Spatio-Temporal Routing Algorithms

Panel on Space-Time Research in GIScience

Intl. Conference on Geographic Information Science 2012

Shashi Shekhar

McKnight Distinguished University Professor

Department of Computer Sc. and Eng., University of Minnesota

Presented by

Dr. Budhendra Bhaduri

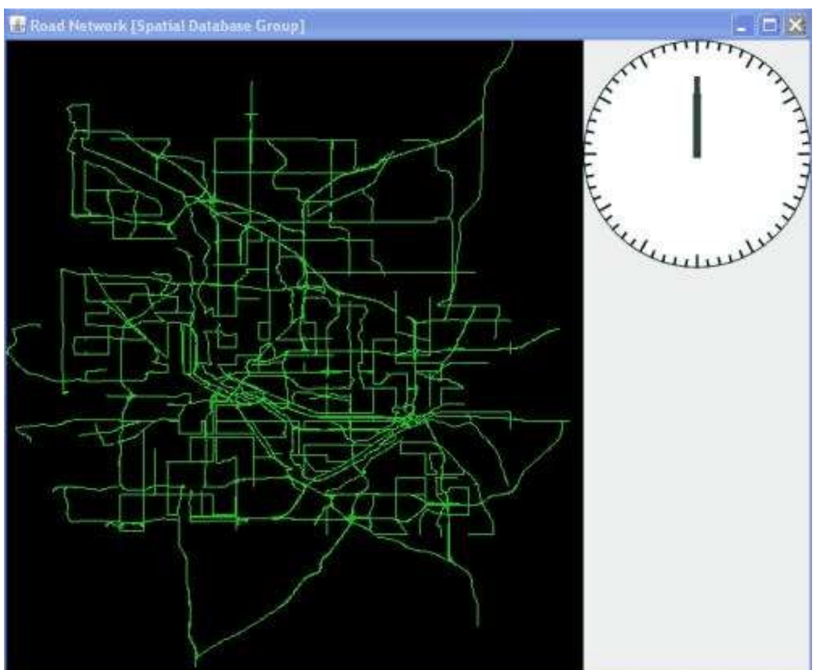
Leader, Geographic Information Science & Technology

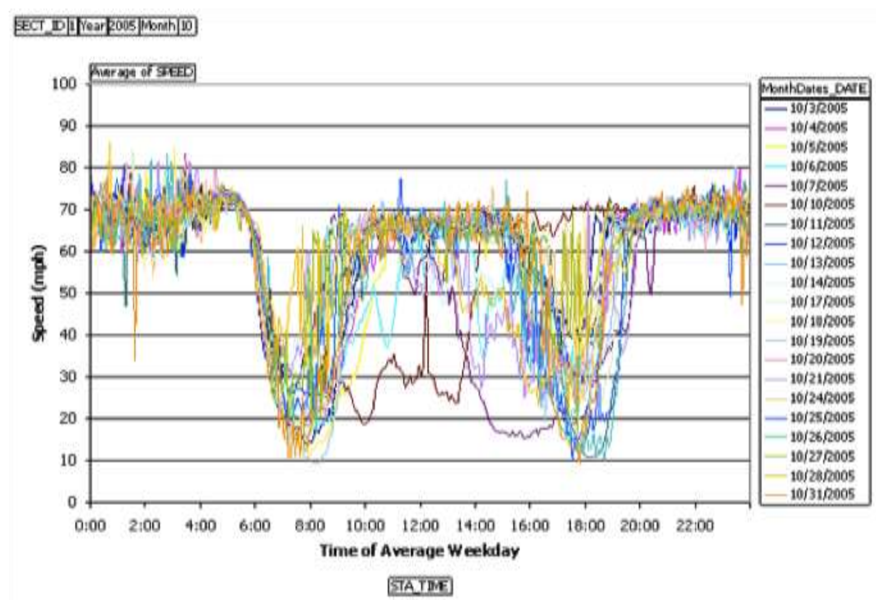
Oak Ridge National Laboratory

Dynamic Nature of Transportation Network

Traffic during non-rush hours

Traffic during Rush hours





Implication of Dynamic Nature

The New York Times

U.P.S. Embraces High-Tech Delivery Methods (July 12, 2007)

*By "The research at U.P.S. is paying off. ....— **saving roughly three million gallons** of fuel in good part by mapping routes that minimize left turns."*

Problem 1: Time-dependent network models

- Input：
  a) A Spatial Network
  b) Temporal changes of the network topology and parameters.

