1.2400,-1.2000)	(-1.2420,-1.2300)
3	(8.0900,2.9500)	(8.1320,2.9470)	(8.7900,3.1500)	(8.6780,3.0700)
연결 비용	171.7382	171.7244	421.9595	421.9342

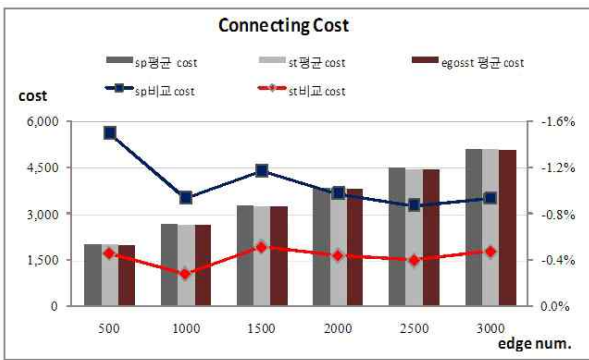
4. 실험 및 결과 분석

본 연구를 위해 사용된 실험 인자는 입력 선분의 수, 각 입력 선분의 최대 가중치, 그리고 최적의 GOSST 포인트를 찾기 위한 그리드 정밀도(grid detail level)이다. 관찰 결과는 입력 가중치 선분들의 연결 비용과 실행 시간이다. 연결

비용과 실행 시간은 최소 신장 트리와 스타이너 최소 트리를 이용한 결과와 비교된다. 실험을 위해 무작위로 생성된 가중 입력 선분의 수는 500, 1000, 1500, 2000, 2500, 3000개이다. 각 선분의 최대 가중치는 5, 10, 15, 20으로 하여 이 범위 내에서 입력 선분들의 가중치를 무작위로 설정하였다. 또한 그리드 정밀도는 0.001, 0.00316, 0.01, 0.03162, 0.1로 이

것의 상용로그 값은 -3, -2.5, -2, -1.5, -1이다. 각 입력 선분들의 위치는 2차원 평면상에서, -10.0과 10.0 사이의 x, y 좌표 값을 무작위로 선택하여 이를 선분의 시작점으로 하고, 이를 중심으로 하여 0과 360도 사이의 각도를 가지는, 0.2와 0.8 사이의 거리에 위치하는 노드를 선분의 끝점으로 하여 시작점과 연결하여 생성하였다. 각 입력 선분의 가중치는 1과 최대 가중치 사이의 임의의 정수로 할당된다. 실험 환경 컴퓨터는 Intel 프로세서와 4 Giga 램이 장착되었고, 비교 방법인 가중치 최소 신장 트리 방법, 가중치 스타이너 최소 트리 방법, 그리고 제안된 EGOSST 방법은 윈도우즈 환경에서 java로 구현되었다.

4.1 입력 가중치 선분의 수



(a) 연결 비용



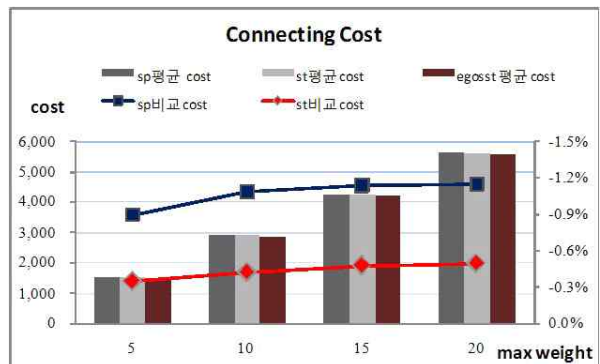
(b) 실행 시간

(그림 12) 입력선분 수의 변화 실험 결과

(그림 12)는 실험 인자인 입력 선분의 수를 변경하면서 본 논문에서 제안하는 방법에 의해 생성된 입력 가중치 선분의 연결 비용 및 실행시간에 대한 분석 결과이다. (그림 12)의 (a)에서 확인할 수 있는 것처럼 입력 가중치 선분의 수가 커질수록 세 방법의 연결 비용은 증가하지만 제안된 방법을 이용한 연결은 최소 신장 트리와 스타이너 최소 트리를 이용한 연결 방법에 비해 비용은 감소된다. 입력 선분의 크기와 최소 신장 트리와 스타이너 최소 트리 방법과 비교한 연결 비용 절감비율에 대해서는 큰 관계를 보이지 않는 데, 최소 신장 트리와 비교할 때는 연결 선분의 수가 500일

