TransDec: A Data-Driven Framework for Decision-Making in Transportation Systems

Cyrus Shahabi
University of Southern California
Los Angeles, CA 90089-0781
shahabi@usc.edu
http://infolab.usc.edu

OUTLINE

- Motivation: Transportation
- Platform: TransDec
- Traffic Profiles
  - Data Reduction Approach
- Applications: Time-Dependent kNN
- Demonstration
- Future Plans

OUTLINE

- Motivation: Transportation
- Platform: TransDec
- Traffic Profiles
  - Data Reduction Approach
- Applications: Time-Dependent kNN
- Demonstration
- Future Plans

Transportation Policy Planning

PROBLEM

- Traffic congestion is a **$87.2 billion annual drain** on the U.S. economy[1]:
  - 4.2 billion lost hours (one work week for every traveler)[1]
  - 2.8 billion gallons of wasted fuel (three weeks worth of gas for every traveler)[1]

[1] Texas Transportation Institute Urban Mobility Report, 2007 data

GOAL

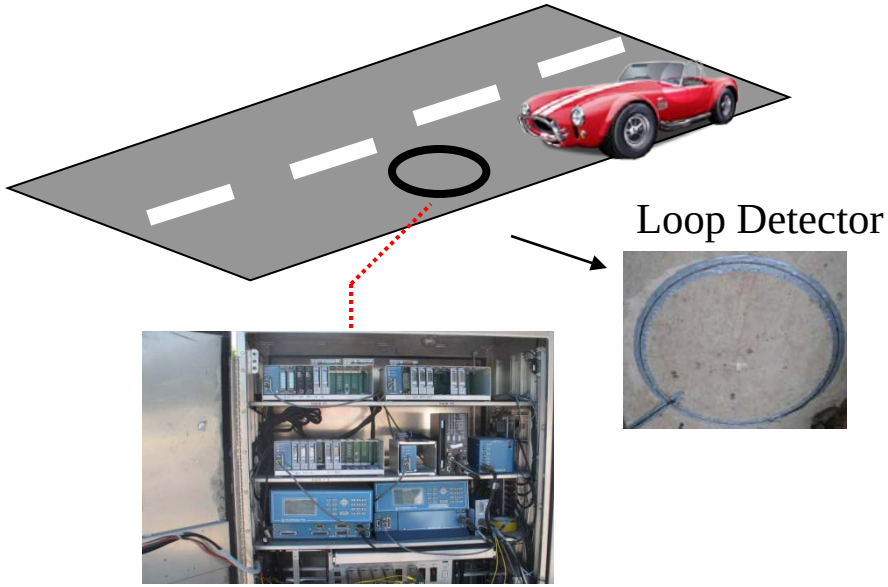
- To improve the performance of the surface transportation network through:
  - Capturing real-time data from infrastructure and vehicles (cars, trucks)
  - Developing data-driven solutions for holistic enhancement of mobility by leveraging optimization opportunities (e.g., path planning for commuter groups)

U.S. Department of Transportation
Research and Innovative Technology Administration (RITA)

IntelliDrive℠
Safer. Smarter. Greener.

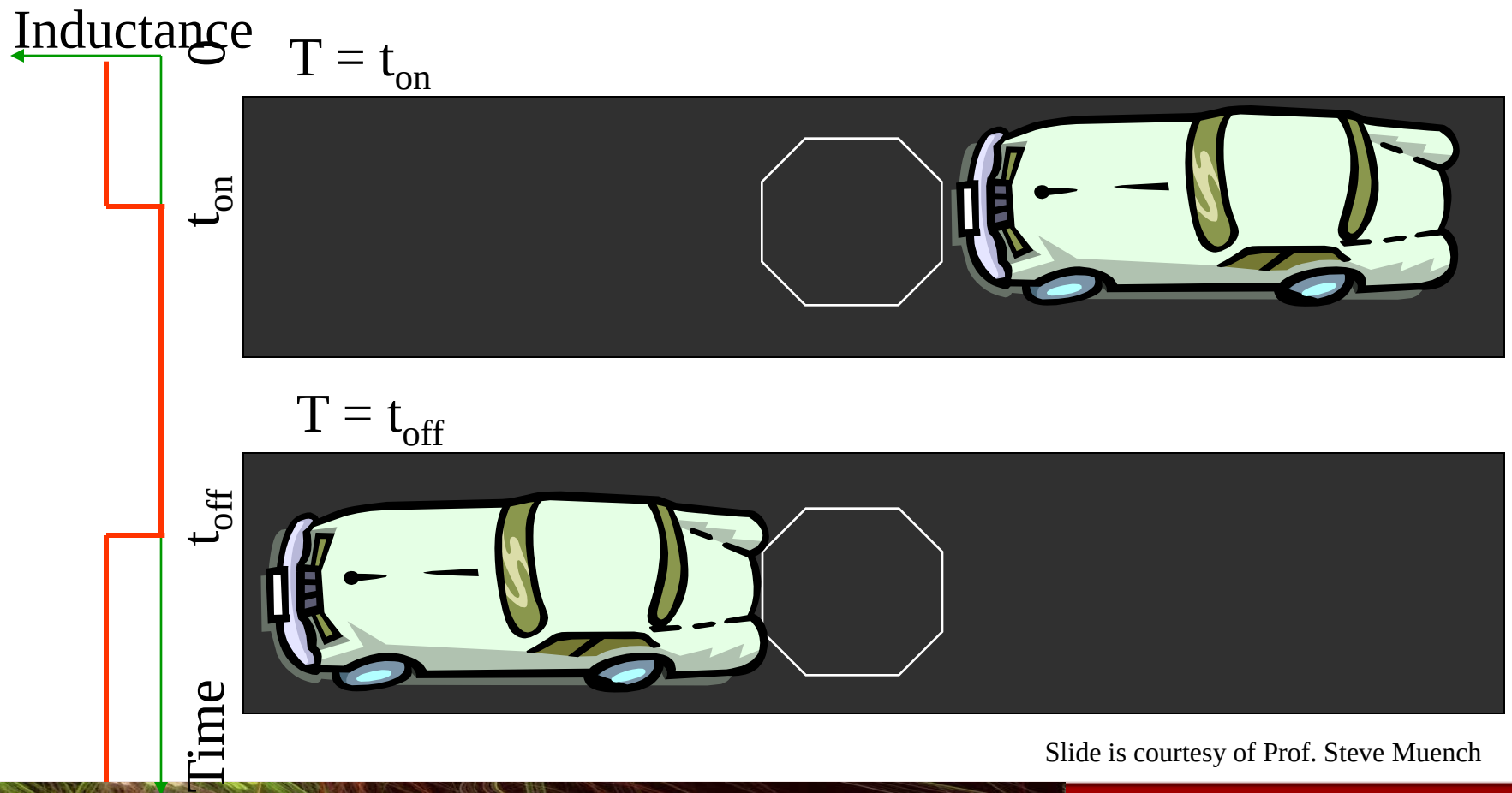
Traffic Data Lifecycle: Loop Detectors

- Loop Detector: most commonly used traffic sensors
- The data is collected in Detector Cabinet and relayed to the service provider.
- Provide two data fields: volume (count) and occupancy (% time a vehicle is over the sensor)



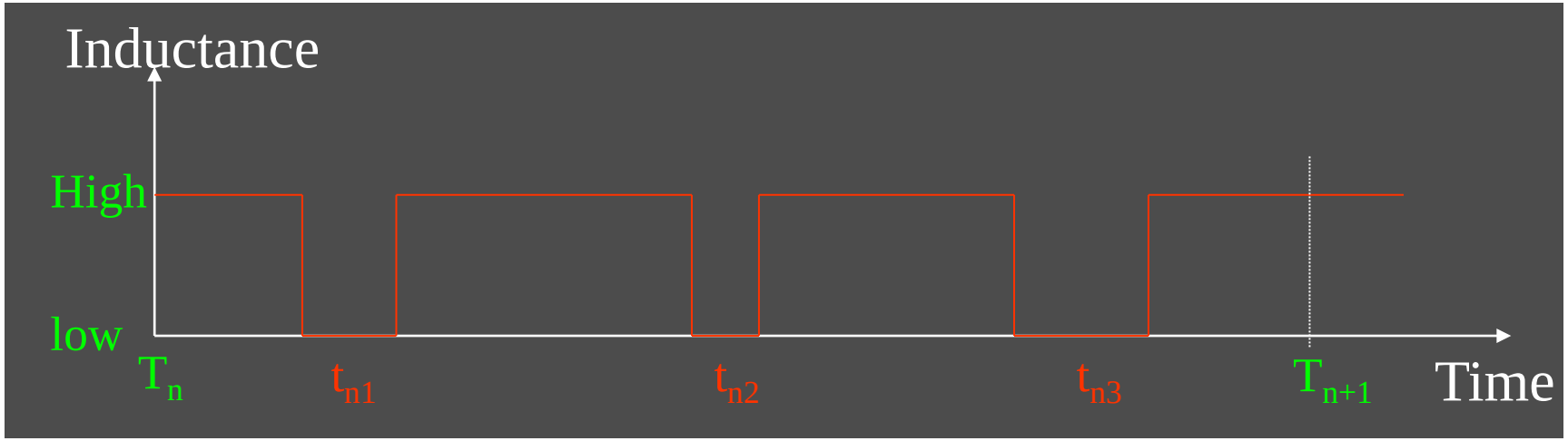
Traffic Data Lifecycle: Loop Detectors

Loop inductance decreases when a car is on top of it.