- Output： A model that supports efficient correct algorithms for computing the query results.

- Objective：Minimize storage and computation costs.

- Constraints：
  (i) Predictable future
  (ii) Changes occur at discrete instants of time,
  (iii) Logical & Physical independence,

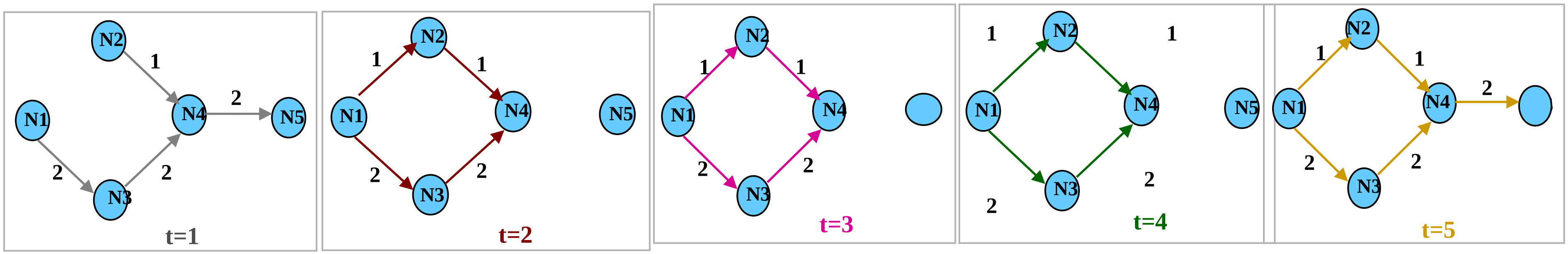
Challenges in Representation

- Conflicting Requirements
  - Expressive power
  - Storage efficiency
- New Semantics for network concepts
  - Lagrangian shortest paths
  - Time dependence of shortest paths
  - Best start-time paths
- Violates assumptions behind algorithms
  - Prefix optimality, stationary ranking of alternative paths
  - Dijkstra's, A*, Dynamic Programming

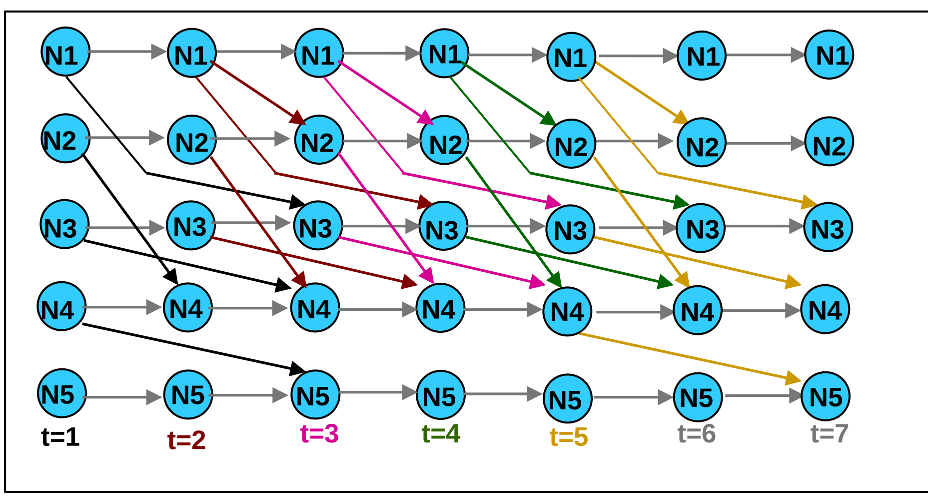
Representations of (Spatio-)temporal Networks

(1) Snapshot Model [Guting 04]

Node: N.. Edge: Travel time



(2) Time Expanded Graph (TEG) [Ford 65]

