

속도 모델을 이용한 차량 궤적의 효율적인 표현 방법

양해정¹, 김태완², 이기준¹

¹ 부산대학교 컴퓨터 공학과 시공간데이터베이스 연구실

² 부산대학교 컴퓨터 및 정보통신 연구소

{hjang, twkim, lik}@isel.cs.pusan.ac.kr

Efficient Representation of Vehicle Trajectories by Velocity Model

Hye-Jung Stephan Yang⁰, Tae-Wan Kim, and Ki-Joune Li

¹Dept. of Computer Engineering, Pusan National University

²Research Institute of Computer Information and Communication

요 약

이동 객체의 위치 정보는 현재 위치뿐만이 아니라 과거 움직였던 궤적 데이터 역시 데이터마이닝과 같은 응용분야에서 중요하다. 대부분의 기존 연구에서 사용하는 GPS(Global Positioning Service) 위치정보는 이 차원 유클리디안 공간의 한 점과 시간만으로 표현된다. 우리는 이러한 표현이 가지는 내재적인 문제점들을 지적하고 이를 해결할 수 있는 새로운 방법을 제시한다. 본 논문에서 우리는 이동 객체의 움직임은 기본속도모델을 따라 움직인다고 가정하고 이를 이용하여 다양한 움직임의 형태를 표현하고 이러한 표현에 의하여 달성할 수 있는 저장 및 처리의 효율성에 대하여 연구한다. 실제 데이터를 이용한 분석에서 우리는 이동 객체가 우리가 제시하는 속도 모델에 따라 움직인다는 사실을 보여주고, 제시하는 표현 방법이 저장뿐만이 아니라 성능적인 면에서도 기존 GPS 위치정보 표현보다 월등하다는 사실을 보여 준다.

1. 서론

최근 텔레매틱스의 발전으로 위치 정보를 이용한 여러 가지 서비스가 사용자에게 제공 되고 있다. 텔레매틱스는 인공 위성을 이용한 GPS(Global Positioning Service)와 GIS(Geographic Information System)을 자동차에 접목하여 휴대폰이나 PDA등을 통해 운전자에게 여러 가지 서비스를 제공해 주는 종합 정보 시스템이다. 텔레매틱스 서비스에는 실시간 교통 및 생활 정보 제공이나 운전 경로 안내 그리고 차량 사고 알림이나 게임 등이 있다. 이는 대부분GPS 위성을 이용하여 사용자의 위치를 측정하여 그것을 기반으로 이루어지고 있다. 현재 GPS 데이터를 이용하여 이동객체의 현재 위치를 추적하거나 효율적으로 저장 및 관리 하는 방법에 대해서는 많은 연구가 수행되어 왔다. 이동 객체의 현재 위치만을 저장 및 관리하는 경우 이전에 움직인 궤적 정보 등을 추출하기 어렵다. 따라서 데이터마이닝과 같은 응용분야를 위해서는 과거 위치에 대한 정보의 저장이 필요하다. 이동 객체의 위치를 저장하려고 하는 경우 매 시간 단위마다 위치 정보를 저장해야 하기 때문에 저장되는 정보의 양은 아주 방대하다.

따라서 이러한 정보를 간결하게 표현하고 효율적으로 처리하는 방법에 대한 연구가 현실적으로 필요하나, 이러한 분야에 대한 연구는 아직 많이 이루어 지지 않았다. 본 논문에서 우리는 이동 객체의 속도 모델을 제안하고, 이를 이용하여 이동 객체의 과거 위치 정보를 효율적으로 처리하는 방법을 제시하고자 한다. 대부분의 연구에서 이동객체의 위치 정보는 GPS에서 제공하는 데이터를 가공하지 않고 사용하고 있다. GPS 데이터는 간략하게 (x, y, t) 로 표현된다. 이 때, 변수 (x, y) 는 정규화된 데이터 공간에서 상대 위치 좌표를 나타내고, 변수 t 는 특정 시간범위에서의 시간을 나타낸다. 이동 객체의 과거 위치 분석을 위해 GPS 데이터를 그대로 이용할 경우 다음과 같은 문제점들이 발생할 수 있다.