Slide is courtesy of Prof. Steve Muench

Traffic Data Lifecycle: Loop Detectors



- **Single loops can measure:**
  - **Occupancy (*O*): % of time loop is occupied (had a car on it) per interval**
  - **Volume (*N*): vehicles per interval**
  - **Speed = (N*L)/O where L is a constant proportional to the average length of a car**

Slide is courtesy of Prof. Steve Muench

Traffic Data Lifecycle: Data Aggregator

RIITS (Regional Integration of Intelligent Transportation Systems)

- A data network affiliated with Los Angeles County Metropolitan Transportation Authority (Metro)
- Collects and serves data from Caltrans, City of Los Angeles Department of Transportation (LADOT), California Highway Patrol (CHP), Long Beach Transit (LBT), Foothill Transit (FHT) and Metro

http://www.riits.net/

Traffic Data Lifecycle: Data Aggregator

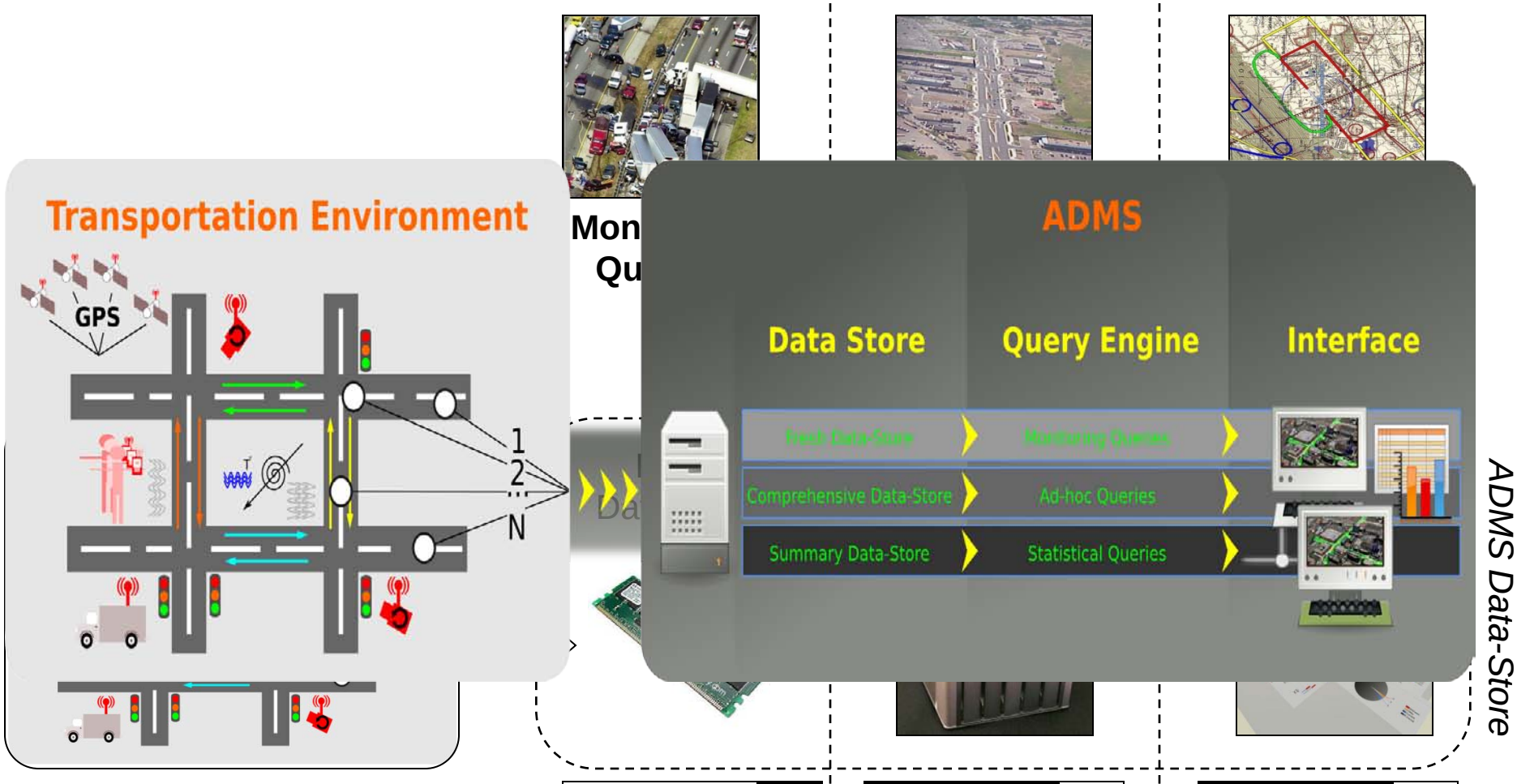
Heterogeneous
(gps, video, loop sensor, events)

Data Type				Hourly (in KB)	Daily (in KB)	Annual (in KB)	3 Years (in KB)
bus_mta_inv2.xml	[illegible]	[illegible]	[illegible]	0.96	23.00	8,395.00	25,185.00
bus_mta_rt2.xml	1065	120	532.50	31,950.00	766,800.00	279,882,000.00	839,646,000.00
cctv_inv.xml	57	86400	0.04	2.38	57.00	20,805.00	62,415.00
cms_inv.xml	52	86400	0.04	2.17	52.00	18,980.00	56,940.00
cms_rt.xml	48	75	38.40	2,304.00	55,296.00	20,183,040.00	60,549,120.00
event_d7.xml	11	75	8.80	528.00	12,672.00	4,625,280.00	13,875,840.00
rail_mta_inv.xml	1	86400	0.00	0.04	1.00	365.00	1,095.00
rail_rt.xml	8	60	8.00	480.00	11,520.00	4,204,800.00	12,614,400.00
rms_inv.xml	865	86400	0.60	36.04	865.00	315,725.00	947,175.00
rms_rt.xml	1236	75	988.80	59,328.00	1,423,872.00	519,713,280.00	1,559,139,840.00
signal_inv.xml	2095	86400	1.45	87.29	2,095.00	764,675.00	2,294,025.00
signal_rt.xml	2636	45	3,514.67	210,880.00	5,061,120.00	1,847,308,800.00	5,541,926,400.00
tt_d7_inv.xml	746	86400	0.52	31.08	746.00	272,290.00	816,870.00
tt_d7_rt.xml	152	60	152.00	9,120.00	218,880.00	79,891,200.00	239,673,600.00
vds_art_d7_inv.xml	115	86400	0.08	4.79	115.00	41,975	[illegible]
vds_art_d7_rt.xml	45	60	45.00	2,700.00	64,800.00	23,652,000	[illegible]
vds_art_ladot_inv.xml	2538	86400	1.76	105.75	2,538.00	926,370.00	2,779,110.00
vds_art_ladot_rt.xml	969	60	969.00	58,140.00	1,395,360.00	509,306,400.00	1,527,919,200.00
vds_fr_d7_inv.xml	957	86400	0.66	39.88	957.00	349,305.00	1,047,915.00
vds_fr_d7_rt.xml	361	30	722.00	43,320.00	1,039,680.00	379,483,200.00	1,138,449,600.00
Total KB from XML data	**13980**	**864660**	**6,985.28**	**41**	[illegible]	[illegible]**,885.00**	**11,012,906,655.00**

Continuous

Large

Requirement (by Metropolitan Transportation Authority) – Archive Data Management System (ADMS)



Technical Challenges of ADMS

- **Data Integration** *to handle Data Heterogeneity*
- **In-Memory Storage** *to handle Data Dynamics*

- **Data Indexing** *to handle Data Size*

- **Data Summarization** *for multi-resolution pre-aggregation*

Monitoring-Query Module

Ad hoc Query Module

Statistical Query Module

ADMS Query Engine

Fresh Data-Store

Comprehensive Data-Store

Summary Data-Store

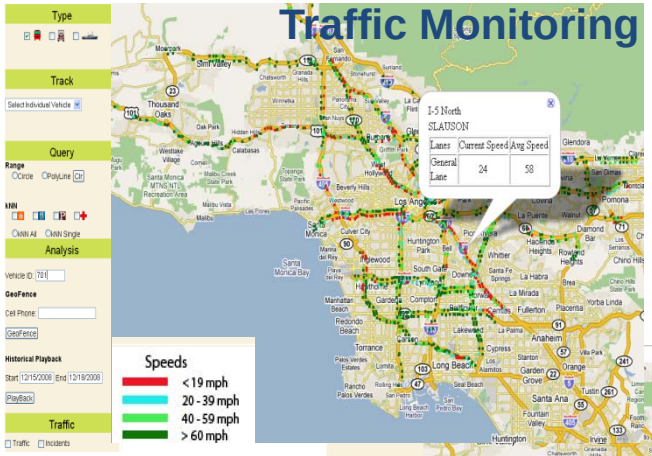
ADMS Data-Store

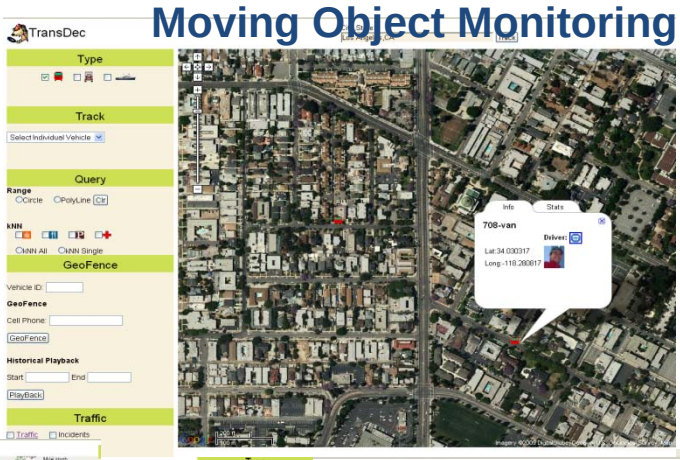
OUTLINE

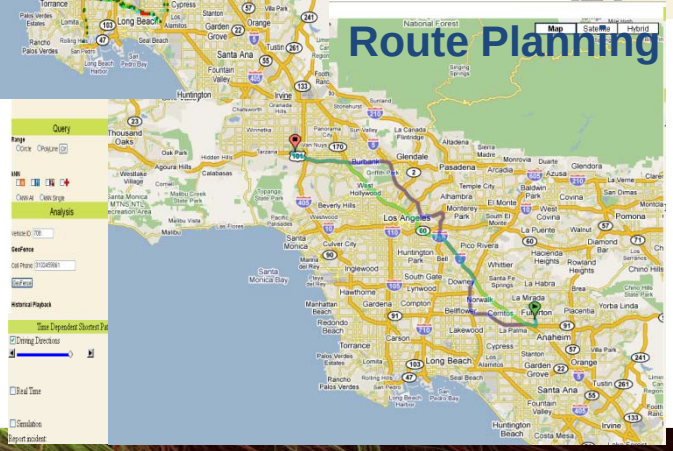
- Motivation: Transportation
- Platform: TransDec
- Traffic Profiles
  - Data Reduction Approach
- Applications: Time-Dependent kNN
- Demonstration
- Future Plans

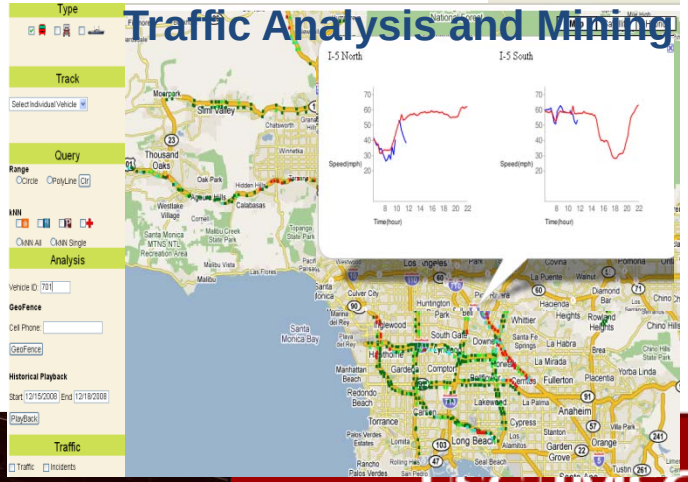
TransDec Mission

A **data-driven** tool that enables **on- the-fly spatio-temporal querying, analysis** and **planning** for transportation networks







