A Content-Centric Network for Autonomous Driving

Swarun Kumar

Lixin Shi, Stephanie Gil, Nabeel Ahmed,
Dina Katabi and Daniela Rus

Much Interest in Autonomous Vehicles

Google's Autonomous Car

- Benefits include lower traffic congestion, higher fuel efficiency, improved productivity
- Projected to save **$100B/yr** in US alone [WPI'07]

Robots for Disaster Areas (Fukushima)

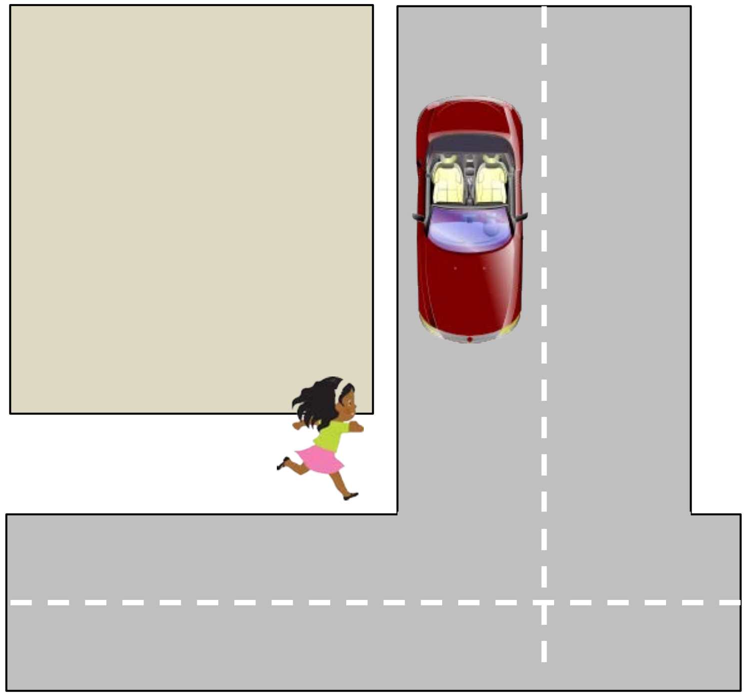
“Expect them on the road by 2020”
- General Motors

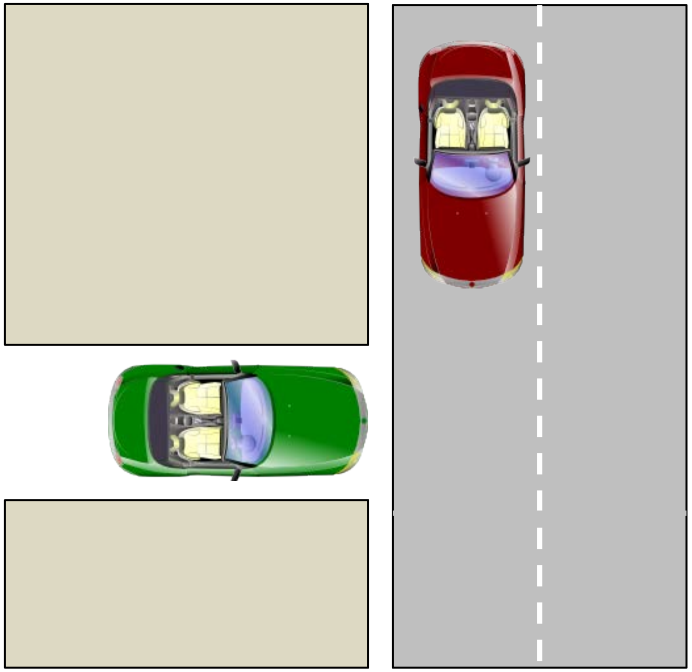
- Nevada legalized testing autonomous vehicles. California, Florida expected to follow.

