8th Workshop on Safety and Security in Intelligent Vehicles (SSIV), with the 52nd IEEE/IFIP Dependable Systems and Networks (DSN), 2022

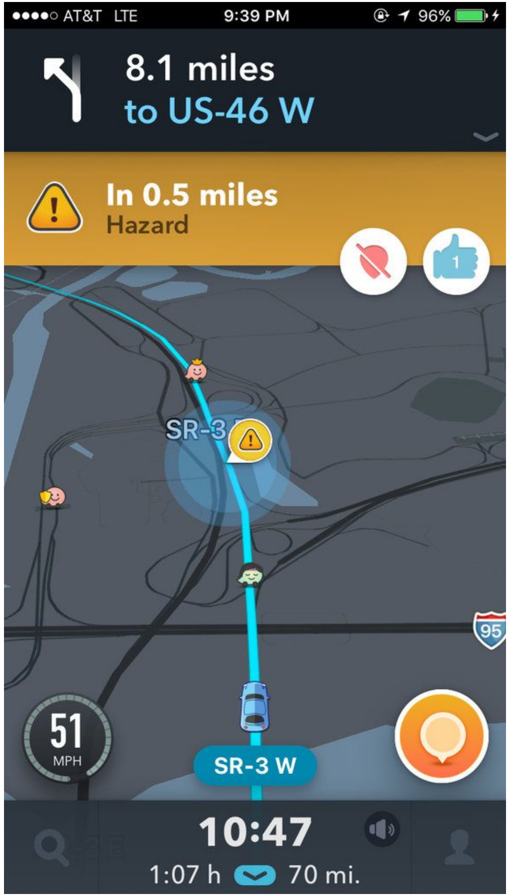
Alternative Route-Based Attacks in Metropolitan Traffic Systems

Sidney La Fontaine*, Naveen Muralidhar*,
Michael Clifford**, Tina Eliassi-Rad*, and Cristina Nita-Rotaru*

*Khoury College of Computer Science, Northeastern University
**Toyota Info Tech

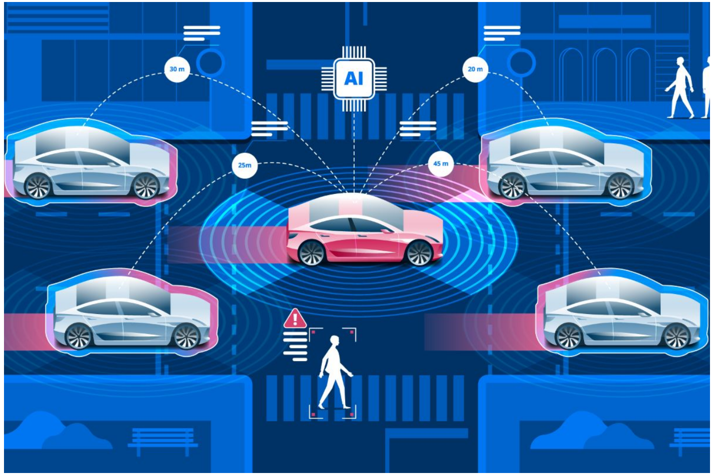
Metropolitan Traffic Systems

- Complexity
  - Los Angeles has 51,716 intersections and 141,992 roads
- Reliance on driving direction applications that reroute drivers for optimal travel times