첫째, GPS 데이터로는 이동 객체의 정확한 과거 궤적을 알 수 없다. GPS 데이터로 과거 이동 객체의 궤적을 표현한다면 (x, y, t) 의 집합으로 나타낼 수 있다. 이동 객체는 도로를 기반으로 움직이기 때문에 이동 객체의 궤적은 시간에 따라 이동 객체가 지나간 도로들의 집합이 된다. 그런데 GPS 데이터는 시간에 따른 점 데이터의 집합이므로 이동 객체가 어떠한 도로를 지나갔는지에 대한 정확한 정보를 얻기 어렵다. 둘째, 이동 객체의 궤적 데이터는 시간에 따라 누적되므로,

* 본 연구는 사업지원부의 지역혁신 인력양성사업 및 한국 과학 재단의 M02-2004-000-20782-0의 연구결과로 수행되었음

데이터의 저장 비용은 시간이 증가함에 따라 선형적으로 증가한다. 따라서 누적된 위치정보 처리를 위한 비용 역시 선형적으로 증가한다. 셋째, GPS 데이터는 시간 단위가 증가할 때 마다 새로운 갱신이 발생한다. 따라서 이동 객체가 정지한 경우의 위치 정보도 새로운 정보로 간주하며 시스템은 이를 저장된다. 넷째, GPS를 이용하여 이동 객체의 위치 정보를 측정하는 경우, 위치는 이 차원 유클리디안 공간에서 움직인다. 그러나 본 연구에서 가정하는 이동 객체는 도로 상에서만 움직이는 운송수단과 같은 이동 객체이기 때문에 이동 객체는 도로 네트워크 공간에서 움직인다.

위와 같이 이동 객체의 위치 정보를 GPS 데이터로 저장 및 관리하는 경우 여러 가지 비효율적인 요인들이 존재함을 알 수 있다. 본 논문에서는 이러한 문제점들을 해결하기 위하여 도로 네트워크상에서 움직이는 이동객체의 움직임에 대한 함수로 표현하는 새로운 방법을 제시한다. 이를 위하여 우리는 우선 이동 객체의 기본 속도 모델을 정의하고, 객체의 다양한 운동 형태를 이러한 모델의 변형을 통하여 표현한다.

본 논문은 다음과 같이 구성된다. 2장에서는 기존의 유클리디안 기반의 과거 궤적 표현 방법 및 도로 기반의 과거 궤적 표현 방법에 대해서 알아본다. 3장에서는 도로 세그먼트에 나타날 수 있는 교통 패턴을 정리하고 그것에 따른 과거 궤적 표현 방법을 제시한다. 그리고 기존의 과거 궤적 표현 방법과 데이터 사이즈를 비교 분석한다. 4장에서는 실제 데이터를 이용하여 본 논문이 제시하는 데이터 표현을 위한 속도 함수의 정확도에 대해 실험한다. 마지막으로, 5장에서는 향후 연구에 대해 소개하고 결론을 맺는다.

2. 관련 연구

유클리디안 공간을 기반으로 한 과거 궤적 표현이 이동 객체의 정확한 이동 경로 및 거리 등을 표현하지 못함에도 불구하고 대부분의 과거 궤적 색인 연구는 유클리디안 공간을 기반으로 이루어져 왔다. 도로 네트워크 기반의 이동 객체 표현에 대한 연구는 많이 이루어지지 않았으며, 미래 움직임 예측과 과거 움직임 분석에 몇몇이 제안되고 있다.

본 장에서는 기존의 유클리디안 공간에서의 과거 궤적 표현 방법[5, 7]과 도로 네트워크 기반의 과거 궤적 표현 방법[6]에 대해서 알아보고 예를 통하여 각각을 비교 분석 해본다. 그림 1에서 R1-R5는 도로 네트워크의 세그먼트이고 값은 거

리를 나타낸다. 만약 이동객체가 R1, R3, R5 도로 세그먼트를 지나왔고, GPS에서 받은 이동객체의 좌표가 L1 = (T1, P1) 과 L2 = (T2, P2) 라고 하자. 이 때 (T1, P1)은 시간 T1에서 위치 좌표 P1=(x1, y1) 이고 (T2, P2)는 시간 T2에서 위치 좌표 P2=(x2, y2) 를 나타낸다. 이 때, T1 ≤ T2 이다. 그림 1 에서 L1 과 L2 는 이동 객체의 위치를 이차원 공간상에 점으로 표현한 것이다.