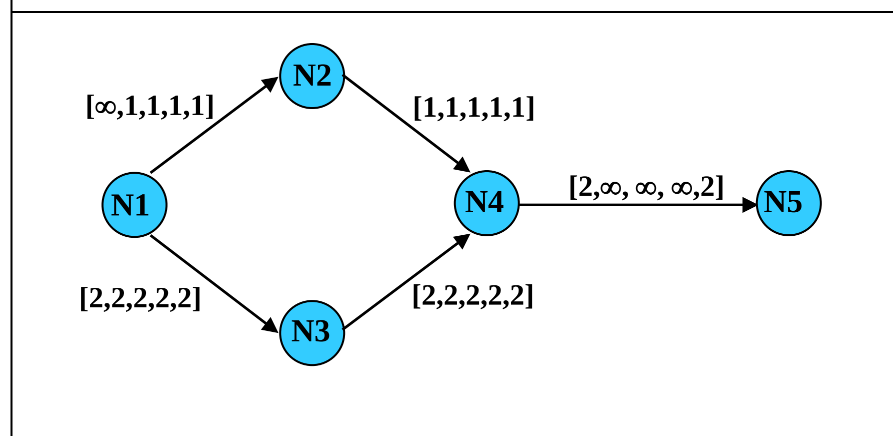


Holdover Edge

Transfer Edges

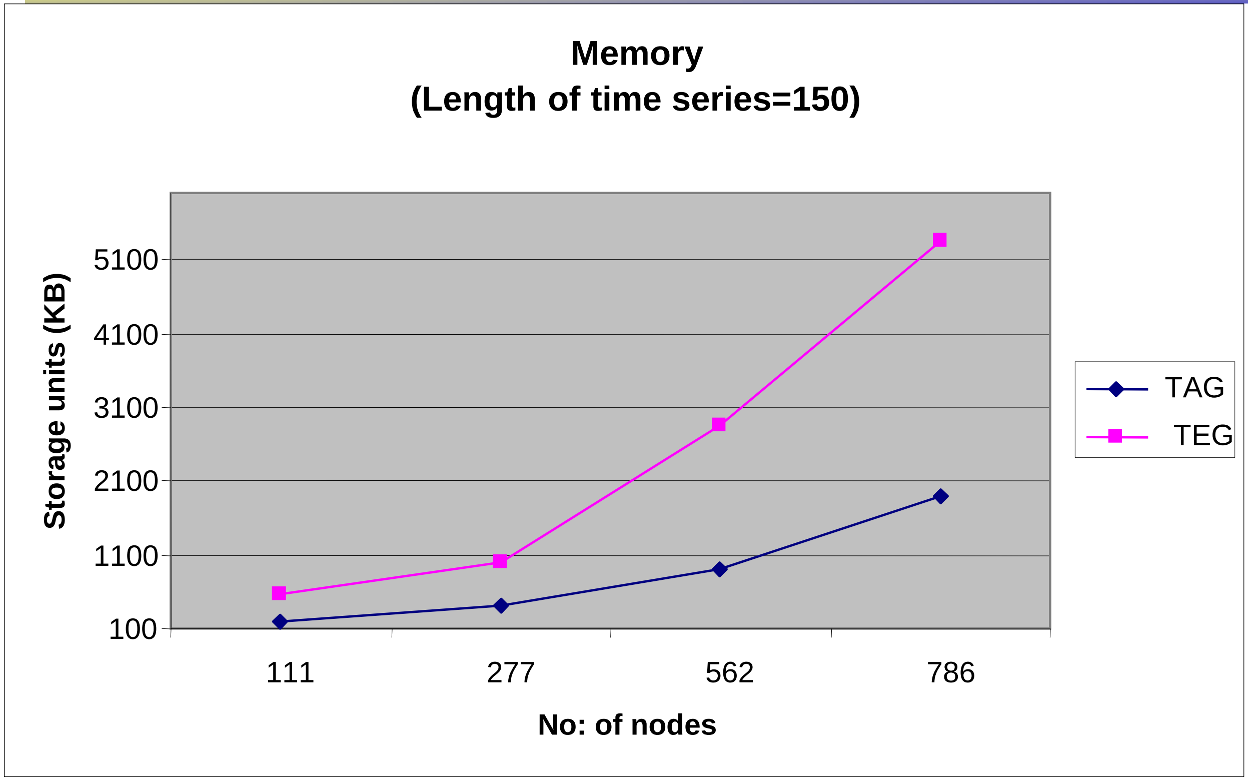
(3) Time Aggregated Graph (TAG) [Our Approach]