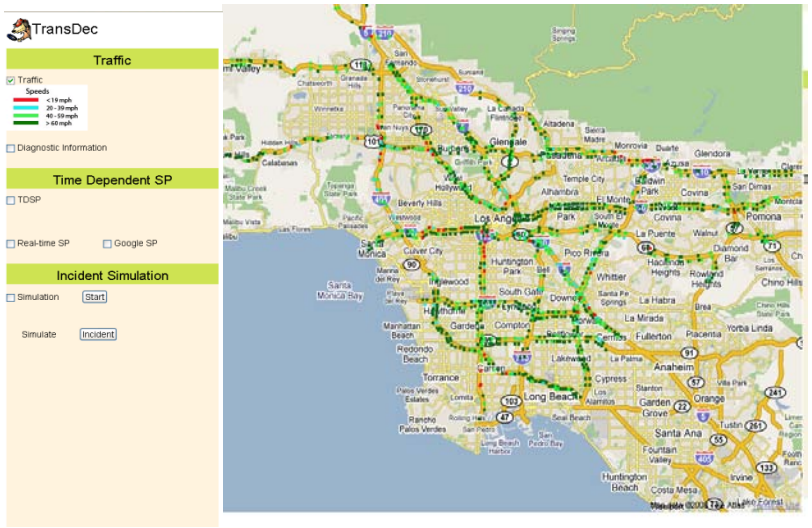


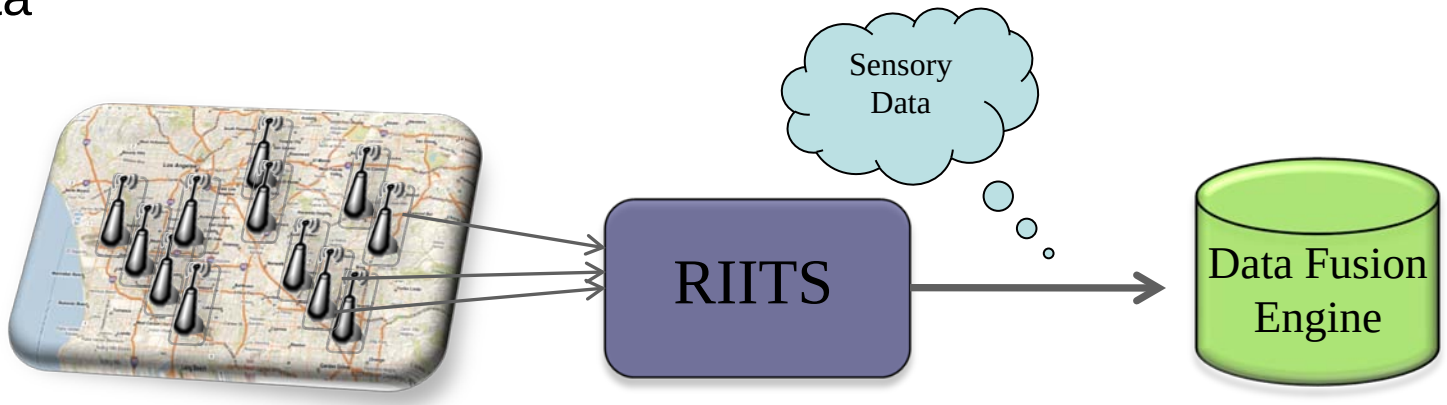
Traffic Sensors (currently)

- Currently provided by RIITS
  - –Real-time Highway Congestion
  - –Real-time Arterial Congestion

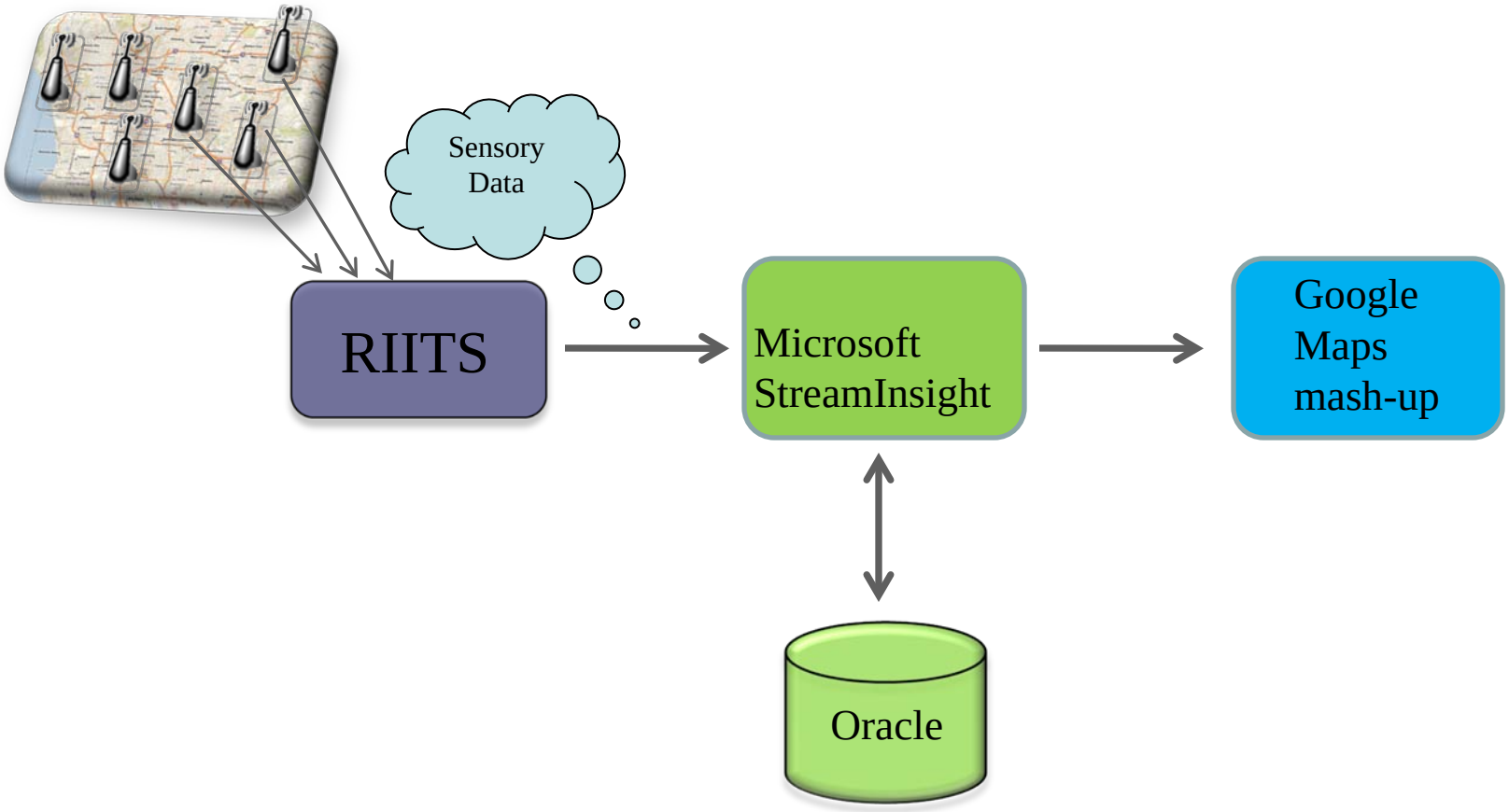
- Total of **6300 sensors** (1800 sensors in 18 highways and 4500 in main streets) covering 3000 miles

- Update rate: every **1 minute**
  - –Daily 9.1 million rows, **800MB** of data

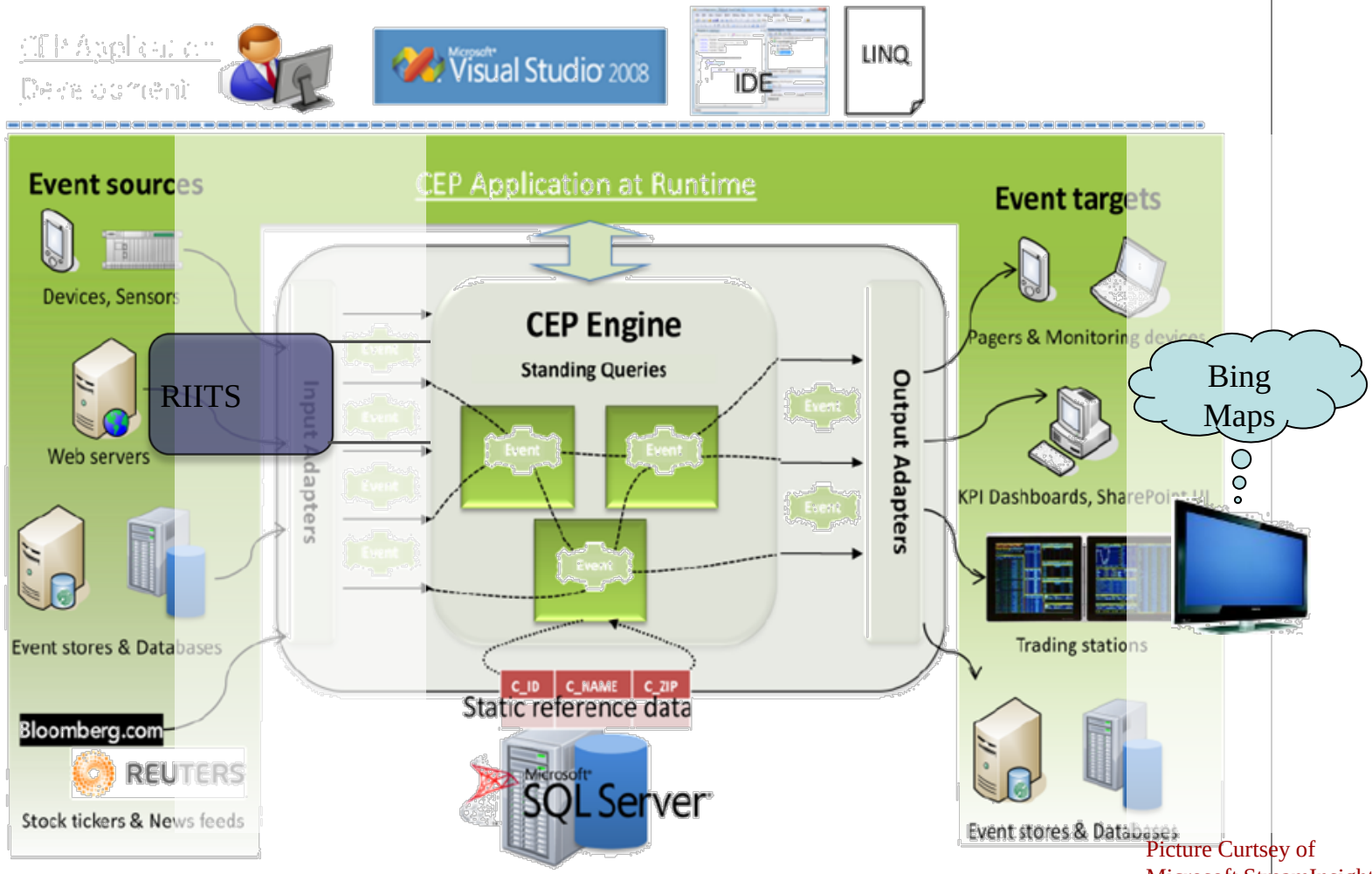




Current System Architecture



Current System Architecture [VLDB2010-demo]



Picture Curtsey of Microsoft StreamInsight

OUTLINE

- Motivation: Transportation
- Platform: TransDec
- Traffic Profiles
  - Data Reduction Approach
- Applications: Time-Dependent kNN
- Demonstration
- Future Plans

Large Dataset:
Should we store everything?