때 최대 인 1.51%, 스타이너 최소 트리와 비교해서는 최대 0.52에 대한 상대적인 절감비율을 보인다. (그림 12)의 (b)에서 보이는 것처럼, 입력 가중치 선분의 수가 커질수록 실행 시간은 증가하지만 제안된 방법을 이용한 연결은 최소 신장 트리와 스타이너 최소 트리를 이용한 연결 방법에 비해 실행 시간의 증가율은 점차 감소된다. 최소 신장 트리와 비교할 때, 연결 선분의 수가 500일 때 시간의 증가율은 최대인 2590.53%, 연결 선분의 수가 3000일 때 최소인 188.92%이고, 스타이너 최소 트리 방법과 비교해서는 연결 선분의 수가 500일 때 시간의 증가율은 최대인 1736.99%, 연결 선분의 수가 3000일 때 최소인 142.89%이다. 실행 시간의 증가율이 입력 선분의 크기가 작을 때는 GOSST 포인트를 찾기 위한 빈 공간이 많아지므로 최소 신장 트리나 스타이너 최소 트리 방법에 비해 실행시간의 증가율이 크지만, 입력 선분의 수가 많을 때는 GOSST 포인트를 찾기 위한 빈 공간이 상대적으로 적으므로 실행시간의 증가율이 적어진다. 또한 입력 선분의 수가 2000 이상인 경우에는 입력 선분의 수가 커지더라도 그 증가율의 변화가 매우 작아짐을 확인할 수 있다.

4.2 입력 선분의 최대 가중치



(a) 연결 비용



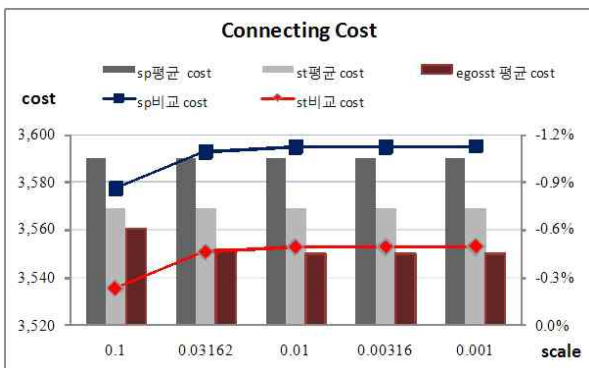
(b) 실행 시간

(그림 13) 입력선분 최대 가중치의 변화 실험 결과

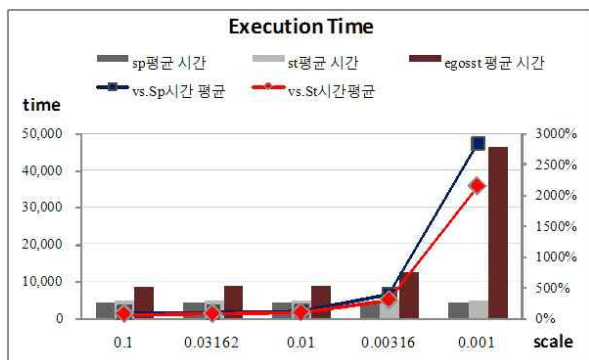
(그림 13)은 실험 인자중의 하나인 입력 선분의 최대 가중치를 변경하면서 본 논문에서 제안하는 방법에 의해 생성

된 입력 가중치 선분의 연결선 비용 및 실행시간에 대한 분석 결과이다. 각 입력 선분의 가중치는 1과 이 최대 가중치 사이의 임의의 정수 값이다. 각 선분의 가중치는 통신선의 대역폭이나 트래픽 양, 도로망의 도로 차선 수 등으로 해석 가능하다. (그림 13)의 (a)에서 확인할 수 있는 것처럼 입력 최대 가중치가 커질수록 최소 신장 트리방법이나, 최소 스타이너 방법에 비해 제안된 방법은 비용 절감율이 조금씩 높아짐을 확인할 수 있다. 입력 선분의 최대 가중치가 5인 경우 최소 신장 트리 방법과 스타이너 최소 트리 방법에 비해 각각 0.90%, 0.35% 연결 비용이 감소되었으나, 최대 가중치가 20인 경우에는 각각 1.15%, 0.49% 감소되었다. (그림 13)의 (b)에서 보이는 것처럼, 입력 선분의 최대 가중치와 실행 시간과는 관계가 없는데, 최소 신장 트리 방법과 스타이너 최소 트리 방법과 비교하여 각각 713%, 535%에 수렴하게 증가한다. 이는 입력 선분의 최대 가중치는 연결 비용에 영향을 미치지만, 실행 시간에 영향을 주는 요인이 아님을 보여 주고 있다.