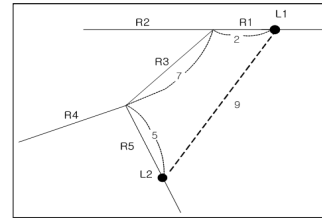


그림 1. 도로 네트워크 위의 과거 위치

2.1 유클리디안 공간을 기반으로 한 과거 궤적 표현 방법

[7]은 이동 객체의 위치데이터를 이차원 공간 정보와 일차원 시간정보를 합하여 3차원으로 표시하고, 이동 객체의 움직임 즉, 이동 객체 과거 궤적을 3차원 다중선 형태로 나타냈다. [7]이 제시하는 과거 궤적 표현은 그림 1에서 L1L2로 점선으로 나타내었다. 실제 이동 객체가 움직인 거리는 14인데 [7]에서 표현하는 궤적의 거리는 9이다. 즉 [7]에서 제시하는 과거 궤적 표현방법으로 과거 이동 객체의 이동 거리나 속도 등을 계산한다면 실제와는 다른 부정확한 데이터를 얻게 된다. [5]는 [7]에서 3차원(x, y, t) 다중선으로 표현된 과거 궤적 데이터를 힐버트 순서화한 후, 1차원의 공간 간격과 시간 간격인 2차원의 간격으로 나타낸다. 그림 1에서 (P1, T0) (P2, T1) 3차원 공간 궤적이 (X0, T0, X1, T0)으로 나타낸다. 이 방법으로 표현한 과거 궤적의 길이도 [7]과 마찬가지로 9이다. 즉 정확한 과거 궤적을 나타내지는 못한다.

2.2 도로 네트워크 기반으로 한 과거 궤적 표현 방법

[6]은 유클리디안 공간을 기반으로 한 과거 궤적 표현의 부정확성을 지적하면서 도로 네트워크 기반의 과거 궤적 표현을 제시했다. [6]은 과거 이동 객체의 궤적을 기본적인 속도 그래프를 적용시켜 표현했다. 그러나 기본적인 속도 그래프를 전체 이동 객체에 동일하게 적용시킴으로써 속도의 다양성을 반영하지 못한다.

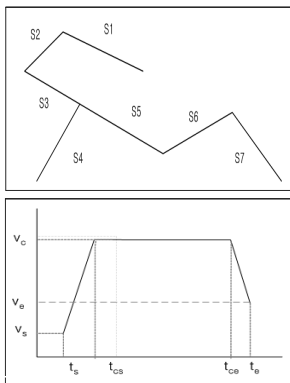


그림 2. 도로 예제 및 기본 속도 그래프

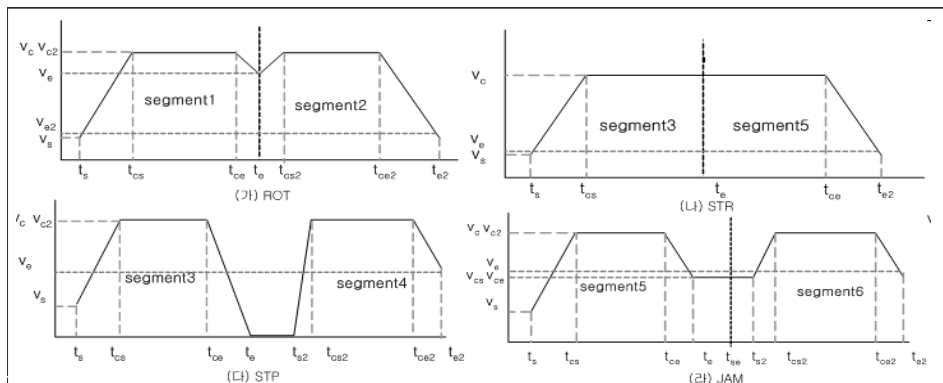


그림 3. 이동 객체의 다양한 속도 형태

3. 속도 그래프를 이용한 과거 궤적 표현 방법

본 논문에서는 유클리디언 공간을 기반으로 한 과거 궤적 표현의 단점을 보완하는 도로 네트워크 기반의 과거 궤적 표현 방법에 대해 제시한다. 이때 과거 궤적은 속도를 이용하여 표현하는데, 본 논문에서는 도로 네트워크에서 발생할 수 있는 속도 패턴을 정의하고 거기에 따른 궤적 표현 방법을 제시한다. 이를 위해 하나의 도로 세그먼트에서 발생하는 기본적인 속도 그래프는 그림 2와 같다고 가정한다[6]. 하나의 도로 세그먼트에서 이동 객체는 일정한 속도로 가속하고 등속도로 움직인 다음 다시 일정한 속도로 감속한다. 이동 객체는 t_s 시간에 v_s 속도로 도로 세그먼트에 진입하고 t_{cs} 시간에 등속도로 움직인 다음 t_{ce} 시간 이후로 감속하여 t_e 시간에 v_e 속도로 이 도로 세그먼트를 지났다 가정한다.