- Can current Database systems answer historic queries in a reasonable time?

  "*What was the average speed in I-10 during summer 2009 from 4:00-5:00 pm?*"

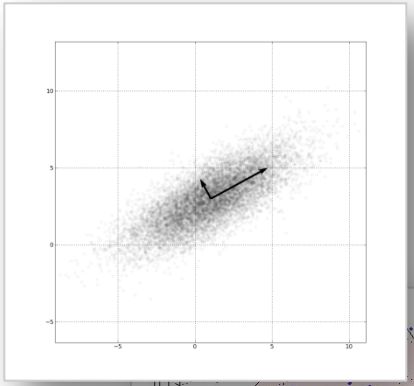
Response Time for the **indexed** table containing data of one year: 58 Seconds!

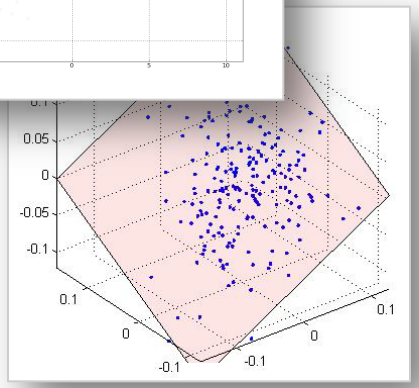
- Lack of methods with short response time
- Can knowledge of Traffic Data help us to reduce data?

Data Reduction Approach

- Instead of storing the whole data in DB, store the sketches **in memory**
- Principal component Analysis (PCA): a mathematical approach for analyzing correlated data
- components are found using singular value decomposition $COV(X) = USV^T$

components

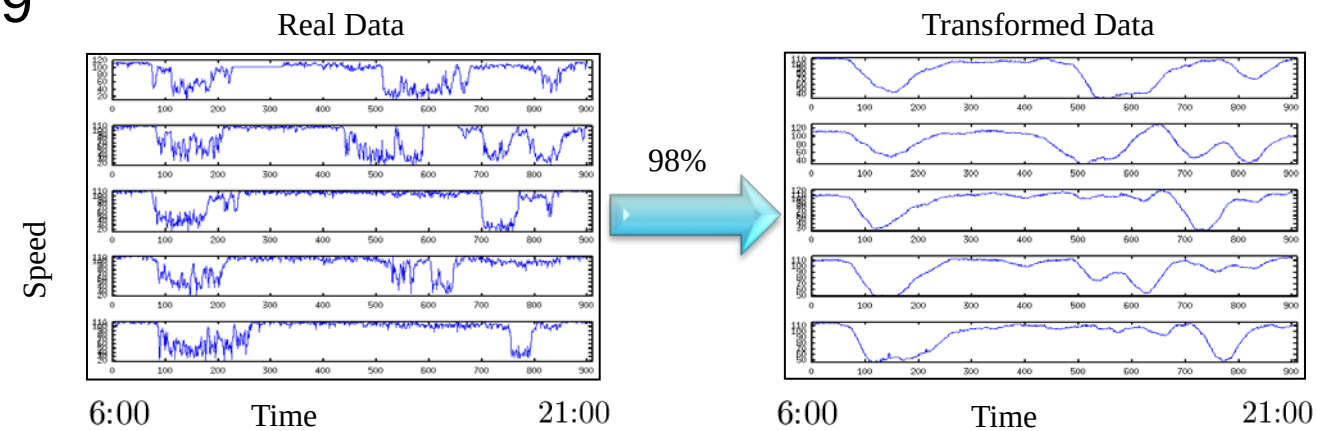




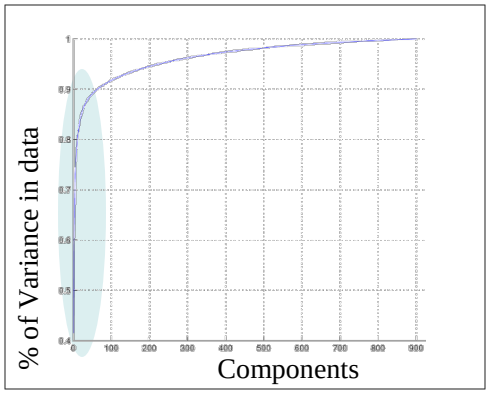
- A number of components with great influence selected as coordinates
- Representing data using new coordinates

PCA for Traffic Data

- Results for a sensor in I-10 for 5 consecutive days in Summer 2009

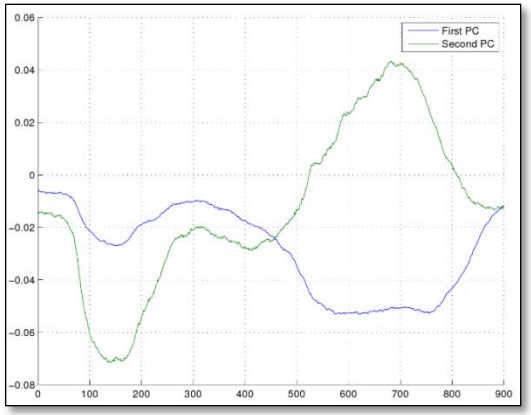


- A small number of components hold the major variance in data
- High data compression rate
  - 98% for highway data



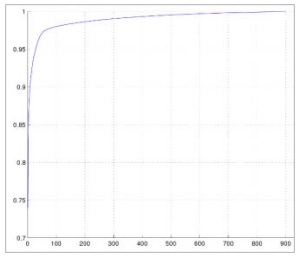
PCA for Traffic Data

- Extra short response time
  - 1 millisecond (compare to 58 sec.)
- Highly accurate for Traffic Data
  - MSE for same query: 10^{-4} Mph
- Finding hidden patterns
  - The Two major components in I-10
  - There are less traffic patterns in I-210 compared to I-10

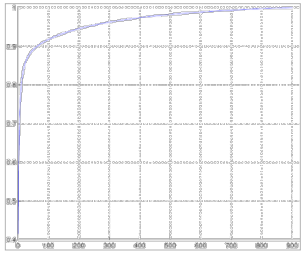


Time

I-210



I-10



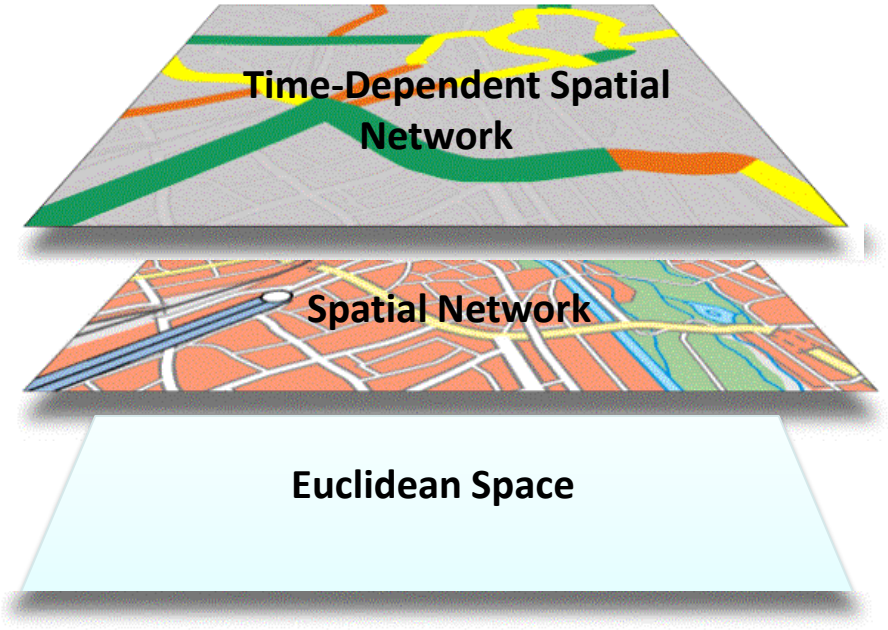
OUTLINE

- Motivation: Transportation
- Platform: TransDec
- Traffic Profiles
  - Data Reduction Approach
- Applications: Time-Dependent kNN
- Demonstration
- Future Plans

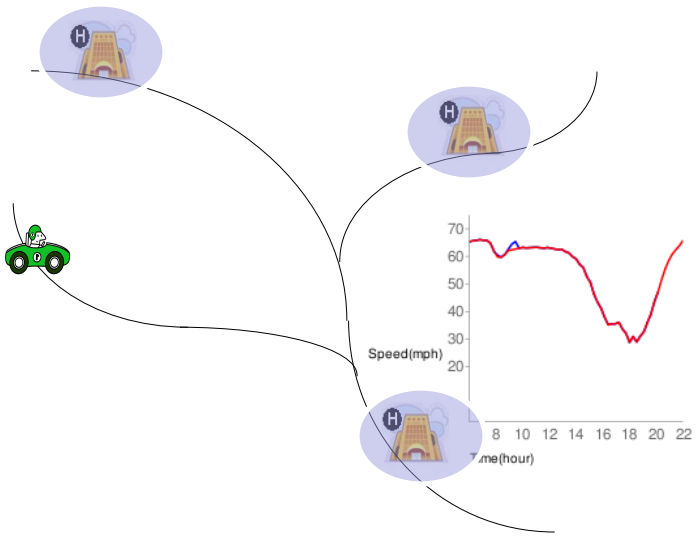
Outline

- KNN Search Past and Present
- Motivation
- Related Work
- Time-Dependent Spatial Network
- TD-kNN Search Approaches
  - Time-Expanded Networks
  - Incremental Network Expansion
  - Indexing Time-Dependent Networks
- Conclusion

KNN Search Past and Present



Time-Dependent Spatial Network



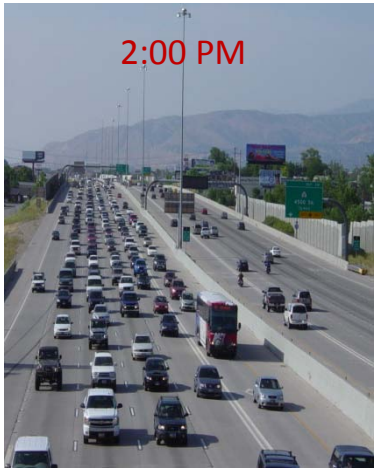
Edge weight change with time

Outline

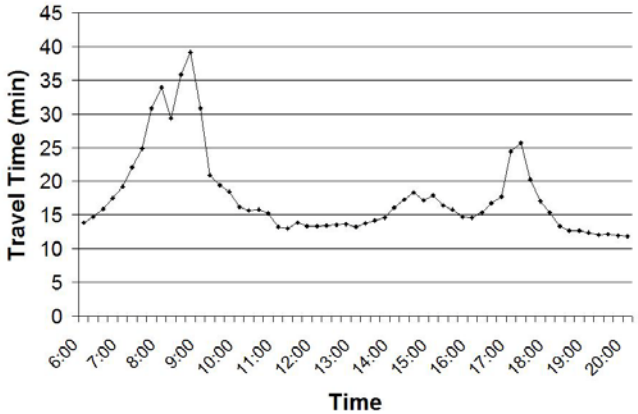
- KNN Search Past and Present
- Motivation
- Related Work
- Time-Dependent Spatial Network
- TD-kNN Search Approaches
  - Time-Expanded Networks
  - Incremental Network Expansion
  - Indexing Time-Dependent Networks
- Conclusion

Motivation

- Existing kNN research and commercial systems
  - Find shortest path based on the **constant** edge weights.
- In Real-world
  - The weight of an edge is a function of time, i.e., **time-dependent.**
  - Arrival-time to an edge determines the travel-time on that edge.
  - Change in travel-time is significant and continuous not abrupt.