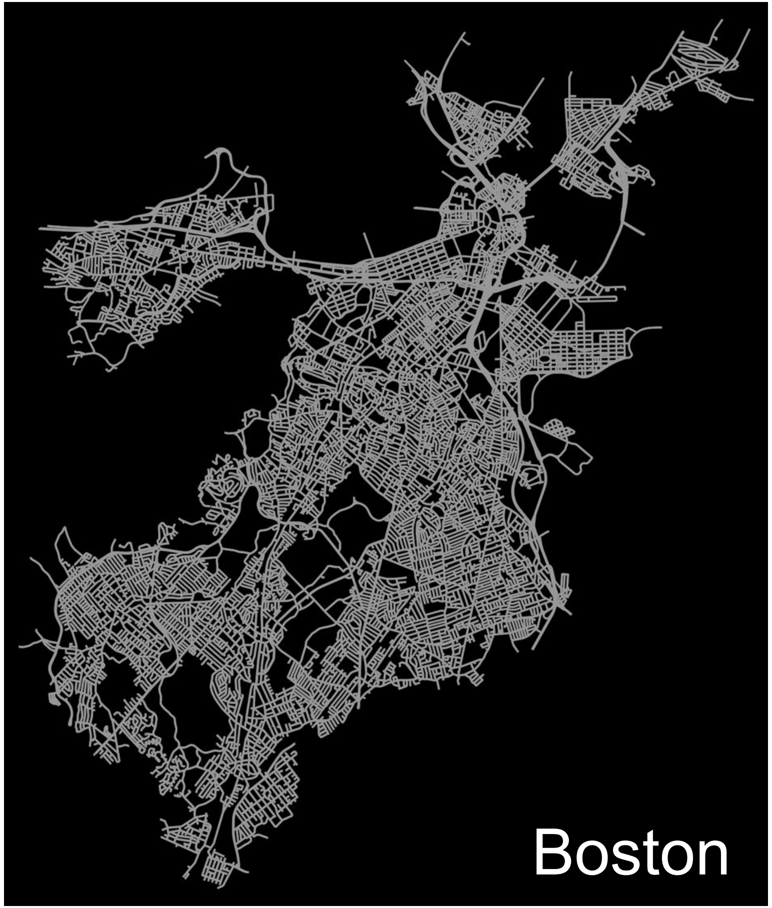
The Future is CAV

- Connected and Autonomous Vehicles (CAVs) are beginning to take over driving
- New promises of comfort, convenience, and safety
- Increased dependence on software introduces **new attack vectors**



Connectivity Attacks

- Traffic map
  - Target communities
  - Target bridges
  - Target a victim
- Attacks can be formulated as graph problems



This Talk

Is it possible to force drivers to take chosen slower alternative routes in cities by blocking roads?

What is the cost incurred by such attacks?

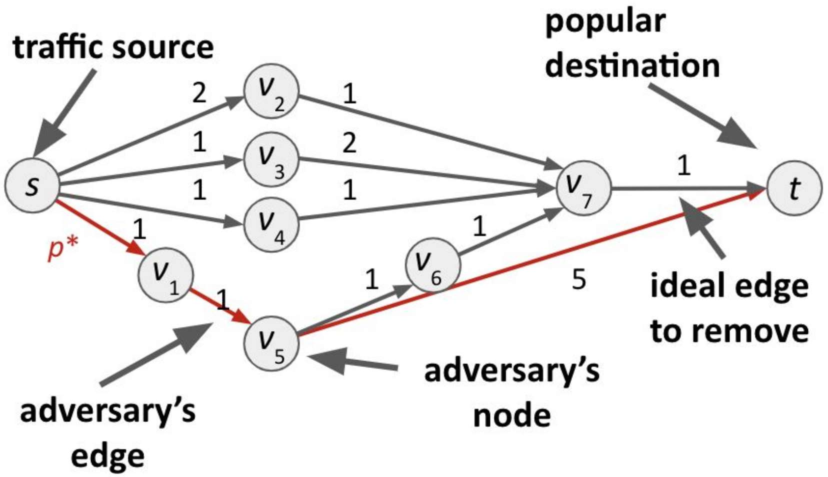
What are the conditions that influence such attacks?

Attacker Model

- Can block roads or make them unusable
- Has a budget
- Has publicly available information about the road map and location of target

Alternative Route-Based Attack

- Given source, destination, travel times, road removal costs, chosen slower alternative route (p*), and budget
- Find cheapest set of roads to remove such that in resulting graph the quickest path between source and destination is p*
- Victim will travel the shortest route to destination



[PATHATTACK: Attacking Shortest Paths in Complex Networks](#)
B.A. Miller, Z. Shafi, W. Ruml, Y. Vorobeychik, Tina Eliassi-Rad, and S. Alfeld, ECML PKDD 2021.