- Autonomous Vehicles tested on Europe’s roads

Challenge: Safely Detecting Hidden Objects

- Sensors on a car see only line of sight objects





Challenge: Safely Detecting Hidden Objects

- Sensors on a car see only line of sight objects

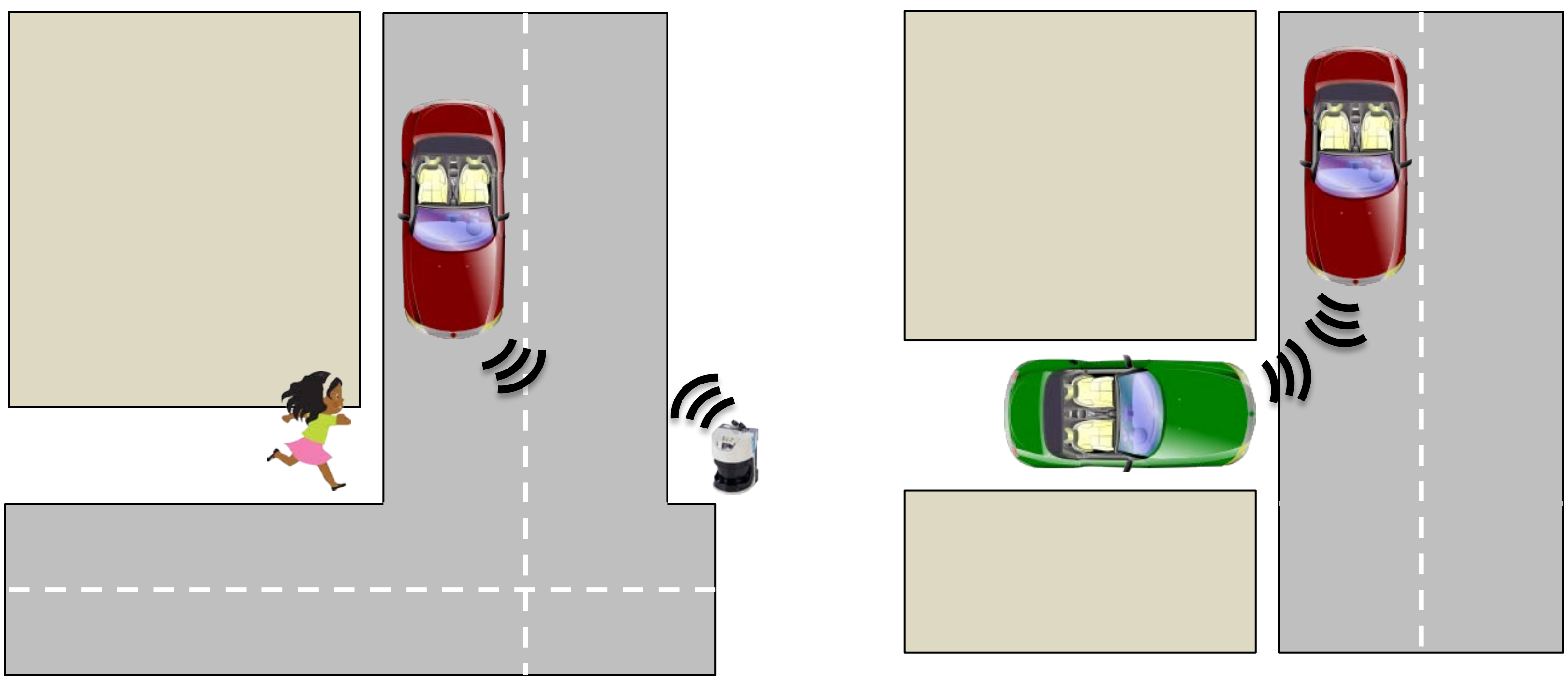
- Hidden objects affect autonomous cars

  - "Google's autonomous car requires occasional human intervention to prevent accident"

  - "Future of autonomous driving depends on detecting hidden objects & blind spots" - DARPA Challenge [JFR08]

How can autonomous vehicles detect hidden objects?

Communication



How can autonomous vehicles detect hidden objects?

Communication

- Expand field of view beyond line of sight
- Also valuable for human drivers – can react faster to objects they couldn't see

Simply use past work on VANETs?

VANETs typically oblivious to application

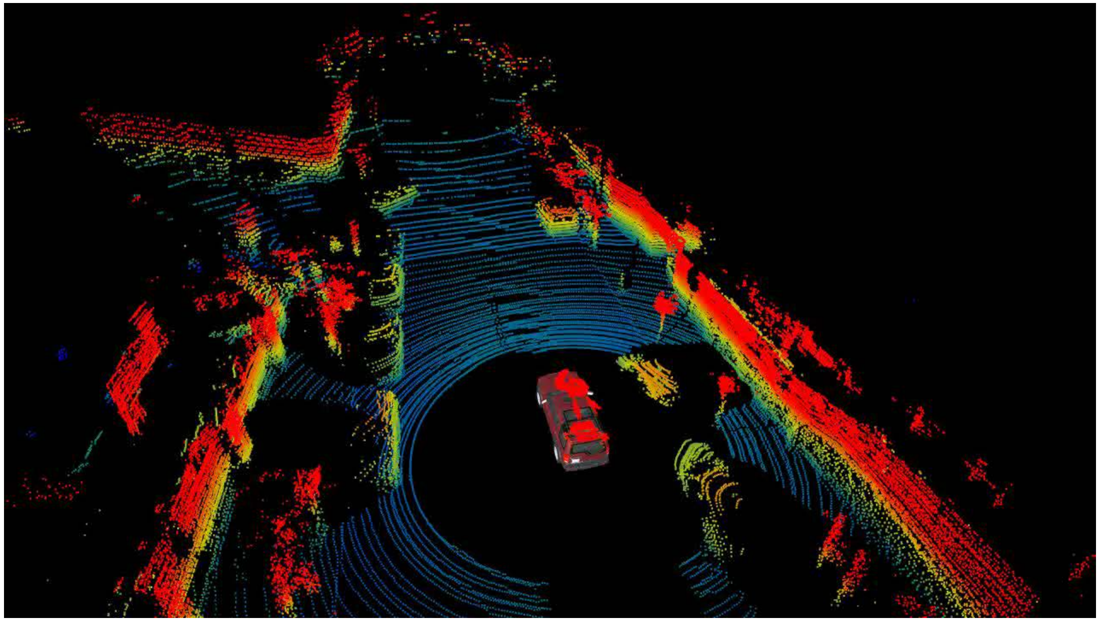
- Efficient routing
- Reliable message delivery
- … (etc)

But, not integrated with specific applications

Autonomous Driving needs Tight Integration with Communication

- Data is huge and time critical

  →Communication should focus on information most critical to the application



Autonomous Driving needs Tight Integration with Communication

- Data is huge and time critical

  →Communication should focus on information most critical to the application

- Don’t know who has the desired content
  - In typical networks, you know your destination
  - Instead, autonomous car seeks sensor data from part of the road, e.g. an intersection
  - It doesn’t know which car has this information
  - Focus on content as opposed to accessing a particular destination

CarSpeak

- Integrates communication with path planning and navigation in autonomous vehicles
- Has a content centric design
  - content, i.e. parts of road, is a first class citizen
- New MAC design where content, not senders, contend for the medium
- Implemented & evaluated on real autonomous vehicles

1. What is “content” and how do we represent it?

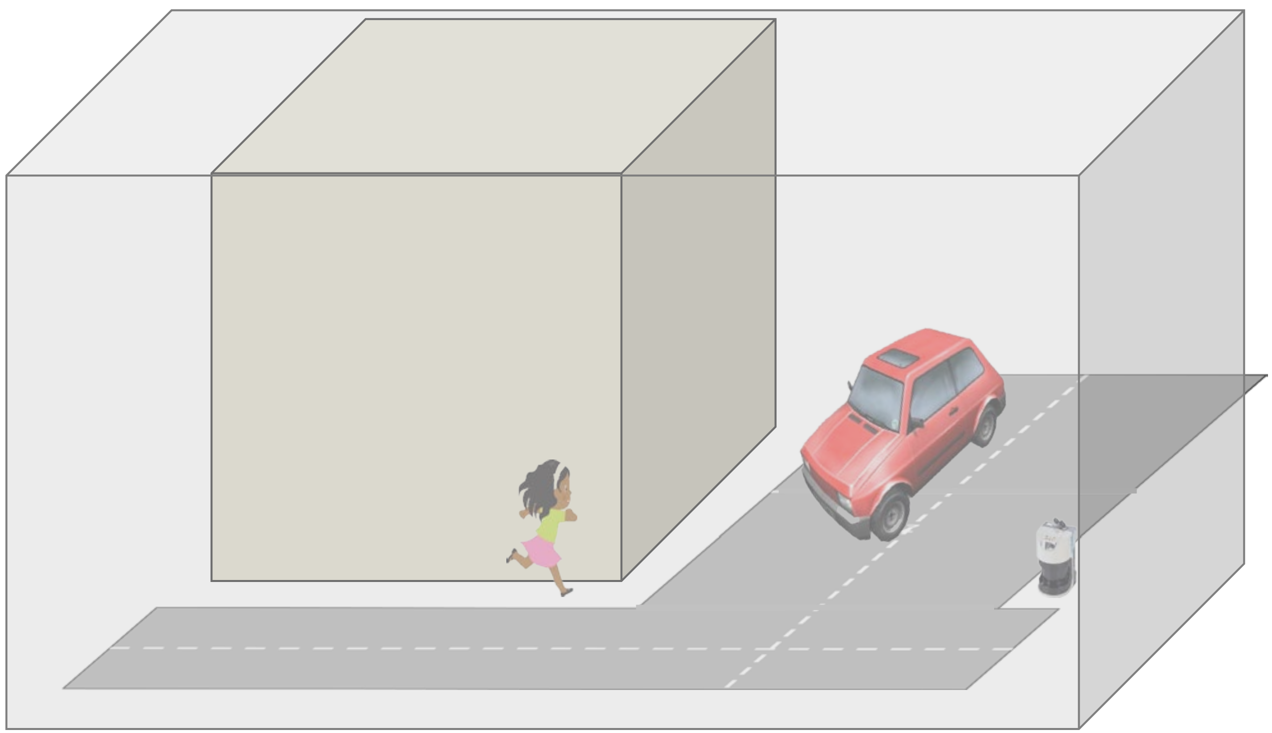
2. How do we disseminate this content?