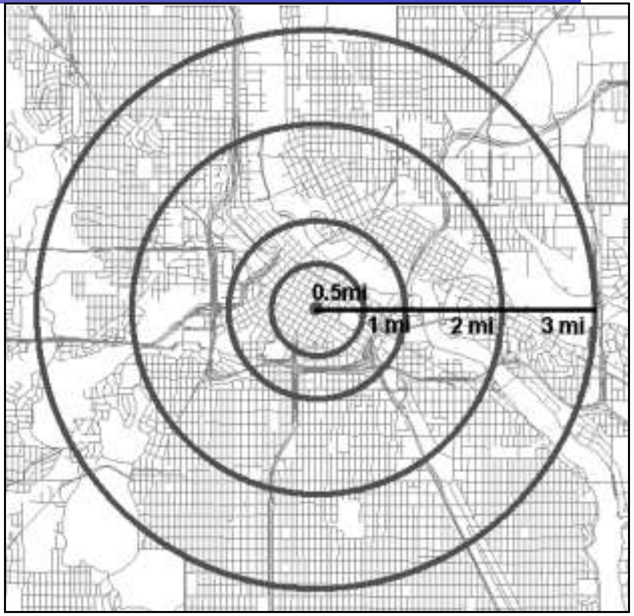
¤ Attributes aggregated over edges and nodes.



Edge $\xrightarrow{[m_1,\ldots,(m_T]}$ m_i- travel time at t=i

TAG vs. TEG: Storage Cost Comparison





Minneapolis CBD
[1/2, 1, 2, 3 miles radii]

Dataset	# Nodes	# Edges
(MPLS -1/2)	111	287
(MPLS -1 mi)	277	674
(MPLS - 2 mi)	562	1443
(MPLS - 3 mi)	786	2106

Trend: TAG better than TEG
on storage overhead!

TAG compared to Related Work

- TAG has lower storage cost
  - No replication of nodes and edges across time-frames
  - Allows sharing/compression of time-series
- TAG leads to faster and scalable algorithms
  - Smaller representation
  - TAG transformations, partitions, …
- Relative to TEG,
  - Provides logical-physical independence
  - Can model properties beyond travel-time

Problem 2: ST Shortest Path Algorithms

- **Input** :
  a) A Spatial Network
  b) Time-series edge-weights.
  c) An (origin, destination) pair
  d) A start time
- **Output** : A spatio-temporal route (and schedule)
- **Objective** : Minimize route cost (e.g., travel-time or fuel consumed)
- **Constraints** :
  (i) Predictable future
  (ii) Changes occur at discrete instants of time,

Challenges

Non Stationary ranking of paths

Time	Preferred Routes
7:30am	Via Hiawatha
8:30am	Via Hiawatha
9:30am	via 35W
10:30am	via 35W

➢ **Violation of stationary assumption dynamic programming**

Non FIFO Behavior

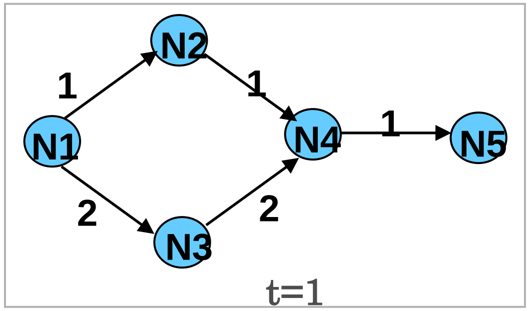
Time	Route	Flight Time
8:30am	via Detroit	6 hrs 31 mins
9:10am	direct flight	2 hrs 51 mins
11:00am	via Memphis	4 hrs 38mins
11:30am	via Atlanta	6 hrs 28 mins
2:30pm	direct flight	2 hrs 51 mins

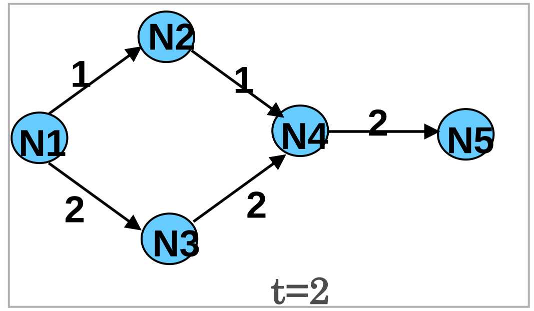
***Flight schedule between Minneapolis and Austin (TX)**