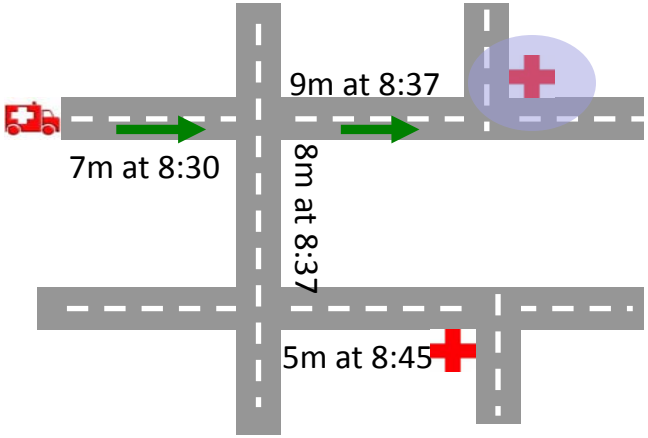
Pictures courtesy : http://www.wfrc.org/cms



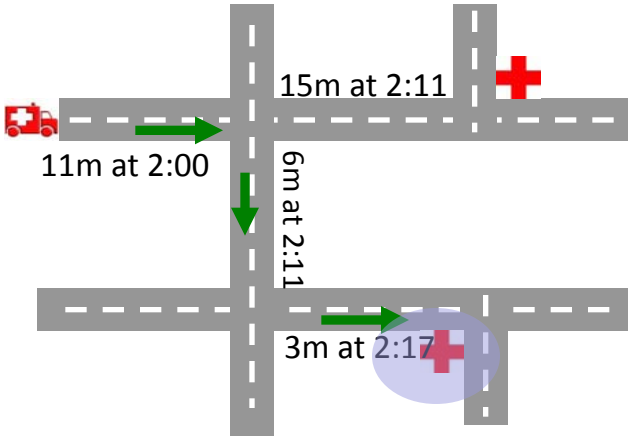
Monday travel-time on a segment of I-405 in LA

Time-dependent kNN (TD-kNN)

- Given a query point *q* and a set of objects *P*, find the k objects in *P* that are closest to *q in time-dependent cost (e.g., travel-time)*



1-NN Query at 8:30 AM



1-NN Query at 2:00 PM

- The shortest path from source to candidate hospitals may change depending on the departure-time
- The result of kNN depend on when the query is issued

Outline

- KNN Search Past and Present
- Motivation
- Related Work
- Time-Dependent Spatial Network
- TD-kNN Search Approaches
  - Time-Expanded Networks
  - Incremental Network Expansion
  - Indexing Time-Dependent Networks
- Conclusion

Related Work

Spatial Networks

Shortest Path

Dijkstra
A*

kNN Search

- Query processing in SNDB : Papadias et al., [VLDB03]
- Voronoi-based kNN in SNDB: Shahabi et al., [VLDB04]
- CNN queries in RN: Cho et al., [VLDB05]
- CNN monitoring in RN: Mouratidis et al., [VLDB06]
- S-Grid: Huang et al., [SSTD07]
- Scalable network distance browsing: Samet et al., [SIGMOD08]

All kNN approaches are based precomputation of static SP algorithms

Related Work

Time-Dependent Spatial Networks

Time-Dependent Shortest Path

Fixed Edge Cost (FC)

- Cook and Halsey, [JMA66] – Dynamic Programming
- Kohler et al., [ESA02]– Time-Expanded Graphs

Variable Edge Cost (VC)

- Dreyfus, [JOR69] – Dijkstra Variant in FIFO
- Harpen, [MMO69]- Non-FIFO, NP Hard
- Orda and Rom, [JACM90] – Bellman Ford Algorithm
- George and Shekhar, [SSTD07]- Time-Aggregated Graphs

Time-Dependent kNN Search

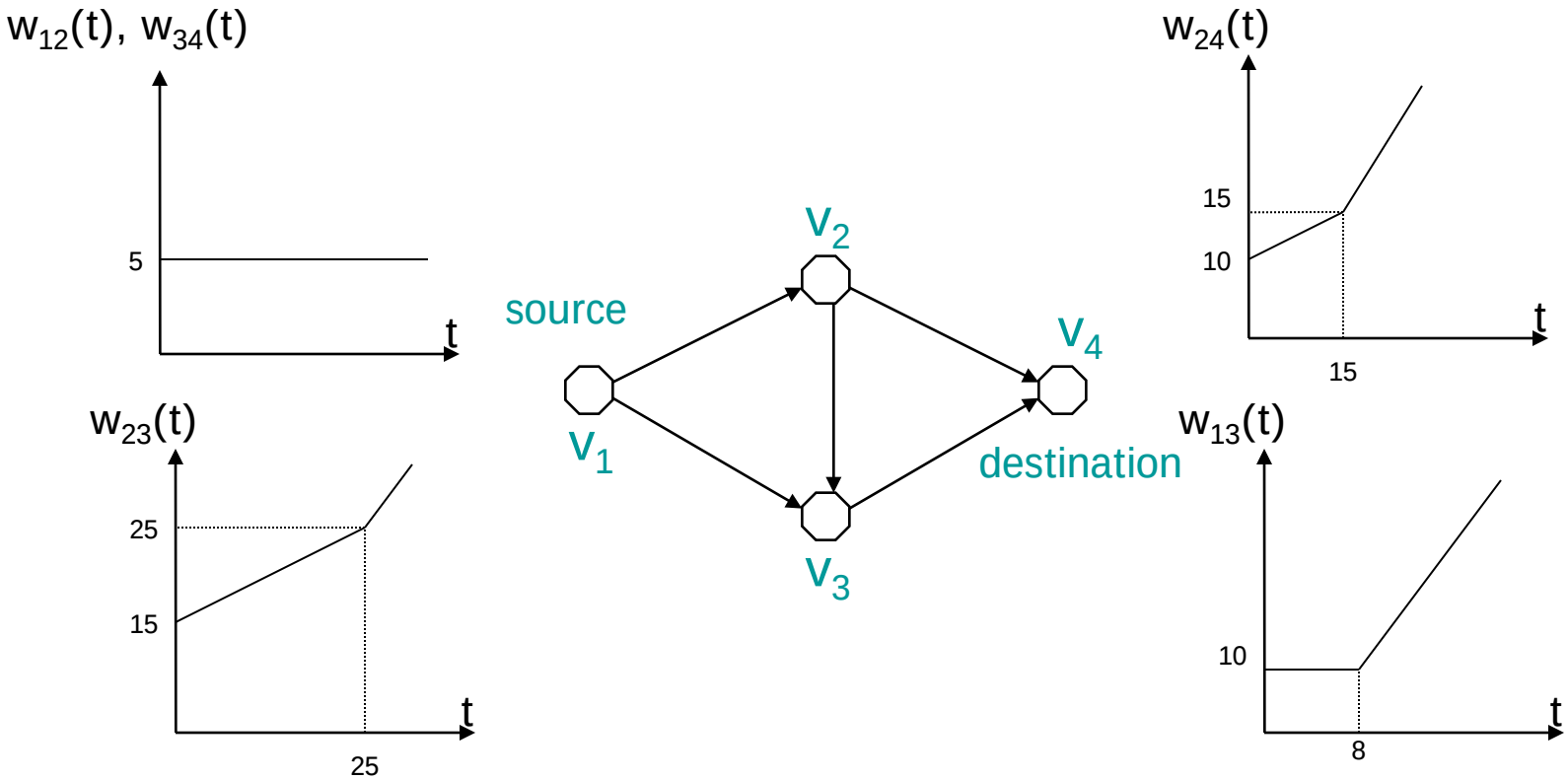
Not Studied

- Both of DT and CT algorithms can be extended to address TD-kNN. **However, they are either approximate or do not scale.**
- Precomputation in time-dependent networks is a big **challenge.**

Outline

- KNN Search Past and Present
- Motivation
- Related Work
- Time-Dependent Spatial Network
- TD-kNN Search Approaches
  - Time-Expanded Networks
  - Incremental Network Expansion
  - Indexing Time-Dependent Networks
- Conclusion

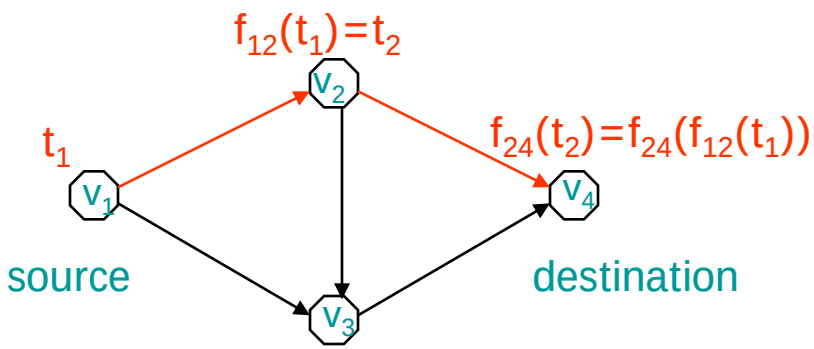
Time-dependent Spatial Network

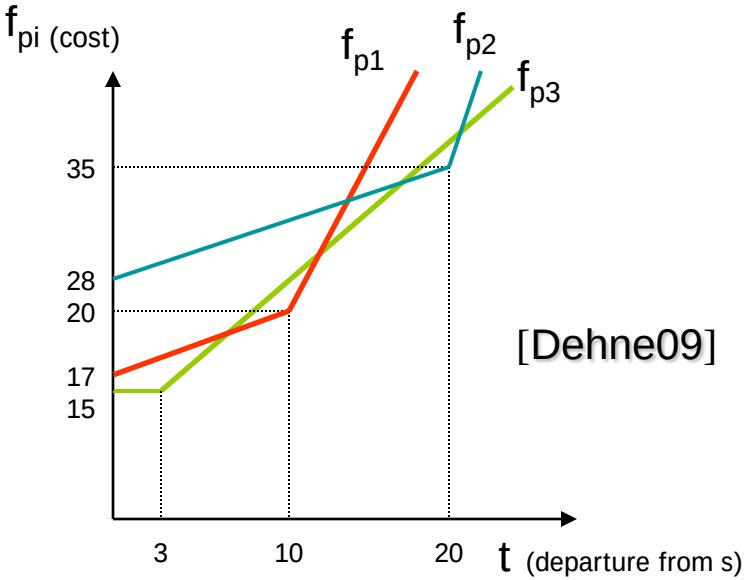


- G(V,E,T): For every edge $e(v_i; v_j)$, there is a cost function $w_{ij}(t)$ which specifies the cost of traveling from v_i to v_j at time t.
- First-In-First-Out (FIFO) Network: Moving objects exit from an edge in the same order as they entered into that edge.