1. What is “content” and how do we represent it?

2. How do we disseminate this content?

What is “content” and how do we represent it?

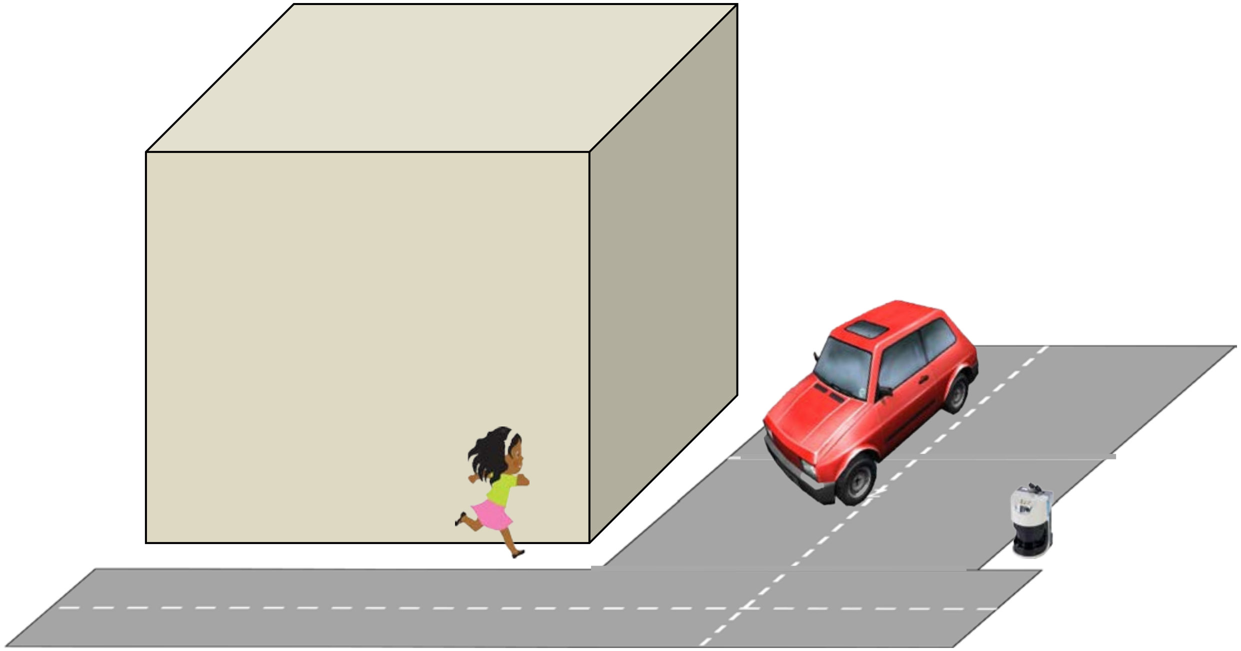
- Content is sensor data from cubes in the environment



How do we represent these cubes in environment?