3.1 속도 그래프를 적용하여 과거 궤적 표현하기

도로 네트워크에서는 신호 상황과 도로 세그먼트의 연결 관계에 따라서 속도가 달라질 수 있다. 그러므로 모든 이동 객체에게 동일한 속도 그래프 (그림 2)를 적용시키는 것은 정확한 궤적 데이터를 표현하는데 무리가 있다. 따라서 본 논문에서는 도로 네트워크에서 발생할 수 있는 교통 패턴을 정리하였다. 그림 2의 도로 예제에서 $S_1 \sim S_7$ 은 각각의 도로 세그먼트 아이디를 나타낸다.

그림 2의 도로 네트워크에서 발생할 수 있는 교통 패턴은 다음과 같다.

- $\overline{S_1 S_2}$ 에서 회전할 경우 (ROT)
- $\overline{S_3 S_5}$ 에서 직진할 경우 (STR)
- $\overline{S_3 S_4}$ 에서 정지신호를 만나는 경우 (STP)
- $\overline{S_5 S_6}$ 에서 교통 정체인 경우 (JAM)

위의 4가지 경우의 속도 그래프와 각각의 과거 궤적 표현은 다음과 같다.

● ROT

그림 3 (가)의 S_1 에서 S_2 로 통행 신호 시 우회전이나 좌회전 때문에 S_1 끝의 일정한 부분부터 속도가 줄어든다. 따라서 이동 객체는 그림 3 (가)와 같은 속도 그래프를 가지게 된다. 이동 객체는 S_1 에 v_s 속도로 t_s 시간에 진입하고 일정한 가속도로 가속한다. 그리고 t_{cs} 지점에서 일정한 속도 v_c 로 움직이다가 S_2 로 가기 위한 회전을 위해 감속해서 t_e 시간에는 v_e 속도를 가지게 된다. 이런 속도 패턴으로 이동하는 이동 객체의 과거 궤적 표현은 다음과 같다.

- $(S_1, (t_s, v_s), (t_{cs}, t_{ce}, v_c))$
- $(S_2, (t_e, v_e), (t_{cs2}, t_{ce2}, v_{c2}))$

● STR

그림 3 (나)에서 S_3 과 S_5 는 직선도로이다. 그래서 만약 S_3 에서 S_5 로 정지 신호 없이 간다고 하면 S_3 이 끝나더라도 이동 객체의 속도는 줄어들지 않고 일정한 속도를 가지고 S_5 로 가게 된다. 따라서 이동 객체는 그림 3 (나)와 같은 속도 그래프를 가지게 된다. 이동 객체는 S_3 에 v_s 속도로 t_s 시간에 진입해서 일정하게 가속한다. 또한 t_{cs} 시간부터 일정한 속도 v_c 를 가지고 이동하는데 S_5 에 진입해서도 그 속도 v_c 를 유지

한다. 이런 속도 패턴으로 이동하는 이동 객체의 과거 궤적 표현은 다음과 같다.

- $(S_3, (t_s, v_s), (t_{cs}, v_c))$
- $(S_5, (t_e, t_{ce}, v_c))$

● STP

그림 3 (다)에서 만약 이동 객체가 S_3 에서 정지 신호 후 S_4 로 움직인다면 이동 객체는 S_3 에서 완전 정지한 후 S_4 로 갈 것이다. 따라서 이동 객체는 그림 3 (다)과 같은 속도 그래프를 가질 것이다. 이동 객체는 S_3 에 v_s 속도로 t_s 시간에 진입해서 일정하게 가속한다. 또한 t_{cs} 시간부터 일정한 속도 v_c 를 가지고 이동하고 t_e 시간에는 신호를 받아 완전히 정지한다. 그리고 t_{s2} 시간에 S_4 로 일정한 가속도를 가지고 이동 할 것이다. 이런 속도 패턴을 이용하여 이동 객체의 과거 궤적 표현하면 다음과 같다.

- $(S_3, (t_s, v_s), (t_{cs}, t_{ce}, v_c), (t_e, 0))$
- $(S_4, (t_{s2}, 0), (t_{cs2}, t_{ce2}, v_{c2}))$

● JAM

그림 3 (다)에서 S_6 으로 이동하는 이동 객체는 S_5 에서의 교통 정체로 인하여 속도에 영향을 받을 수 있다. 이때 이동 객체의 속도 그래프는 그림 3 (다)와 같다. 이동 객체는 S_5 에 v_s 속도로 t_s 시간에 진입해서 일정하게 가속한다. 그리고 시간 t_{ce} 에서 교통 정체로 인하여 속도가 감속되다가 다시 일정한 속도 v_{cs} 를 가지고 S_6 으로 간다. 이런 속도 패턴을 이용하여 이동 객체의 과거 궤적 표현하면 다음과 같다.