Solutions

- Problem is NP-complete, Tradeoff between runtime and performance
- Optimal solution
  - **LP-PathCover**: Linear Programming optimization approach
- Heuristics that scale better, proceed iteratively:
  - **GreedyEdge**: Cuts the lowest cost edge on the current shortest path
  - **GreedyEig**: Cuts the edge with the highest eigenvalue to cost ratio on the current shortest path
  - **GreedyPathCover**: Cuts the edge that removes the most paths shorter than the chosen alternative path

PATHATTACK: Attacking Shortest Paths in Complex Networks B.A. Miller, Z. Shafi, W. Ruml, Y. Vorobeychik, Tina Eliassi-Rad, and S. Alfeld, ECML PKDD 2021.

Experimental Methodology

- Modeling city transportation maps
- Selecting targets
- Modeling attack costs and distances
- Metrics

Modeling City Transportation Maps

- Graph representation of cities from OpenStreetMap
- Metadata: road length, speed limit, number of lanes, etc.

CITY GRAPH SUMMARIES

City	Nodes	Edges	Avg. Node Degree
Boston	11171	25715	4.60
San Francisco	9659	269002	5.57
Chicago	29299	78046	5.33
Los Angeles	51716	141992	5.08

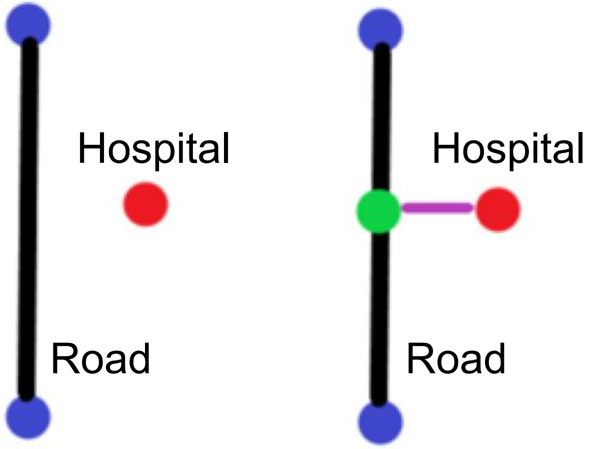
	osmid	oneway	lanes	highway	maxspeed	length	geometry	name	width	ref	bridge	access	tunnel	junction	u	v
0	197230699	True	2	residential	25 mph	33.072	LINESTRING (-71.02182 42.36761, -71.02178 42.3...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	30730954	61441677
1	197230701	False	NaN	residential	25 mph	278.886	LINESTRING (-71.02182 42.36761, -71.02185 42.3...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	30730954	1102741801
2	8603503	False	2	residential	25 mph	65.327	LINESTRING (-71.09390 42.38221, -71.09354 42.3...	Munroe Street	15.2	NaN	NaN	NaN	NaN	NaN	61151272	61151274
3	8603503	False	2	residential	25 mph	171.080	LINESTRING (-71.09390 42.38221, -71.09396 42.3...	Munroe Street	15.2	NaN	NaN	NaN	NaN	NaN	61151272	71953402
4	172307046	False	NaN	residential	25 mph	76.432	LINESTRING (-71.09390 42.38221, -71.09386 42.3...	Bigelow Street	NaN	NaN	NaN	NaN	NaN	NaN	61151272	71921695

OpenStreetMap

Selecting Targets

- Converted OpenStreetMap datasets and metadata into NetworkX Directed Graph
  - Vertices represent the start and end of roads, intersections, tolls, traffic signals, points of interest, etc.
  - Edges represent one-way roads with multiple properties including length, speed limit, number of lanes, etc.
- Added targets to city graphs
  - Hospitals

NetworkX
Network Analysis in Python



Modeling Attack Costs and Distances

- Cost of removing an edge
  - UNIFORM: all roads have same cost
  - LANES: cost assigned based on number of lanes
  - WIDTH: cost assigned based on width of the road
- Distance of an edge:
  - LENGTH: distance assigned based on length of the road
  - TIME: distance assigned based on time to travel a road at its speed limit