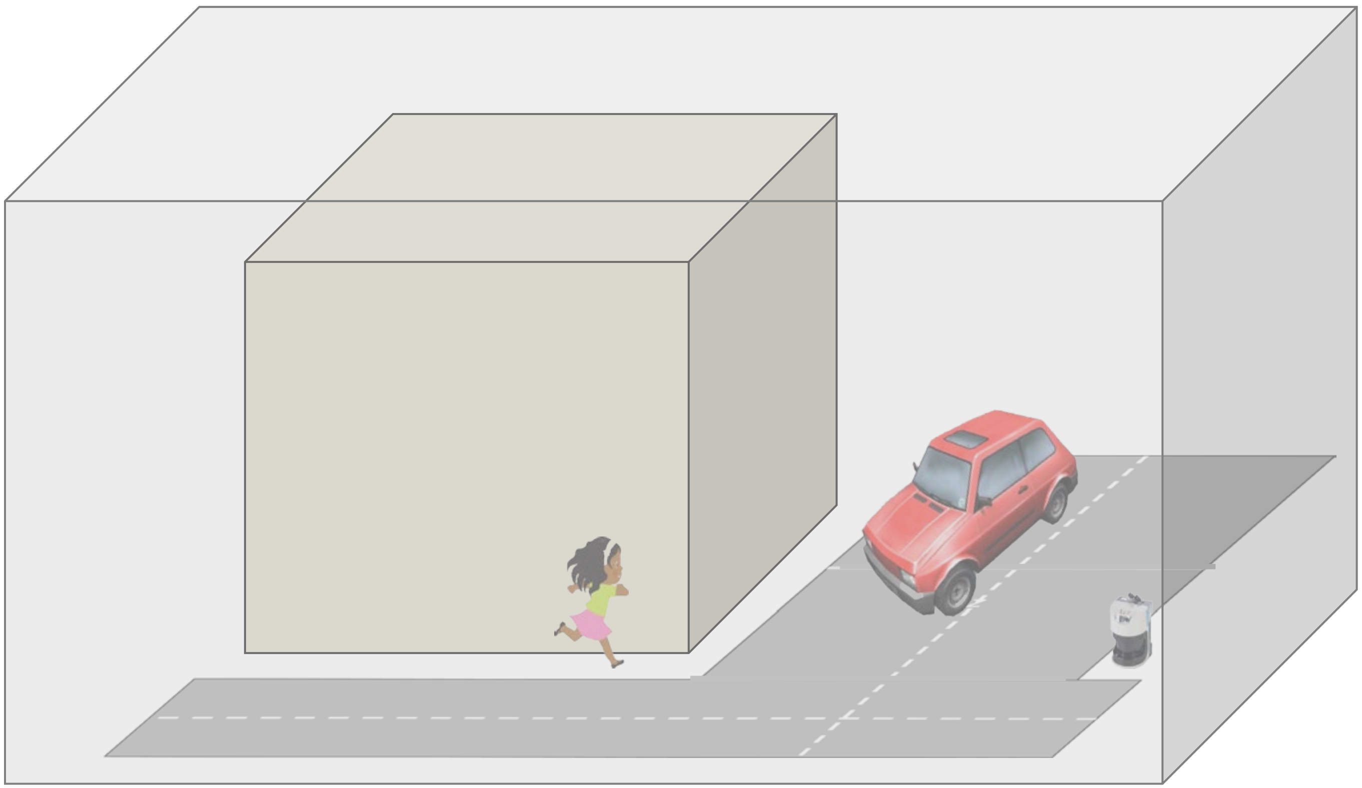
Ideally…

- Obtain low resolution view of environment
- Zoom in for higher resolution view of a smaller part of environment



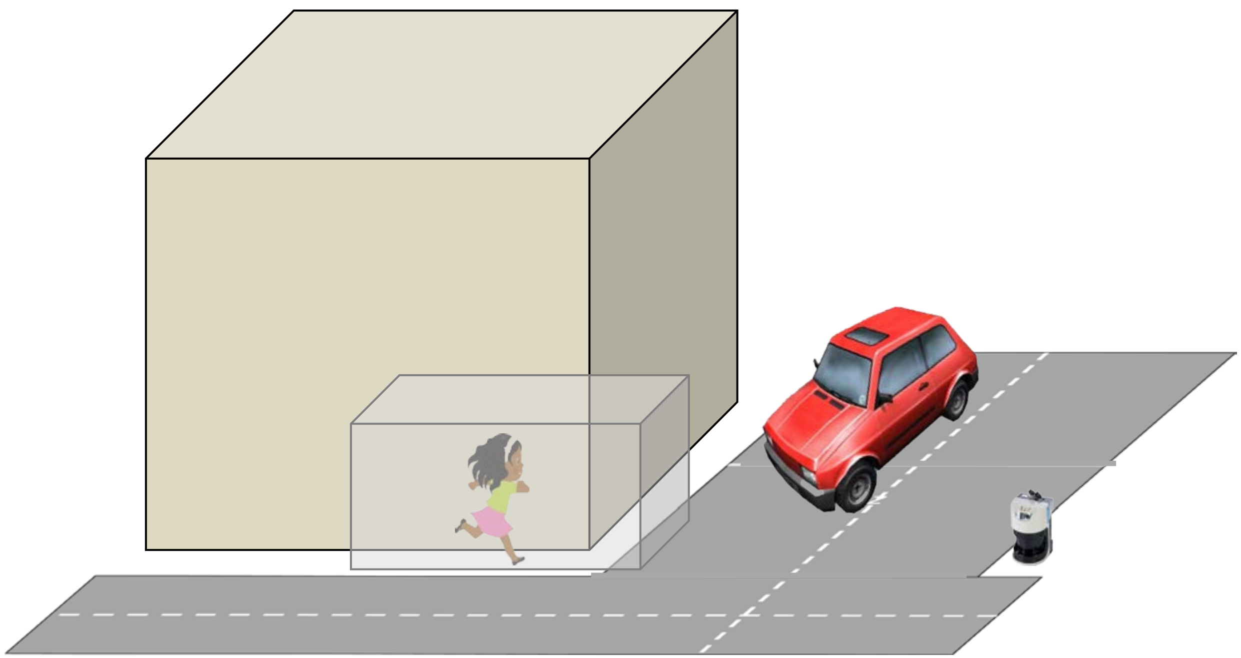
Ideally…

- Obtain low resolution view of environment
- Zoom in for higher resolution view of a smaller part of environment



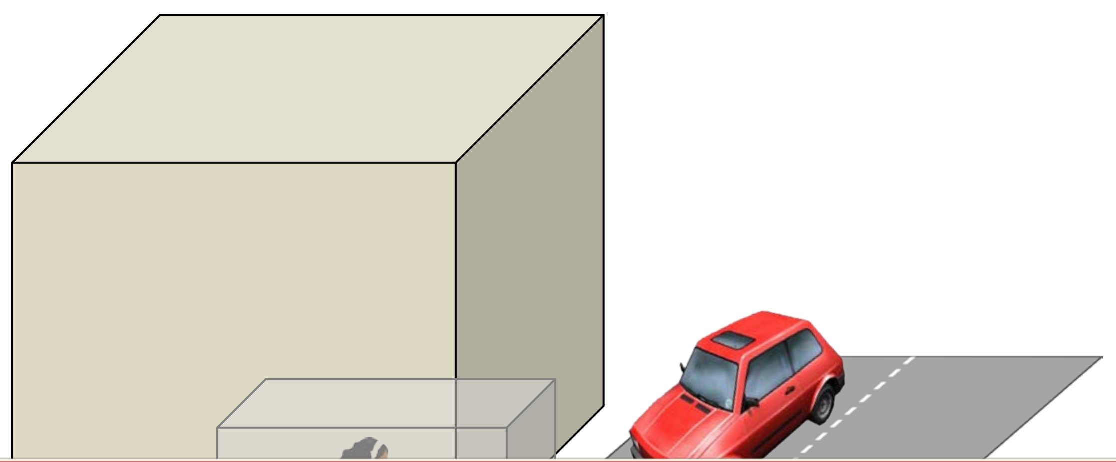
Ideally…

- Obtain low resolution view of environment
- Zoom in for higher resolution view of a smaller part of environment



Ideally…

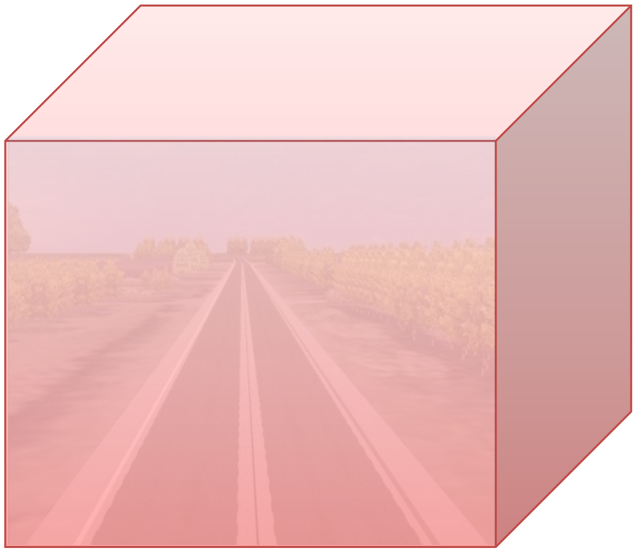
- Obtain low resolution view of environment
- Zoom in for higher resolution view of a smaller part of environment