- $(S_3, (t_s, v_s), (t_{cs}, t_{ce}, v_c), (t_e, v_{cs}))$
- $(S_4, (t_{se}, t_{s2}, v_{ce}), (t_{cs2}, t_{ce2}, v_{c2}))$

3.2 기존의 과거 궤적과 데이터 사이즈 비교

기존 유클리디언 공간 기반의 과거 이동 객체 궤적 데이터는 시간에 따라 용량이 커진다. 그래서 저장 비용이 증가하고 거기에 따라 분석 비용 또한 증가한다. 따라서 본 장에서는 유클리디언 공간 기반의 데이터 사이즈와 도로 네트워크 기반의 데이터 사이즈, 본 논문에서 제시하는 속도를 이용한 데이터의 사이즈를 비교 분석한다.

표1. 데이터 사이즈 비교

	1 record/Seg	k record/Seg
(x, y, t)	12bytes	12*k bytes
$(Sid, offset, t)$	12bytes	4+ 8*k bytes
(Sid, t, v)	약 24bytes	약 24bytes

표 1은 3개의 데이터 표현의 사이즈를 비교한 것이다. 유클리디언 공간을 기반으로 한 (x, y, t) 의 데이터 표현 방법에서 하나의 레코드 크기는 4 bytes * 3 = 12 bytes이다. 도로 네트워크 기반의 표현 방법 중 하나인 $(Sid, offset, t)$ 데이터 표현 방법에서 하나의 레코드 크기는 4 bytes * 3 = 12 bytes이다. 본 논문에서 제시하는 속도를 이용한 과거 데이터 표현 방법의 평균 사이즈는 약 6 * 4bytes = 24bytes이다. 한 개의 과거 위치 데이터에 대해 도로 네트워크 기반의 표현 방법 $(Sid, offset, t)$ 은 유클리디언 공간을 기반으로 한 데이터 표현 방법과 레코드의 크기가 같고, 본 논문이 제시하는 방법은 그것의 2배 크기이다. 그러나 하나의 도로 세그

먼트에 k 개의 위치 데이터가 있는 경우 (x, y, t) 는 $12 * k$ bytes이고 $(Sid, offset, t)$ 는 $Sid(4bytes)$ 와 k 개의 $(offset, t)$ 로 $4 + 8 * k$ bytes가 된다. 반면에 (Sid, t, v) 는 하나의 도로 세그먼트에 대해 전체 속도로 표현한 방법이므로 k 개의 데이터에 대해서도 한 개의 데이터 크기 만큼인 24bytes이다. 하나의 도로 세그먼트에 2.5개 이상의 데이터가 있을 경우 유클리디언 공간 표현보다 압축률이 높다.

4. 실험

본 장에서는 본 논문이 제시하는 도로 네트워크 기반 위치 데이터 표현에 이용한 속도의 정확도를 알아본다.

4.1 성능 평가 데이터 및 환경

속도의 정확도를 알아보는 실험을 위해, 부산시 도로 위를 움직이는 하나의 이동 객체에 대해 1초 간격으로 GPS 데이터 6300개를 수집하였다. 22893의 에지를 가지는 전체 부산시 도로 네트워크의 한 부분인, 18946의 에지를 가지는 부산시 도로 네트워크 위를 움직이는 485개의 데이터에 대해서 실험 하였다. 그림 4는 실험할 실 데이터를 나타낸다.