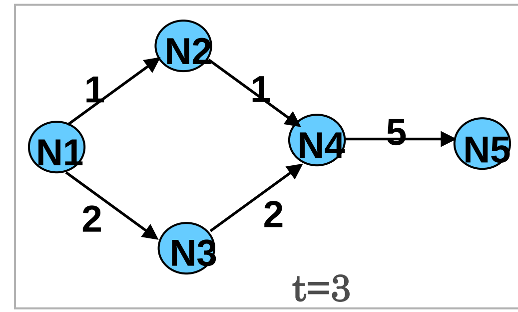
➢ **Violates the no wait assumption of Dijkstra/A***

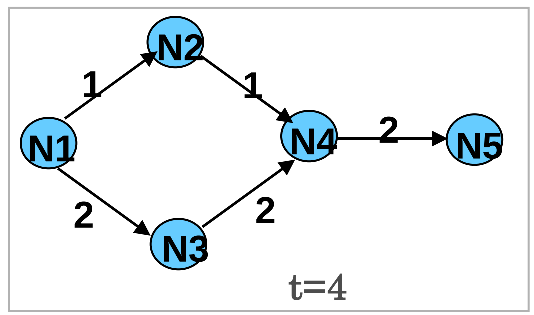
Dealing with non-FIFO edges using Waits

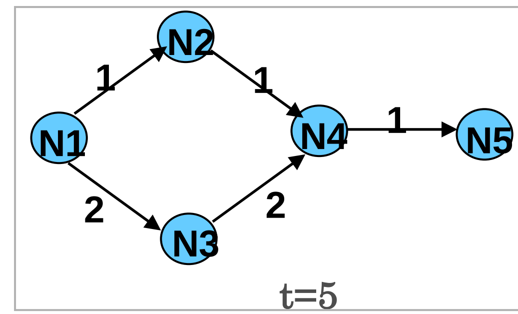
Find the shortest path travel time from N1 to N5 for start time t = 1.











	N1	N2	N3	N4	N5
1	~~1~~	∞	∞	∞	∞
2	1	~~2~~	3	∞	∞
3	1	2	~~3~~	3	∞
4	1	2	3	~~3~~	∞
5	1	2	3	3	8

Optimal path: Reach N4 at t=3;

Wait for t=4;