4.3 GOSST 포인트 검색 정밀도



(a) 연결 비용



(b) 실행 시간

(그림 14) 그리드 정밀도의 변화 실험 결과

(그림 14)는 실험 인자인 GOSST 포인트 검색을 위한 그리드 정밀도를 변경하면서 본 논문에서 제안하는 방법에 의해 생성된 연결 구조체의 가중치 선분의 연결 비용 및 실행 시간에 대한 분석 결과이다. 정밀도가 높으면 높을수록 최

적의 위치에 근접할 수 있으므로 (그림 14)의 (a)와 같이 전체적인 연결비용은 예상대로 감소될 것이다. 또한 최소 신장 트리 방법과 스타이너 최소 트리 방법에 비해서 그 절감율이 커질 것이다. 정밀도가 0.1인 경우에는 최소 신장 트리 방법과 스타이너 최소 트리 방법에 비해 연결 비용의 절감율이 최소인 각각 0.86%와 0.23%이었으나, 0.001인 경우에는 절감율이 최대인 1.13%와 0.50%이었다. 그러나 그리드 정밀도가 0.03162 부터는 그 변화율이 차이가 크지 않음을 확인할 수 있다. (그림 14)의 (b)의 실행 시간에 관한 결과는 정밀도가 높을수록 처리해야 대상이 늘어나므로 실행 시간이 증가함을 확인할 수 있다. 정밀도가 0.1인 경우에 최소 신장 트리와 스타이너 최소 트리와 비교하여 실행시간의 증가율은 각각 97.02%, 69.30%이었으나 0.001인 경우에는 급속히 증가되어 각각 2843.87%, 2147.63%이었다. 그러나 정밀도가 0.01보다 작은 경우에는 실행시간의 증가율의 변화가 그리 심하지 않음을 확인할 수 있는데, 이는 처리 대상의 증가가 다른 처리 과정의 소요 시간에 비해 그리 크지 않기 때문으로 판단되어진다. 따라서 응용에 따라 비용 감소율과 시간 증가율에 대한 요구에 따라 적절한 검색 정밀도를 선택한다면 좋은 성능을 기대할 수 있을 것이다.

5. 결론

본 논문에서는 가중치를 갖는 여러 이동 경로들에 대하여 이들을 최소 비용으로 모두 연결하는 문제에 대하여, 가중치 경로를 가중치를 갖는 입력 선분으로 변환하고, 이들 입력 선분들에 대해서, GOSST 포인트를 적절히 생성, 활용하여 가중치를 고려한 최소 비용으로 연결하는 방법인 EGOSSST 방법을 제안한다. 본 논문에서 다루는 가중치 입력 선분은 물체의 이동 경로뿐만 아니라, 통신 네트워크에서의 대역폭과 트래픽을 고려한 통신선, 동적 객체의 트래픽, 교통량을 고려한 도로나 철도, 항로 등으로 확대 적용할 수 있다. 이와 같은 입력 가중치 선분 연결 문제는 단순히 위치와 길이만 고려하는 선분 연결 문제에 비해서, 네트워크에서의 라우팅, ad-hoc 네트워크, 회로 설계, 항로 결정, 도로 연결 등에 좀 더 현실적으로 적용될 수 있다. 제안된 EGOSSST 방법은 비교 대상인 가중치 최소 신장 트리와 가중치 스타이너 최소 트리를 이용한 연결 방법에 비해 실행 시간은 증가시켰지만, 그 연결 비용을 감소시켰다. 최소 신장 트리를 이용하여 입력 가중치 선분을 연결을 생성하는 시간은 평균 713.30% 증가시켰으나, 연결 비용 평균 1.07% 감소시켰고, 또 다른 비교 대상인 스타이너 최소 트리를 이용한 방법 보다는 실행 시간이 평균 535.29% 증가되었으나, 연결 비용은 평균 0.43% 감소시켰다. 그러나 외부 파라미터인 GOSST 포인트 탐색을 위한 그리드 정밀도를 0.1로 설정할 경우에는 최소 신장 트리 방법에 비해 연결비용이 0.86% 감소되었지만 실행시간은 97.02% 증가함에 그친다. 이는 본 논문에서 제안하는 방법이 외부 입력 파라미터인 그리드 정밀도를 조절함으로써, 연결 비용의 절감 또는 실행시

간의 단축에 초점을 두고 특정 응용분야에 적용할 수 있음을 보인다.