Time-dependent Spatial Network (Cont)

- Path (s-d) Cost Functions
  - Composition of edge arrival functions.
  - Arrival-time (and cost) to destination is a function of departure-time from the source.





[Dehne09]

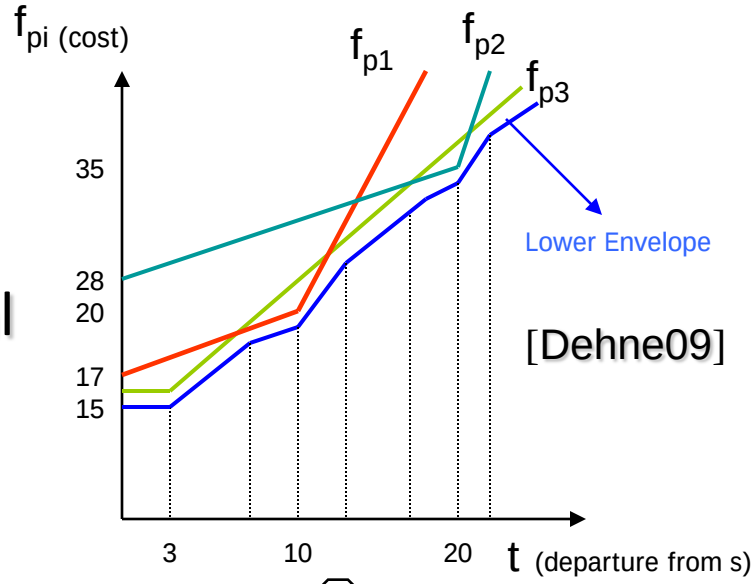
$p_1 = (v_1,v_2,v_4)$ $\quad f_{p1}=f_{24}(f_{12}(t))$

$p_2 = (v_1,v_2,v_3,v_4)$ $\quad f_{p2}=f_{34}(f_{23}(f_{12}(t)))$

$p_3 = (v_1,v_3,v_4)$ $\quad f_{p3}=f_{34}(f_{13}(t))$

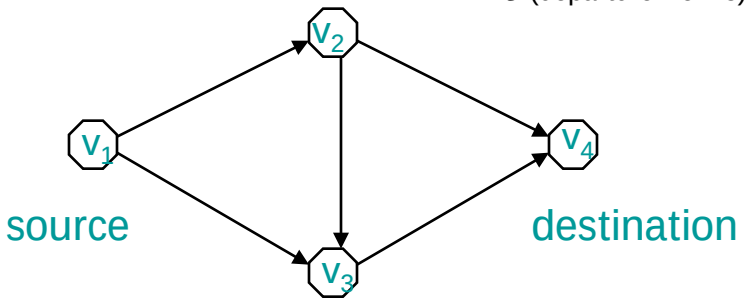
Time-dependent Spatial Network (Cont)

- Path (s-d) Cost Functions
  - Shortest path is not unique and changes based on the departure time
  - The lower-envelope of $f_p(t)$
    - Each piece gives the shortest path for the corresponding time-interval
    - Exponential pieces [Dean04] → exponential number of paths



Pre-computation

One needs to consider all possible paths (since SP is not unique) between all possible source and destination nodes

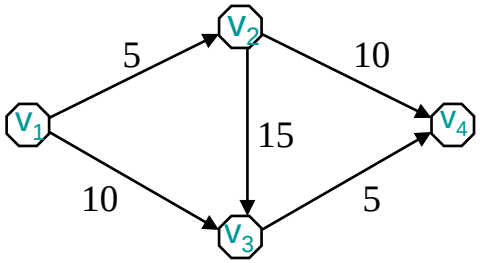


Outline

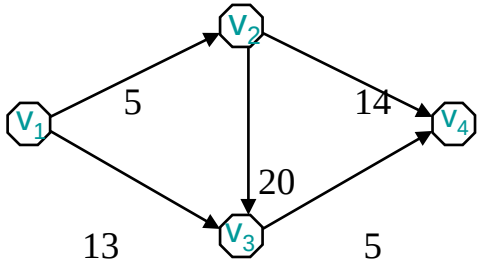
- KNN Search Past and Present
- Motivation
- Related Work
- Time-Dependent Spatial Network
- TD-kNN Search Approaches
  - Time-Expanded Networks
  - Incremental Network Expansion
  - Indexing Time-Dependent Networks
- Conclusion

TD-kNN Approach-1
Time Expanded Networks (TEN)

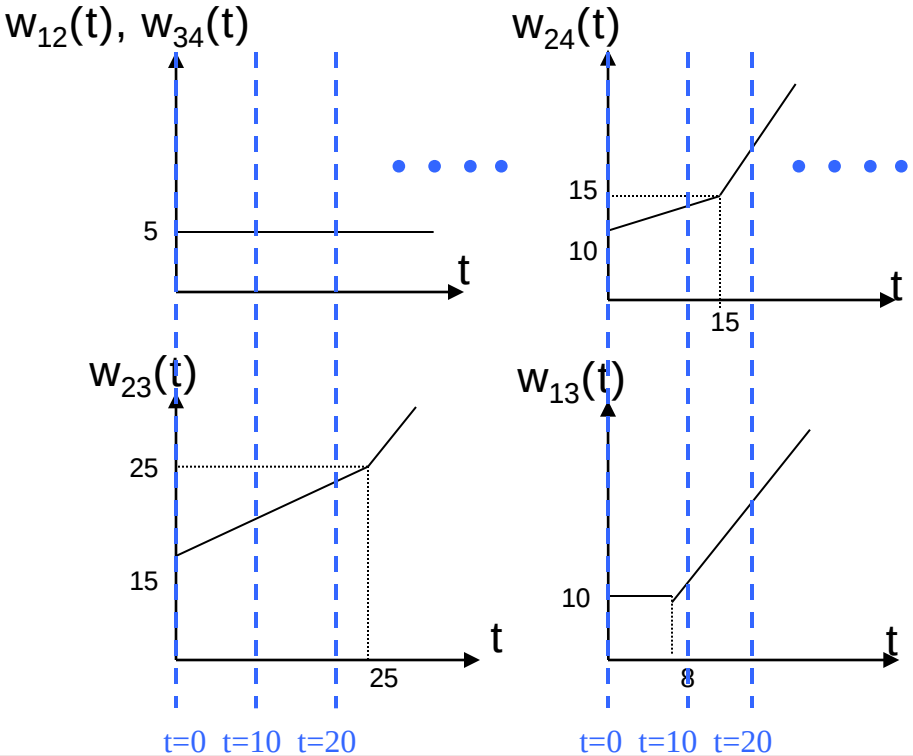
- Given G(V,E,T), discretize the time domain T into n points of time, and construct n G(V,E) graph. [Koehler02]



t=0

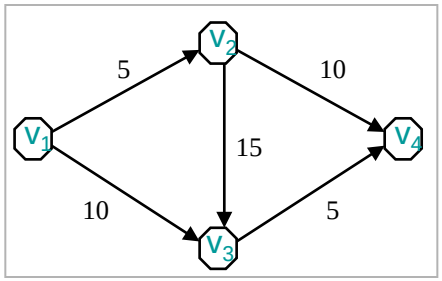


t=10

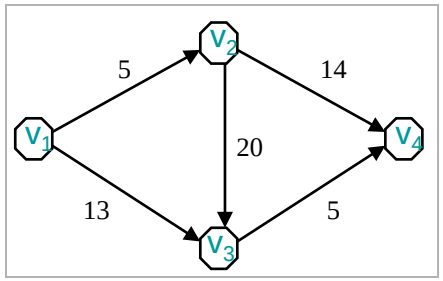


TD-kNN Approach-1
Time Expanded Networks (TEN)

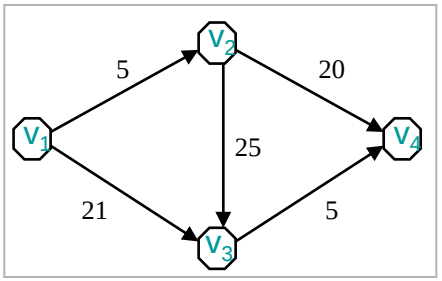
- The weight of an edge in TEN is the time difference between the time events associated with its endpoints
- A time-dependent edge cost is represented as a static flow in the corresponding TEN



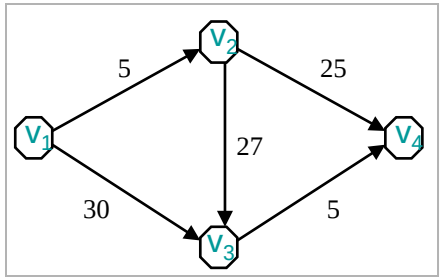
(a) t=0



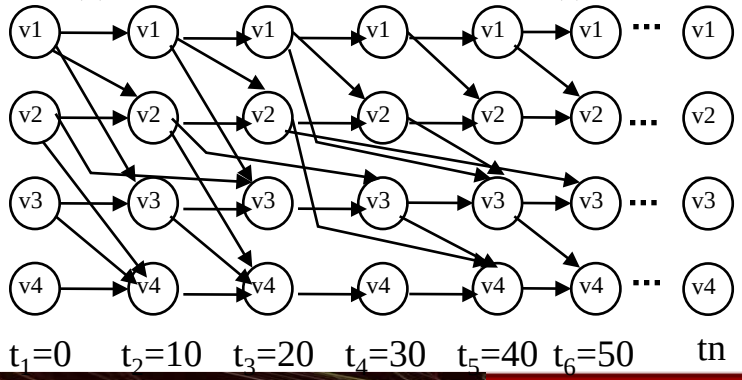
(b) t=10



(c) t=20



(d) t=30



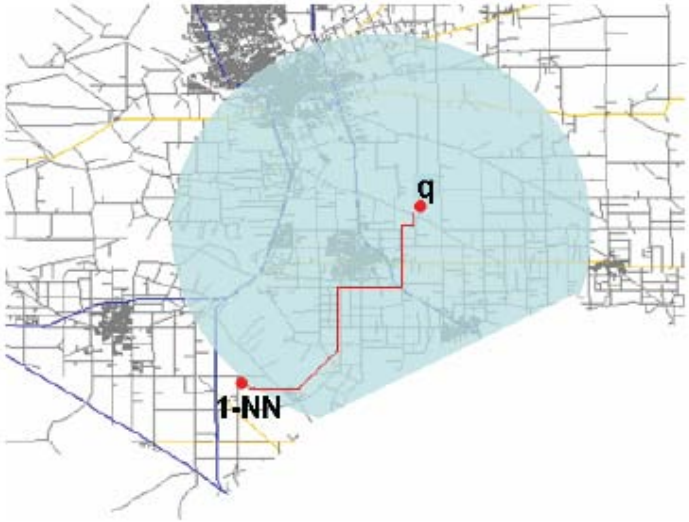
Static Network

TD-kNN Approach-1
Time Expanded Networks (TEN)

- Pros
  - Enables time-dependent kNN problem to be solved by applying techniques developed for static networks hence recomputation is possible
- Cons
  - High storage: The network size is increased proportional to the number of snapshots (n)
  - Approximate Results: The state of the network between two snapshots is not captured
    - The query time can be between two snapshots t_1 and t_2 (e.g., t=12). However, only the edge weights at t_1 or t_2 can be used, hence causing errors. The error is accumulated along the path and is unbounded.