Reach N5 at t=6

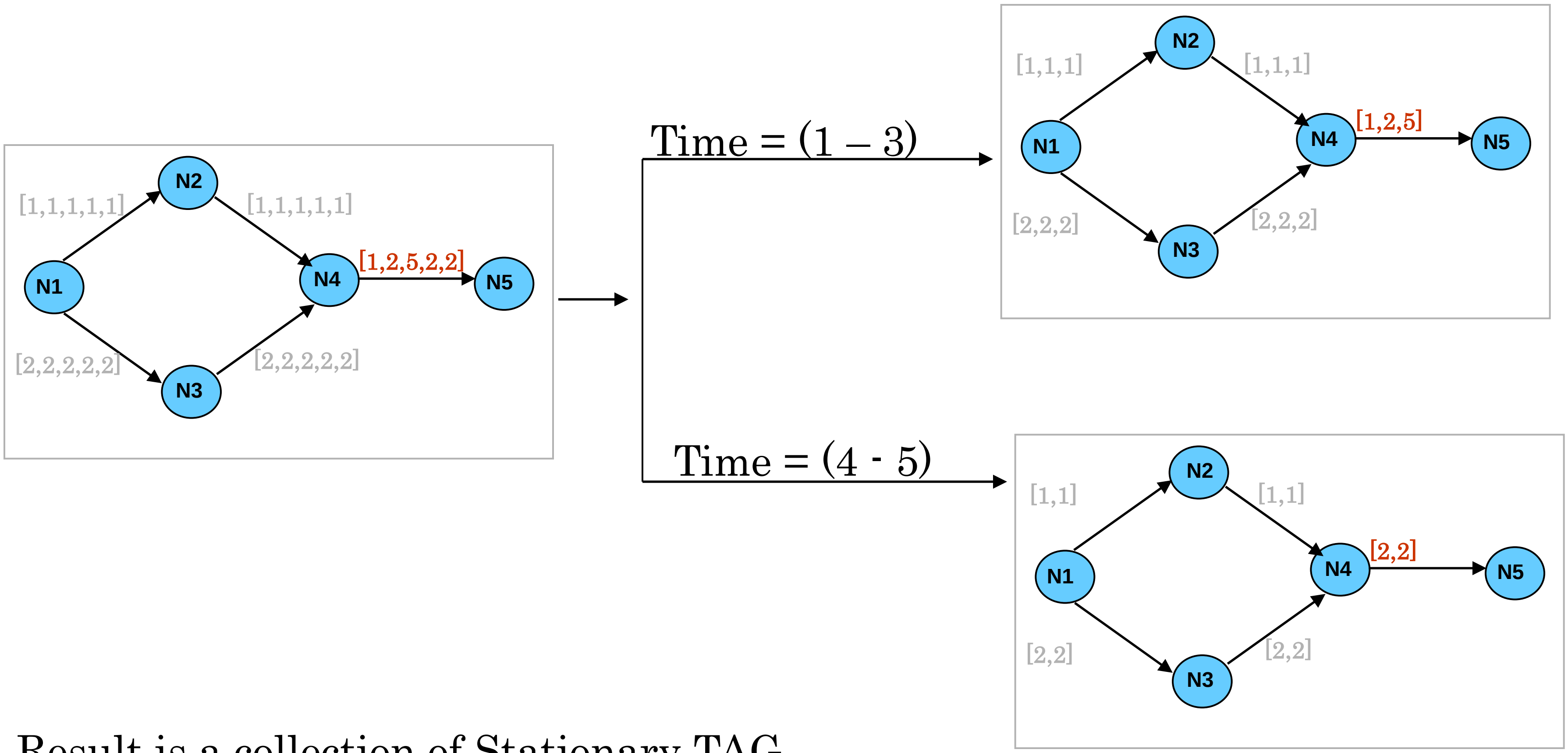
Total time = 5

Dijktra's algo.: Reaches N5 at t=8.

Total time = 7

Dealing with non-stationary ranking of routes

Idea: Divide into time-intervals with stationary ranking of routes



Result is a collection of Stationary TAG.

Dynamic programming may be used within each sub-TAG !

Summary: ST Routing Algorithms

Predictable Future → Stationary — Dijkstra's, A*….

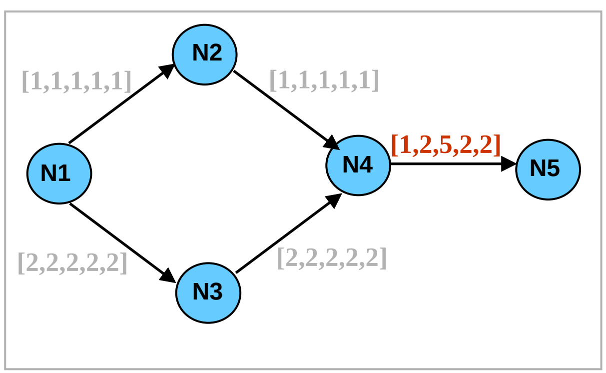
Predictable Future → Non-stationary → Special case (FIFO)

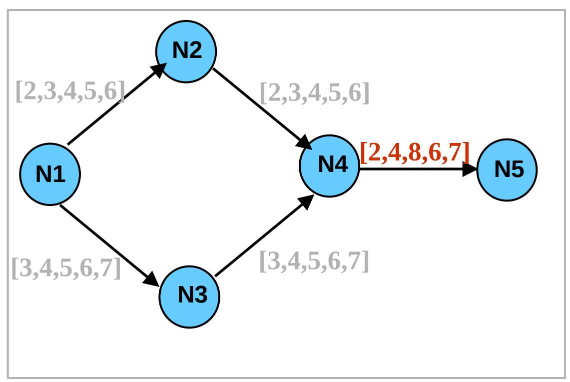
Predictable Future → Non-stationary → General Case — TEG: LP, Label-correcting