Need recursive representation that makes best use of available wireless bandwidth

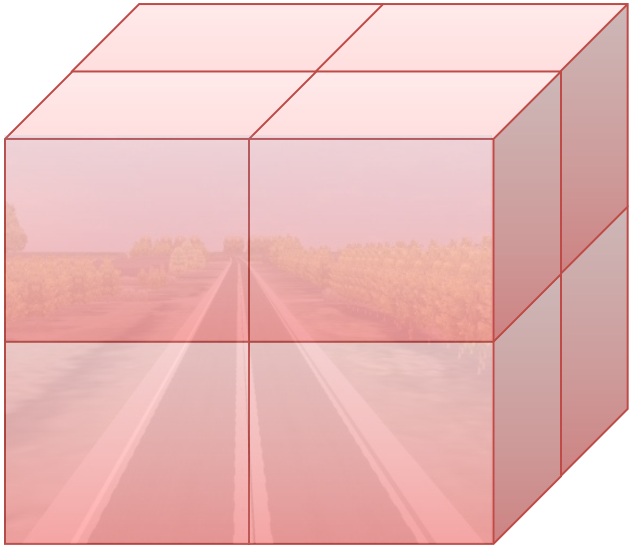
Representing Content in CarSpeak

- Consider large cube encompassing environment



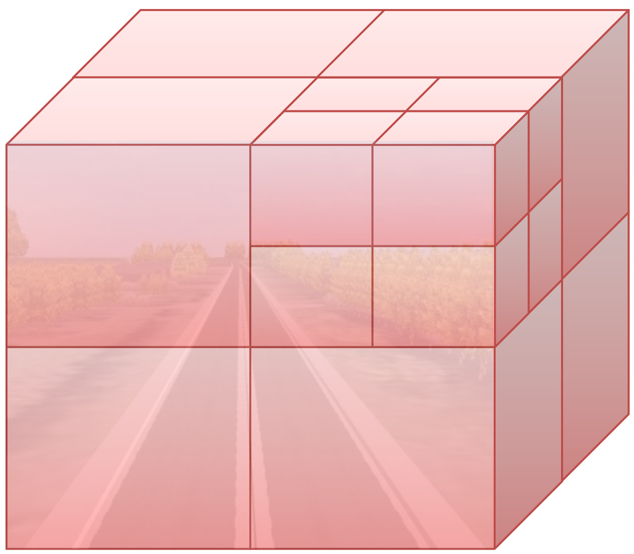
Representing Content in CarSpeak

- Consider large cube encompassing environment
- Recursively divide into 8 smaller cubes



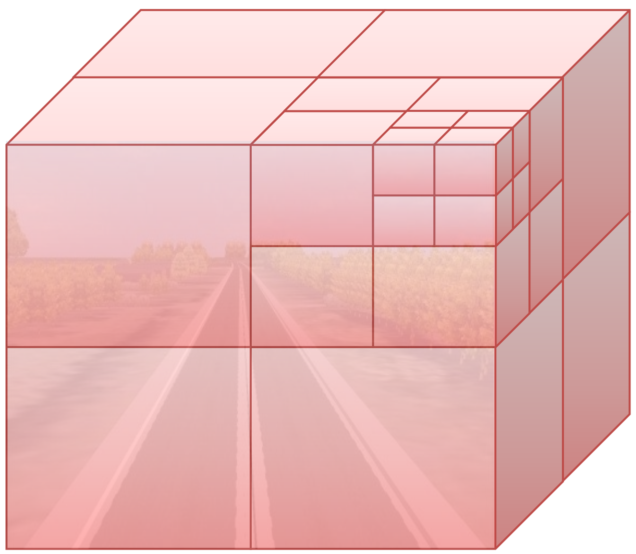
Representing Content in CarSpeak

- Consider large cube encompassing environment
- Recursively divide into 8 smaller cubes



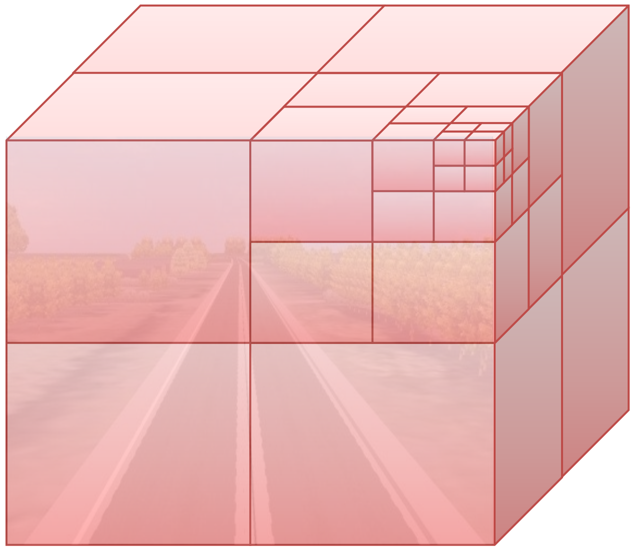
Representing Content in CarSpeak

- Consider large cube encompassing environment
- Recursively divide into 8 smaller cubes



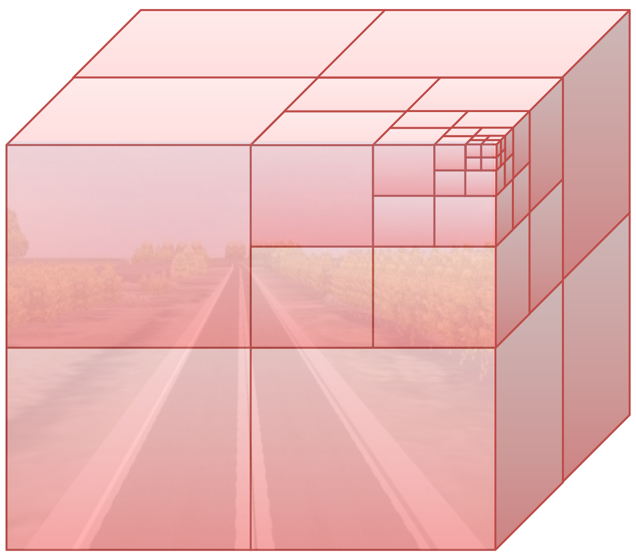
Representing Content in CarSpeak

- Consider large cube encompassing environment
- Recursively divide into 8 smaller cubes