TD-kNN Approach-2
Incremental Network Expansion (INE)

- Modified Dijkstra algorithm for time-dependent spatial networks **[Dreyfus 69]**
  - Analogous to shortest path distances, arrival-time to nodes is used as the labels that form the basis of the greedy algorithm
- Expand the network based on the arrival-time to each node around *q* until *k* objects are found

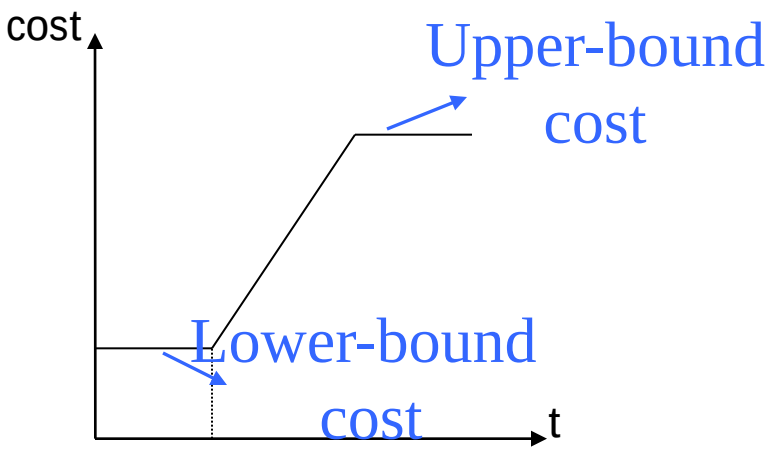


TD-kNN Approach-2
Incremental Network Expansion (INE)

- Pros
  - INE provides exact results as compared to TEN
- Cons
  - Slow response time: the overhead of network expansion is very high particularly in large networks with a sparse set of data objects, hence not applicable to online apps

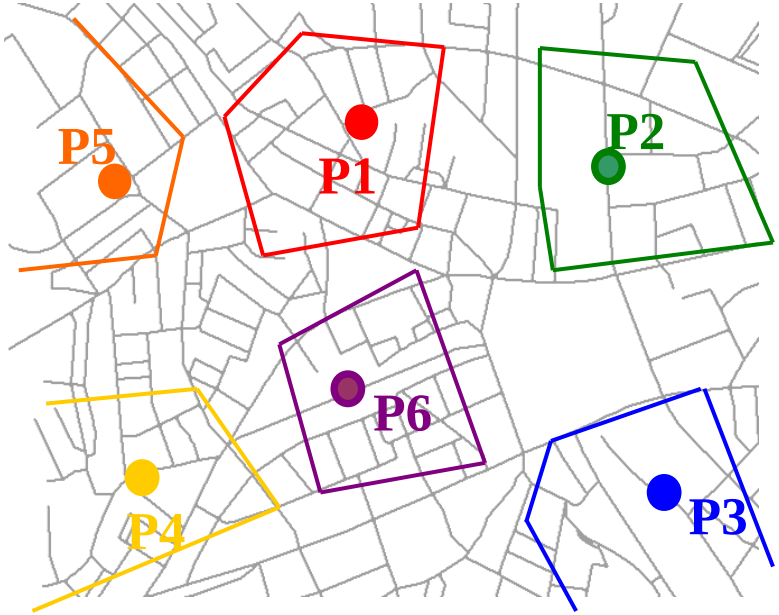
TD-kNN Approach-3
Indexing Time-Dependent Spatial Network [DEXA2010]

- Edge cost function



Lower-bound travel-time (LTT) of an edge is traversing that edge with Maximum possible speed

Tight Cells (TC)

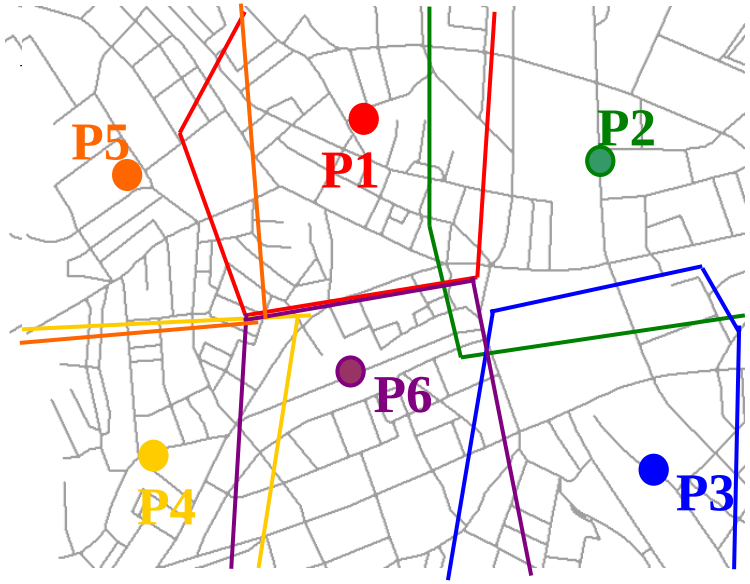


Grow SP trees from each site simultaneously using UTT for one site and LTT for the other sites

Repeat the process for all sites and find Tight Cells (TC)

TD-kNN Approach-3
Indexing Time-Dependent Spatial Network

- Loose Cells (LC)



- Loose Cells cover the entire network.
- Any query point q outside of the Loose Cell of p is guaranteed not to have p as its NN
- If q is not inside any TC, it must be inside one (or more) LC(s) and the generator of those LCs are the only NN candidates.

Grow SP trees from each site simultaneously using LTT for one site and UTT for the other sites

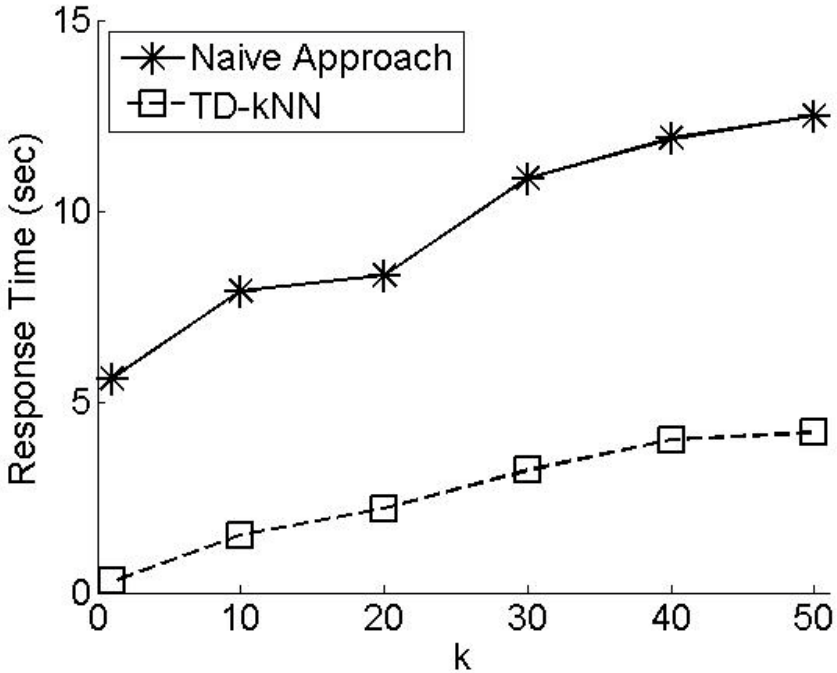
Index TCs and LCs with a spatial index (e.g., R-tree, Quad-tree) to expedite the process of finding the tight/loose cell(s) that contain q.

TD-kNN Approach-3
Indexing Time-Dependent Spatial Network

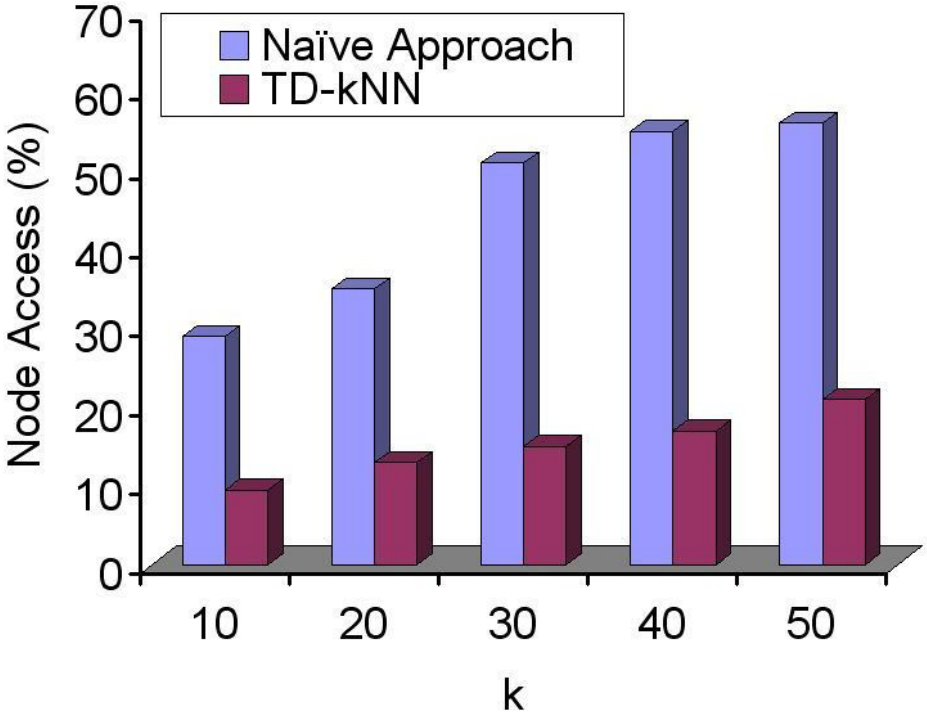
- Pros
  - Provides exact results
  - Localize the NNs and minimize the need for time-dependent SP calculation
  - Scalable and efficient for large set of query and data objects, and large networks
- Cons

Experimental Results [DEXA'10]

Naïve Approach= INE with Dreyfus's Dijkstra



k vs Response Time



k vs Network Node Access

Outline

- KNN Search Past and Present
- Motivation
- Related Work
- Time-Dependent Spatial Network
- TD-kNN Search Approaches
  - Time-Expanded Networks
  - Incremental Network Expansion
  - Indexing Time-Dependent Networks
- Conclusion