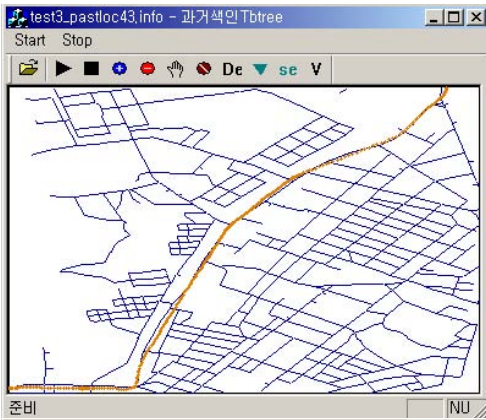


그림 4. 부산시 도로 네트워크와 이동 객체의 궤적

4.2 속도의 정확도

그림 5는 485개 실 데이터에 대한 속도와 본 논문이 제시하는 위치 데이터 표현을 위한 속도를 나타낸다. 속도는 $[0, 1]$ 로 정규화했고, 실제 데이터의 속도를 분석하여 본 논문의 속도 함수를 구하였다. 속도에 특정 임계값을 주고 계속 증가하는 구간과 감소하는 구간을 구하고 증가와 감속 구간 사이에 있는 값들을 평균 내어 그래프를 그렸다. 그림 5에서 보면 112초와 191초에 속도가 줄어들었다가 다시 증가함을 볼 수 있다. 이는 도로의 커브로 인한 것인데 즉 그림 3 (가)와 같은 속도 그래프를 가짐을 알 수 있다. 그리고 21초에서 86초에는 정지 신호로 인하여 거의 정지한 상태로 그림 3 (다)와 같은 속도 그래프를 가진다. 또 130초에서 134초 사이에 서로 다른 도로 세그먼트를 달리고 있지만 속도 그래프에서는 거의 비슷한 속도를 나타내고 있다. 이는 거의 직선 도로로 정지 신호 없이 지나간 경우이기 때문에 그림 3 (나)와 같은 속도 그래프를 가진다. 그림 5의 실제 데이터의 속도와 본 논문에서 제시하는 표현을 위한 속도의 분산 값은 약 0.3% 이하로 거의 동일함을 알 수 있다.

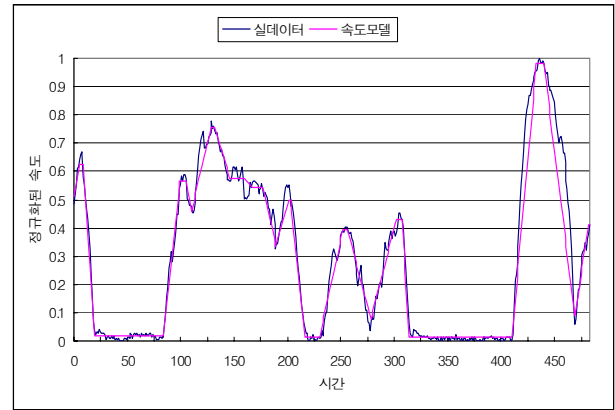


그림 5. 실 데이터와 속도 모델의 비교

5. 결론

본 논문에서는 속도를 이용한 도로 네트워크 기반의 데이터 표현 방법에 대해 제시했다. 이 방법은 도로 세그먼트에 나타날 수 있는 교통 패턴에 따라 궤적을 표현함으로써 기존의 유클리디언 기반 데이터 표현 방법보다 저장 측면에서 효율적임을 보였다. 우리의 연구는 GPS 데이터와 도로 데이터에 대한 일차 작업을 사전에 수행하여 저장하였기 때문에 누적된 과거 데이터에 대한 접근 및 분석 작업을 보다 더 효율적으로 수행할 수 있을 것이다. 이후의 연구에서 우리는 이러한 특성을 이용한 데이터마이닝 방법을 개발할 것이며, 속도 그래프를 적용한 합성 데이터 툴을 개발할 것이다.

6. 참고 문헌

- [1]Y. Tao, C. Faloutsos, D. Papadias, and B. Liu, "Prediction and Indexing of Moving Objects with Unknown Motion Patterns", *proc. SIGMOD*, pp. 611-622, 2004
- [2]V.T. de Almeida and R.H. Güting, "Indexing the Trajectories of Moving Objects in Networks", *Fernuniversitat Hagen, Informatik-Report 309*, March 2004.
- [3]N. Meratnia and R. A. de By, "Spatiotemporal Compression Techniques for Moving Point Objects", *EDBT*, pp. 765-782, 2004
- [4]L. Speicys, C. S. Jensen, and A. Kligys, "Computational data modeling for network-constrained moving objects", *ACM-GIS*, pp. 118-125, 2003
- [5]D. Pfoser and C. S. Jensen, "Indexing of network constrained moving objects", *ACM-GIS*, pp. 25-32, 2003
- [6]N. Meratnia and R. A. de By, "Trajectory Representation in Location-based Services: Problems & Solution", *W2GIS*, 2003
- [7]D. Pfoser, C. S. Jensen, and Y. Theodoridis, "Novel Approaches in Query Processing for Moving Object Trajectories", *proc. VLDB*, pp.395-406, 2000