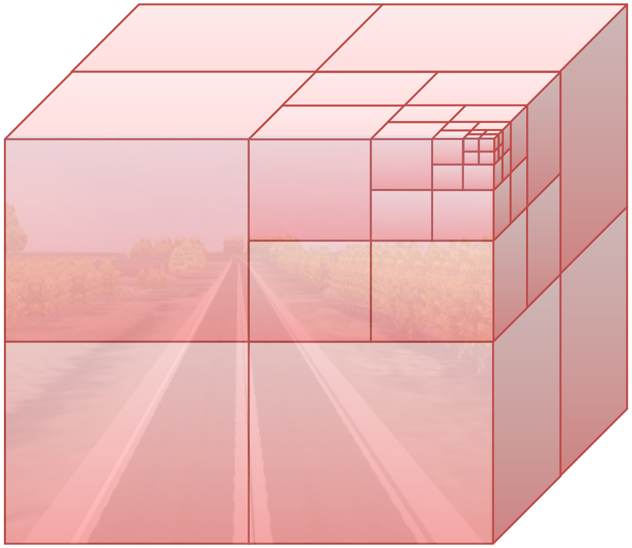
Representing Content in CarSpeak

- Consider large cube encompassing environment
- Recursively divide into 8 smaller cubes

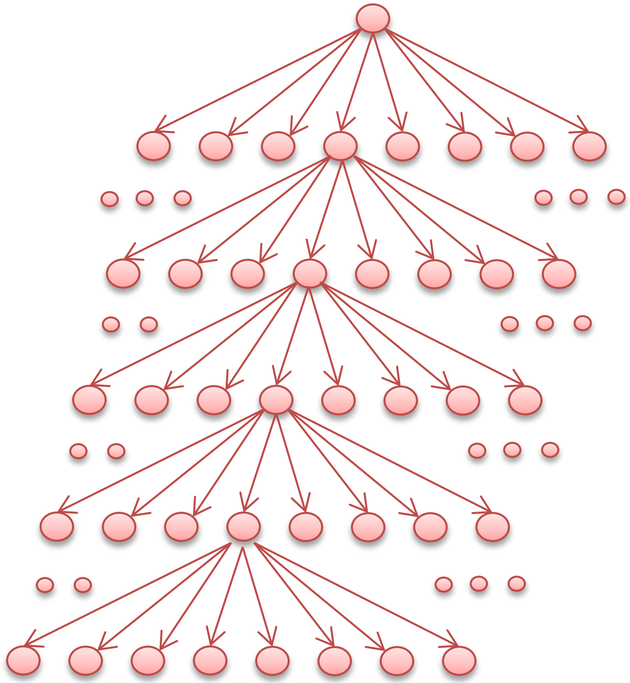


Representing Content in CarSpeak

- Consider large cube encompassing environment
- Recursively divide into 8 smaller cubes



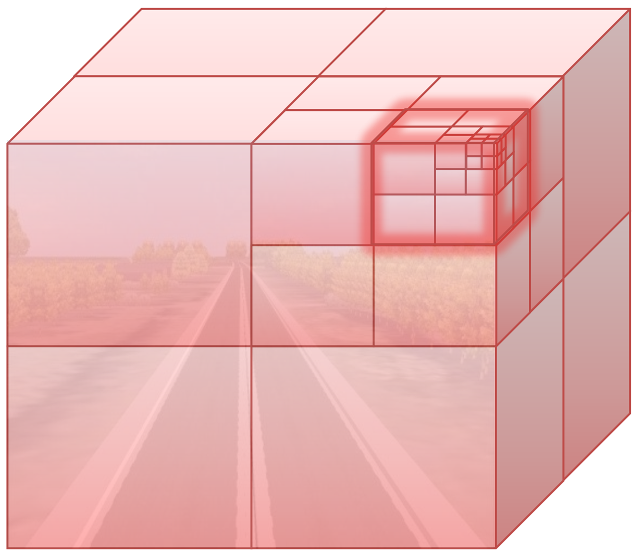
Cubes



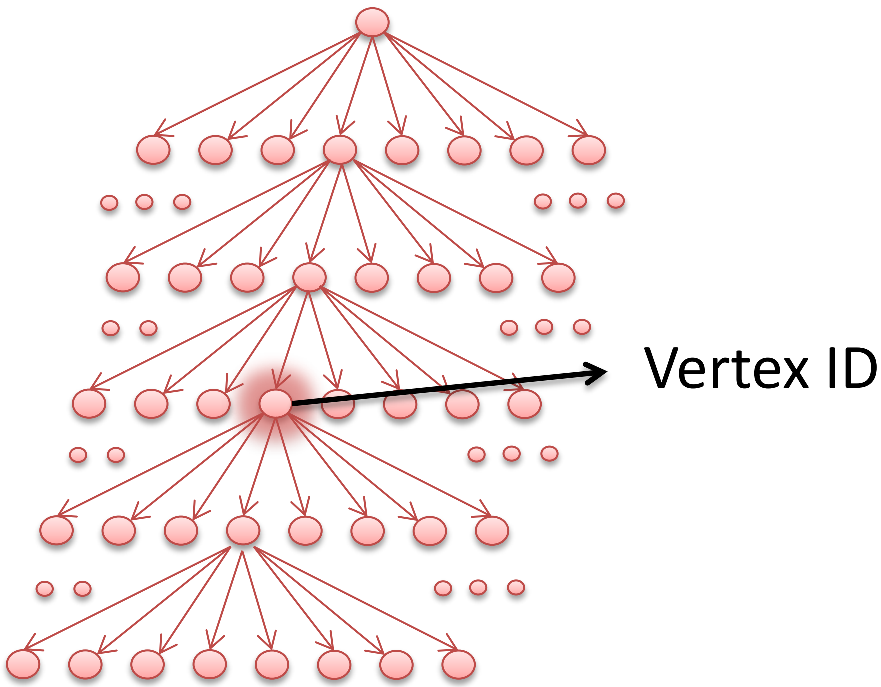
Tree Representation

Representing Content in CarSpeak

- Consider large cube encompassing environment
- Recursively divide into 8 smaller cubes