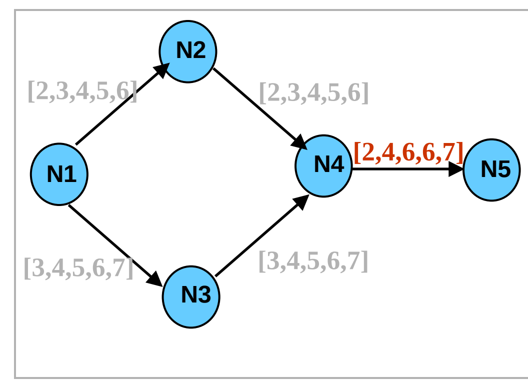
TAG: Transform to Stationary TAG

Unpredictable Future

travel times → arrival times at end node → Min. arrival time series







Non-stationary TAG → Stationary TAG

More ST Shortest Path Problems

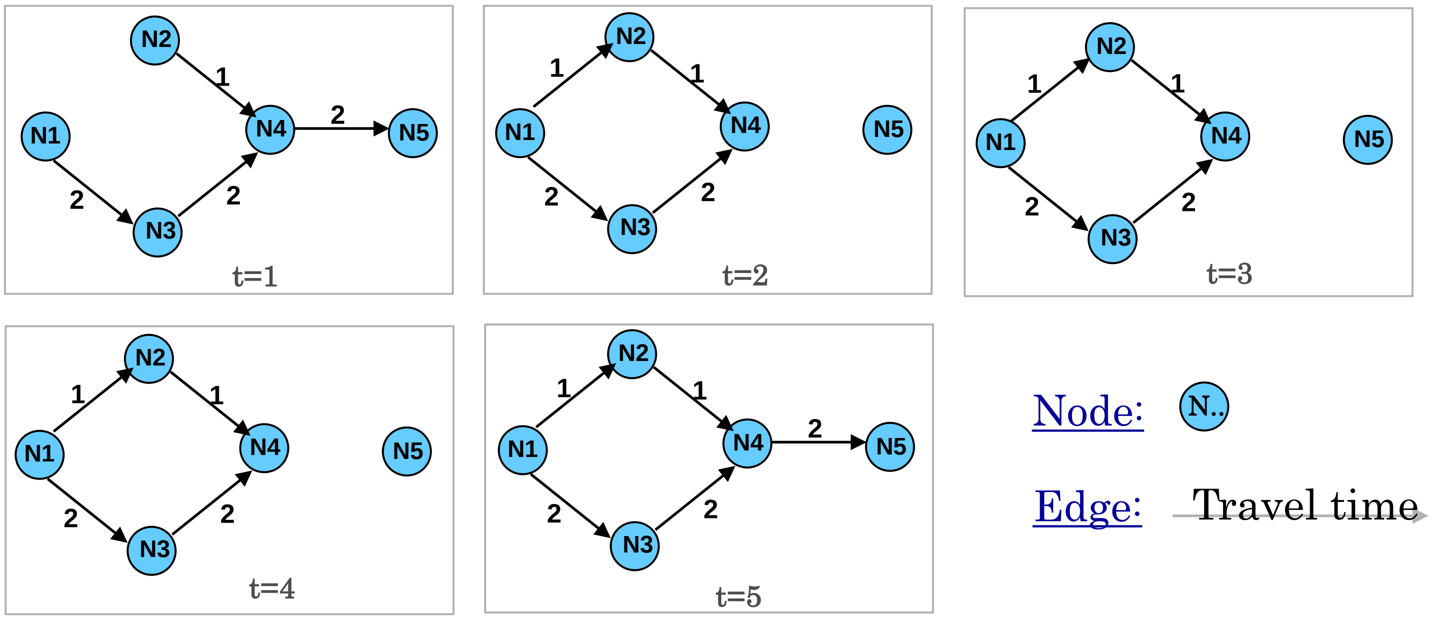
Static	Time-Variant
Which is the shortest travel time path from downtown Minneapolis to airport?	Which is the shortest travel time path from downtown Minneapolis to airport at different times of a work day?
What is the capacity of Twin-Cities freeway network to evacuate downtown Minneapolis ?	What is the capacity of Twin-Cities freeway network to evacuate downtown Minneapolis at different times in a work day?

- New Routing Questions
  - Best start time to minimize time spent on network
  - Account for delays at signals, rush hour, etc.

Dealing with new Semantics, e.g., Best start time

*Identify **best start-time** for travel from N1 to N5.,*

if Shortest Path is dependent on start time!!



Start at t=1:

Shortest Path is N1-N3-N4-N5;
Travel time is **6** units.

Start at t=3:
Shortest Path is N1-N2-N4-N5;
Travel time is **4** units.

Best Start Time is 3