Metrics

- **Average Algorithm Runtime** (Avg. Runtime): average time algorithm over 40 experiment sets
- **Average Number of Edges Removed** (ANER): average number of road segments removed to ensure p* is the shortest path between source and destination, over 40 experiment sets
- **Average Cost of Removed Edges** (ACRE): average cost of road segments removed to ensure p* is the shortest path between source and destination, over 40 experiment sets

Impact of Algorithms

- GreedyPathCover most effective without taking too long
- LP-PathCover has longest runtime
- GreedyEgde and GreedyEig quickest and most expensive

BOSTON, WEIGHT TYPE: TIME

Algorithm	UNIFORM			LANES			WIDTH		
	Avg. Runtime	ANER	ACRE	Avg. Runtime	ANER	ACRE	Avg. Runtime	ANER	ACRE
LP-PathCover	66.82	3.78	3.78	21.17	4.18	6.6	19.56	3.58	7.48
GreedyPathCover	5.76	3.78	3.78	4.25	4.15	6.55	4.33	3.58	7.48
GreedyEdge	2.02	4.65	4.65	1.56	4.48	6.9	1.66	4.38	9.16
GreedyEig	3.22	4.65	4.65	2.77	4.48	8.33	2.92	4.4	9.21

Impact of Cities

- More lattice-like cities: baseline algorithms capable of finding close to optimal cost
- Less lattice cities: larger gap in cost between baseline and intelligent algorithms

Chicago: more lattice

Algorithm	UNIFORM		
	Avg. Runtime	ANER	ACRE
LP-PathCover	41.38	3.5	3.5
GreedyPathCover	8	3.5	3.5
GreedyEdge	1.51	4.1	4.1
GreedyEig	2.12	4.5	4.5

Boston: less lattice

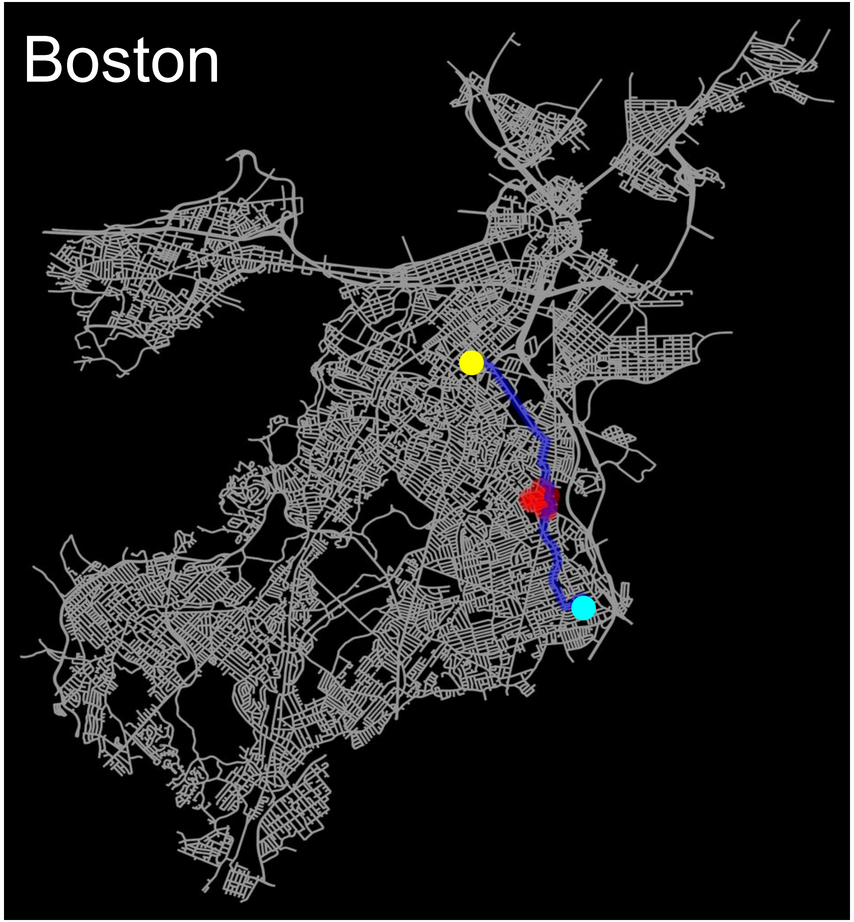
Algorithm	LANES		
	Avg. Runtime	ANER	ACRE
LP-PathCover	58.31	3.75	5
GreedyPathCover	6.72	3.78	5.03
GreedyEdge	3.78	5.25	6.5
GreedyEig	4.99	4.65	7.65