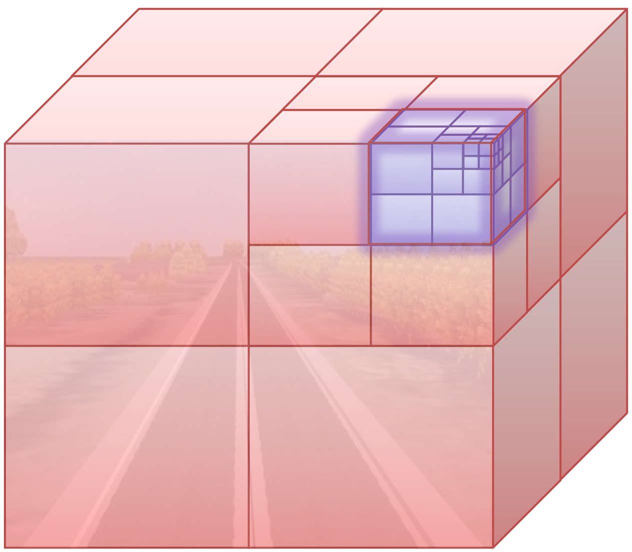
Cubes



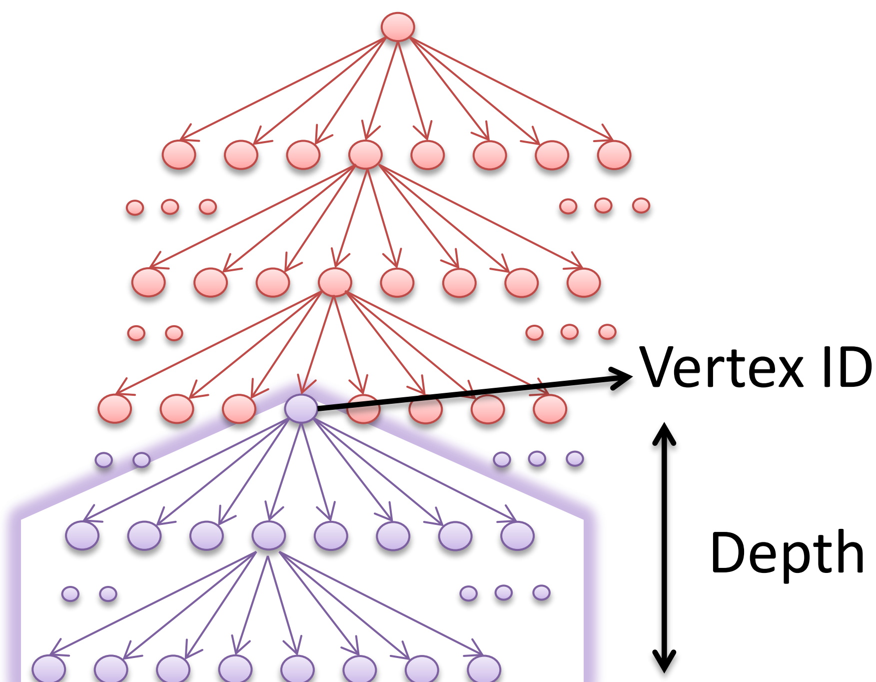
Tree Representation

Representing Content in CarSpeak

- Consider large cube encompassing environment
- Recursively divide into 8 smaller cubes
- Car needs cube at resolution (vertex ID, depth)



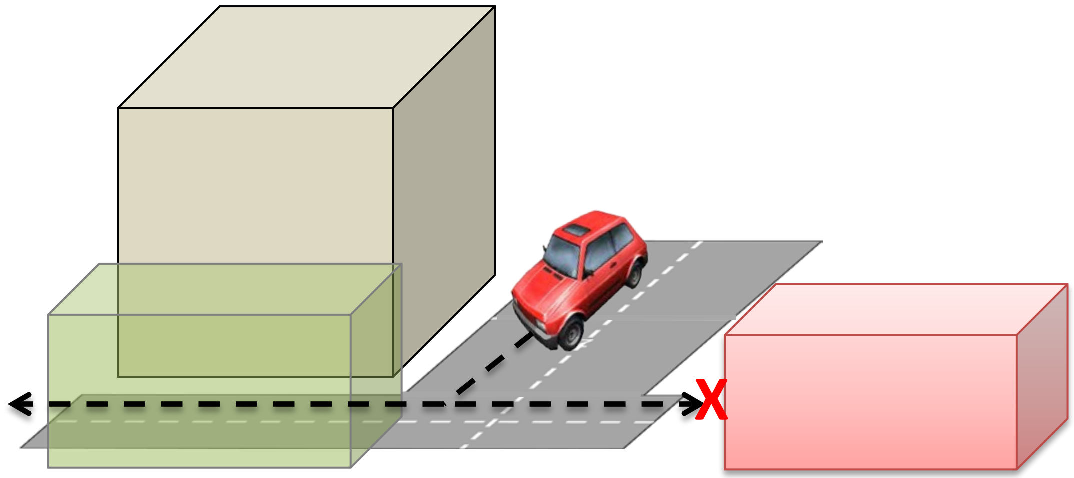
Cubes



Tree Representation

What Info does Autonomous Car Need?

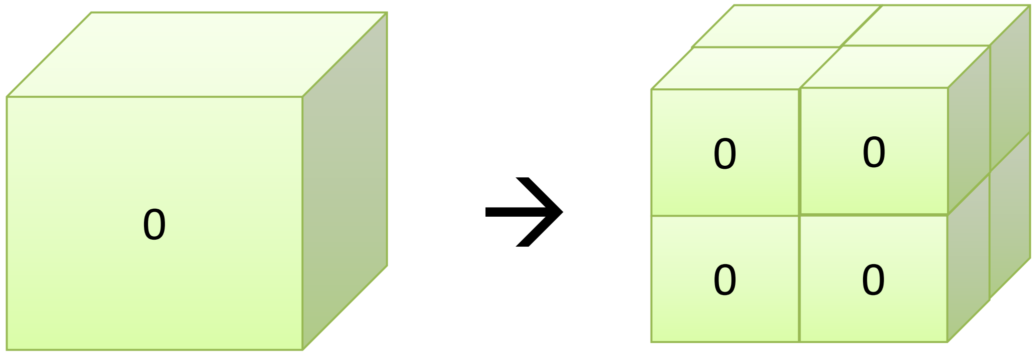
- Looks for obstacle free paths to destination
- Needs to know which parts of environment:
  - Are empty and safe to pass through
  - Are occupied and unsafe to pass through



What Info does Autonomous Car Need?