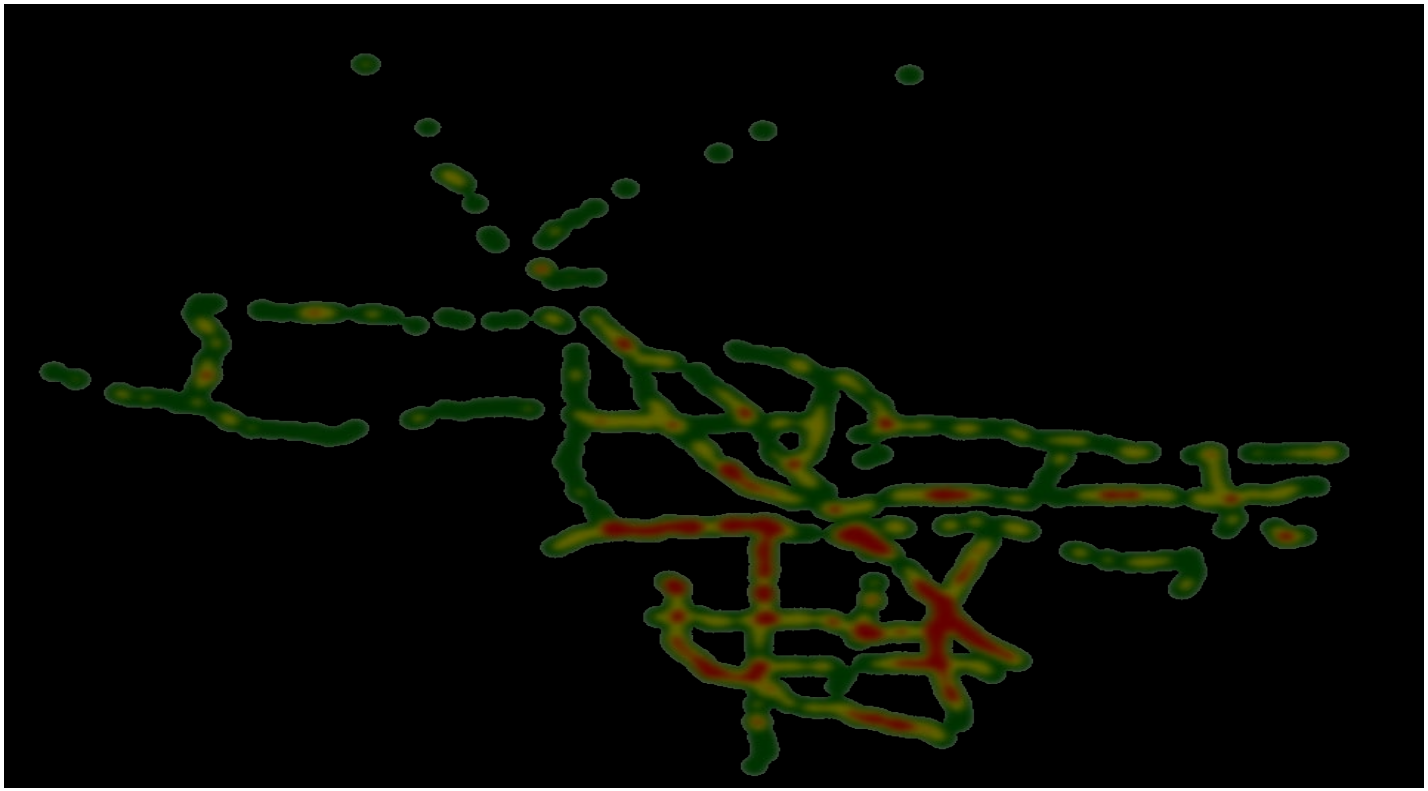
Conclusion

- In real-world edge weights are time-dependent.
  - New techniques needed to extend spatial query processing (such as kNN queries) in RN to a new family of time-dependent query processing solutions.
- Existing approaches (TEN and INE) can be extended to address time-dependent kNN; not efficient and exact.
- Indexing time-dependent network (on TCs and LCs) is an efficient and scalable approach.

OUTLINE

- Motivation: Transportation
- Platform: TransDec
- Traffic Profiles
  - Data Reduction Approach
- Applications: Time-Dependent kNN
- Demonstration
- Future Plans

GeoInsight: A framework for spatiotemporal analysis of traffic density

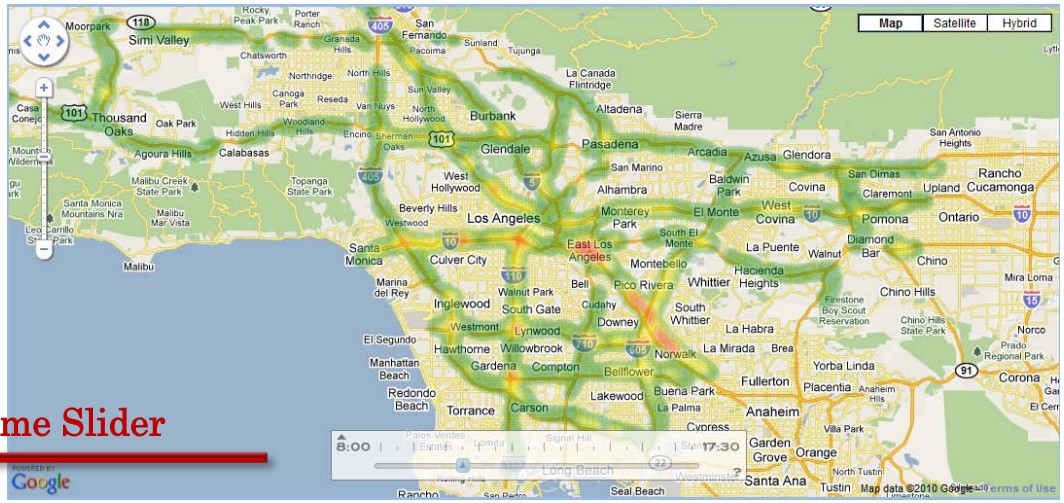


Traffic Heatmaps: Spatiotemporal analysis of traffic congestion in LA

GeoInsight: A framework for spatiotemporal analysis of traffic density

Season: Spring
Day: Monday
Time: 9:30 AM

Traffic patterns in north-east direction

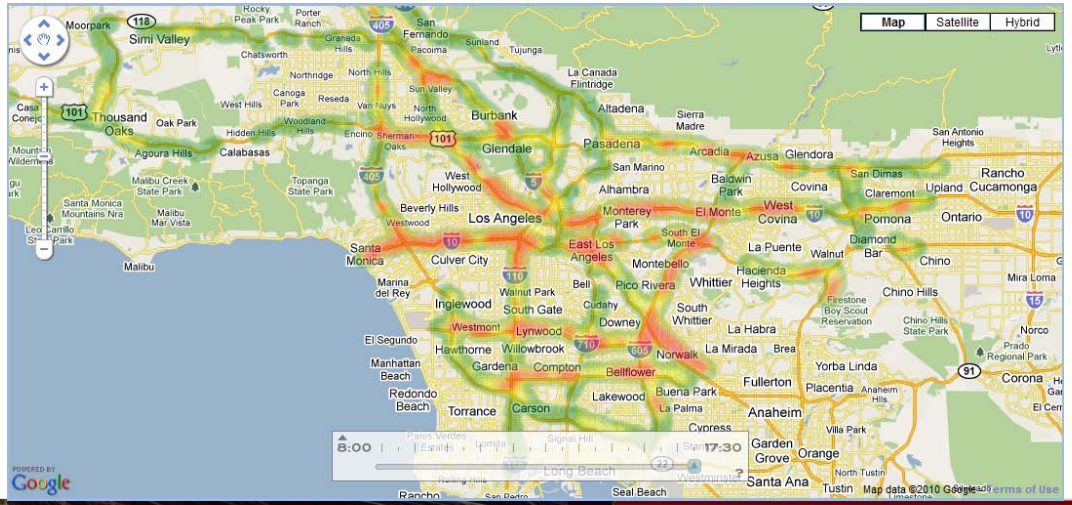


Time Slider

Navigate in time for different Season/Day/Hour traffic pattern visualization

Season: Spring
Day: Monday
Time: 5:30 PM

Traffic patterns in north-east direction



OUTLINE

- Motivation: Transportation
- Platform: TransDec
- Traffic Profiles
  - Data Reduction Approach
- Applications: Time-Dependent kNN
- Demonstration
- Future Plans

Future Work

- Data Reduction & Modeling
  - Trying other approaches: Sampling, Wavelets, …
  - Add temporal tags to spatial tags
  - Use models to generate traffic data for un-equipped roads
- Time-dependent shortest path
  - Evaluate the proposed indexing techniques
  - Other spatial queries: RkNN, spatial-skyline, …
- Closed-loop, holistic, real-time transportation optimization
  - Complementing analysis with decision-making and actuation to close the optimization loop
  - Developing scalable optimization solutions that consider the integrated transportation system as a whole
  - Introducing efficient data analysis techniques to enable on-the-fly decision making

References

- Ugur Demiryurek, Farnoush Banaei-Kashani, and Cyrus Shahabi, Efficient K-Nearest Neighbor Search in Time-Dependent Spatial Networks, 21st International Conference on Database and Expert Systems Applications (DEXA10), Bilbao, Spain, August 2010
- Ugur Demiryurek, Bei Pan, Farnoush Banaei-Kashani, and Cyrus Shahabi, Towards Modeling the Traffic Data on Road Networks, ACMGIS 2009 Workshop on Computational Transportation Science , Seattle, Washington, November 2009
- L. Cooke and E. Halsey. *The shortest route through a network with time-dependent intermnodal transit times*. In JMA, 1966.
- P. Dreyfus. *An appraisal of some shortest path algorithms*. In JOR, 1969.
- E.Köhler, K. Langkau and M. Skutella. *Time-Expanded graphs for flow-dependent transit times*. In ESA, 2002
- B. George, S. Kim, and S. Shekhar. *Spatio-temporal network databases and routing algorithms: A summary of results*. In SSTD, 2007.
- B.C. Dean. *Shortest paths in time-dependent FIFO networks: Theory and Algorithms*. MIT Technical Report, 2004.
- F. Dehne, M.T. Omran, and J. Sack. *Shortest paths in time-dependent FIFO networks using edge load forecasts*. ACMGIS-IWCTS, 2009
- J. Halpern. *Shortest route with time dependent length of edges and limited delay possibilities* .In MMO, 1969.
- X. Huang, C. S. Jensen, and S. Saltenis. S-grid: *A versatile approach to efficient query processing in spatial networks*. In SSTD, 2007.
- M. Kolahdouzan and C. Shahabi. *Voronoi-based k nn search in spatial networks*. In VLDB, 2004.
- K. Mouratidis, L. Yiu, D. Papadias and N. Mamoulis. *Continuous nearest neighbor monitoring in road networks.* In VLDB, 2006
- A. Orda and R. Rom. *Shortest-path and minimum-delay algorithms in networks with time-dependent edge-length*. J ACM, 1990.
- D. Papadias, J. Zhang, N. Mamoulis, and Y. Tao. *Query processing in spatial networks.* In VLDB,2003.
- H. Samet, J. Sankaranarayanan, and H. Alborzi. *Scalable network distance browsing in spatial databases*. In SIGMOD, 2008.

Thanks!