향 후 연구는, 매우 많은 입력 가중치 선분들을 대상으로 연결하고자 할 때, 그리드 정밀도를 조절하여 실행시간을 줄이는 방법이 아닌 다른 신속하게 연결 방법을 찾을 수 있는 다항 적 시간 근사구조인 PTAS(Polynomial Time Approximation Scheme)의 적용에 관한 것이다[12]. 이것은 비-다항식 적 문제와 같이, 현실적으로 계산하기 힘든 문제를 세분화시켜 각각 Brute-force 방법으로 해결한 후, 이를 bottom-up 방식으로 통합하여 해를 찾는다. 이 연구는 현실 세계에서 실제로 당면할 수 있는 매우 많은 이동 경로의 연결방법과 관련된 문제의 해결 방법을 신속히 찾을 수 있는 개선된 방법을 제안할 수 있을 것이다.

참 고 문 헌

[1] G.L. Xue, G.H. Lin. and D.Z. Du, "Grade of Service Steiner Minimum Trees in Euclidean Plane", *Algorithmica*, Vol.31, pp.479-500, 2001.

[2] 박수창, 이의신, 진민숙, 최영환, 김상하, "무선 센서 네트워크에서 예측 가능한 싱크 이동성을 기반으로 한 데이터 전달 프로토콜", *한국정보과학회논문지*, Vol.14, No.3, pp.276-280, 2008.

[3] Wing Ho Yuen, Roy D. Yates, Chi Wan Sung, "Effect of Node Mobility on Highway Mobile Infostation networks", *MSWIM '03 Proceedings of the 6th ACM international workshop on Modeling analysis and simulation of wireless and mobile systems*, San Diego, USA, pp.82-91, September, 2003.

[4] D. Bunnjaveht, M. A. Wickert, "Capacity enhancement of a multi-beam technique in a WCDMA network using multiple path connections", *Mobility '06 Proceedings of the 3rd international conference on Mobile technology, applications & systems*, Article No.37, Bangkok, Thailand, 2006.

[5] Dongfang Zheng, Xian Liu, Mrinal Mandal, Weidong Lu, "Virtual Traffic Path Optimization in Connection-Oriented Networks with Stochastic Traffic", *Journal of Network and Systems Management*, Vol.12, No.2, pp.231-249, 2004.

[6] A. Arvidsson, S. A. Berezner, A. E. Krzesinski, "The Design and Management of ATM Virtual Path Connection Networks", *MASCOTS '99 Proceedings of the 7th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, College Park, USA, pp.2-9, 1999.

[7] Yi-Ting Chien, Mong-Fong Horng, Yu-Cheng Sheu, Jeng-Pong Hsu, Yau-Hwang, Kuo, Chin-Shiun Shieh, "An Alternative-Path Approach to Improve Robustness of TCP Connections in Wireless Mesh Networks", *2010 Sixth International Conference on Intelligent Information Hiding and Multimedia Signal*, Darmstadt, Germany, pp.248-251, 2010.

[8] T.H. Cormen, C.E. Leiserson, R.L. Rivest and C. Stein, *Introduction to Algorithm*, 2ndEd., MIT Press, pp.561-579, 2001.

[9] http://en.wikipedia.org/wiki/Steiner_tree, December, 2009.

[10] B. Bell, "Steiner Minimal Tree Problem", <http://www.css.taylor.edu/~bbell/steiner/>, January 1999.

[11] 김인범, 김수인, "선분상의 포탈을 이용한 근사 선분 최소 신장 트리의 생성", *정보처리학회 논문지*, 12월, 2009

[12] J. Kim, M. Cardei, I. Cardei and X. Jia, "A Polynomial Time Approximation Scheme for the Grade of Services Steiner Minimum Tree Problem", *Journal of Global Optimization* Vol.24, pp.437-448, 2002.

[13] 김준모, "센서 네트워크 구축에서의 Combinatorial 기법 적용", *대한전자공학회논문지TC*, Vol.45, No.7, pp.9-16, 2008.



김 인 범

e-mail : ibkim@kimpo.ac.kr

1989년 서울대학교 컴퓨터공학과(학사)

1991년 서울대학교 컴퓨터공학과(공학석사)

2007년 위스컨신주립대-밀워키 전산학

(공학박사)

1996년~현 재 김포대학 IT학부 부교수

관심분야 : 네트워크 알고리즘, 데이터베이스, 컴퓨터 이론