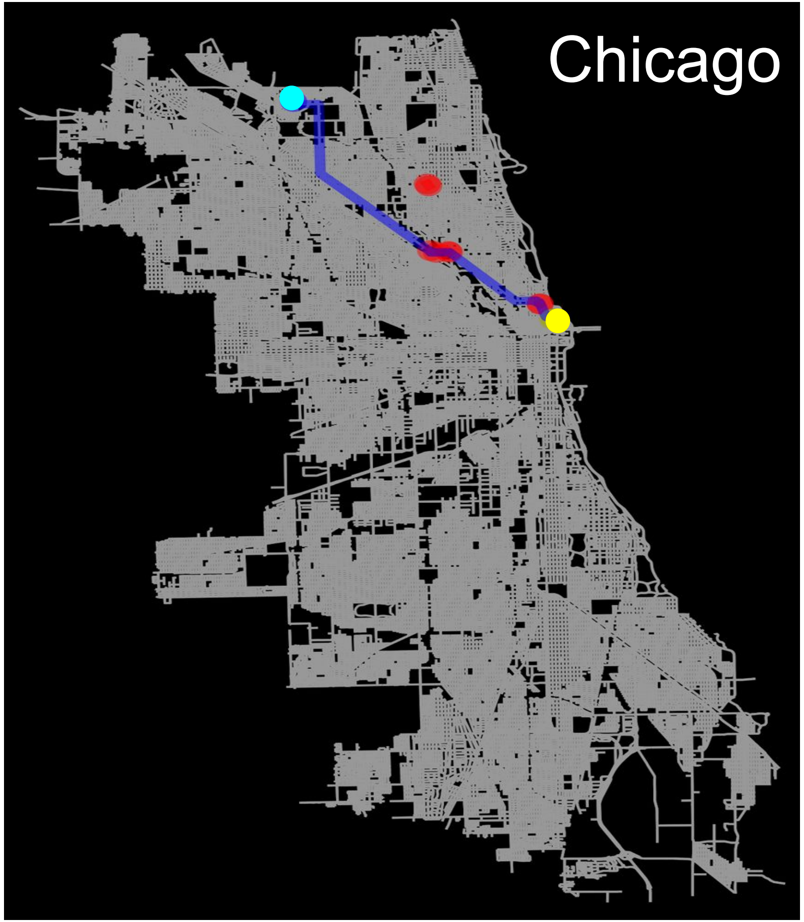
Impact of Cost

- Cost increases from UNIFORM to LANES to WIDTH
- Attacker can choose option based on their ability to cause interruptions

SAN FRANCISCO, WEIGHT TYPE: LENGTH

Algorithm	UNIFORM			LANES			WIDTH		
	Avg. Runtime	ANER	ACRE	Avg. Runtime	ANER	ACRE	Avg. Runtime	ANER	ACRE
LP-PathCover	37.4	3.68	3.68	85.35	4.18	5.38	48.4	3.65	7.64
GreedyPathCover	6.44	3.68	3.68	5.81	4.43	5.68	5.74	3.65	7.65
GreedyEdge	2.2	6.58	6.58	2.14	7.5	8.45	2.33	6.28	13.13
GreedyEig	3.6	5.78	5.78	3.35	5.93	8.58	3.56	5.05	10.57


Boston


Chicago

Summary

- Defined Alternative Route-Based Attack
- Found that GreedyPathCover was the most efficient algorithm
- Naive baseline algorithms found near optimal solutions on more lattice cities
- Less lattice cities required more intelligent algorithms to find cost effective solutions