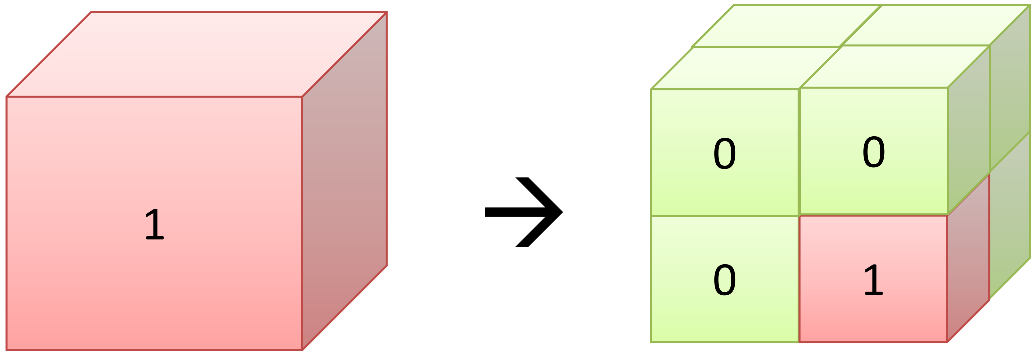
- Each cube has one bit: Empty (0) or Occupied (1)
- If cube is empty

  →all cubes inside are empty



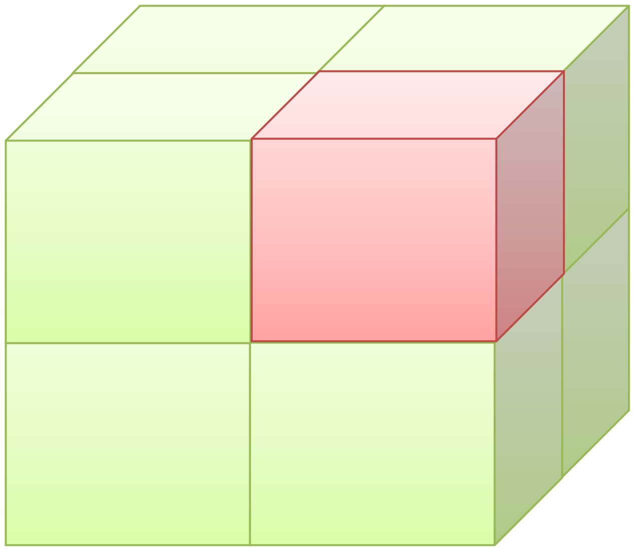
What Info does Autonomous Car Need?

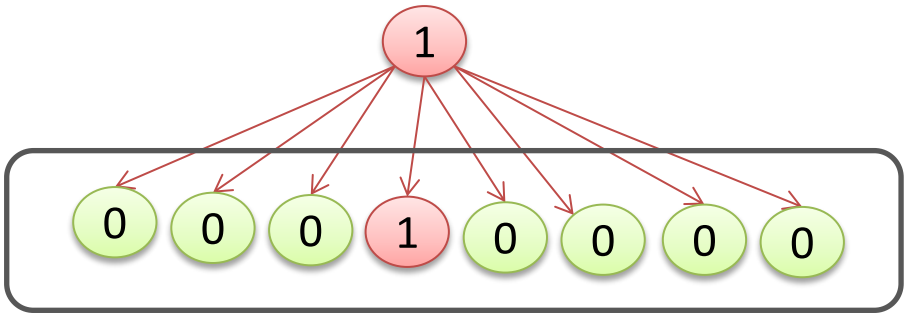
- Each cube has one bit: Empty (0) or Occupied (1)
- If cube is empty
  → all cubes inside are empty
- If cube is occupied
  → at least one cube inside is occupied



Compactly Representing Data

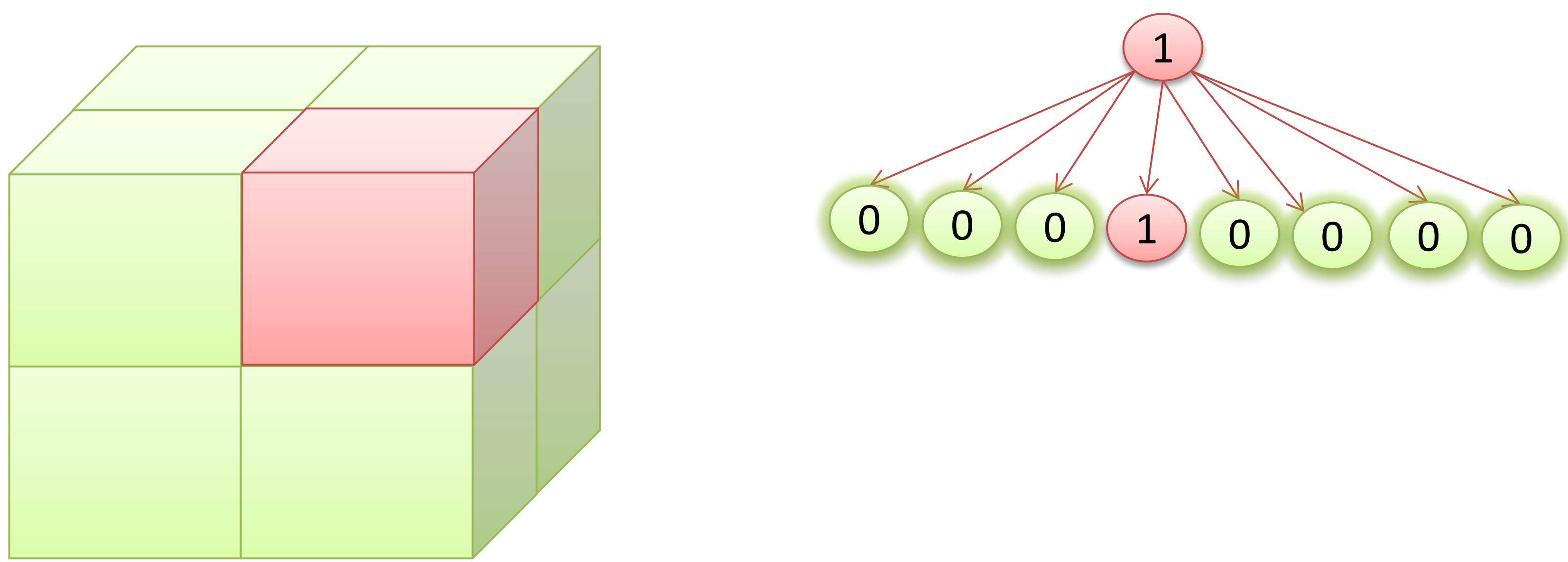
- Level 1 has 8 bits where 0-empty, 1-occupied





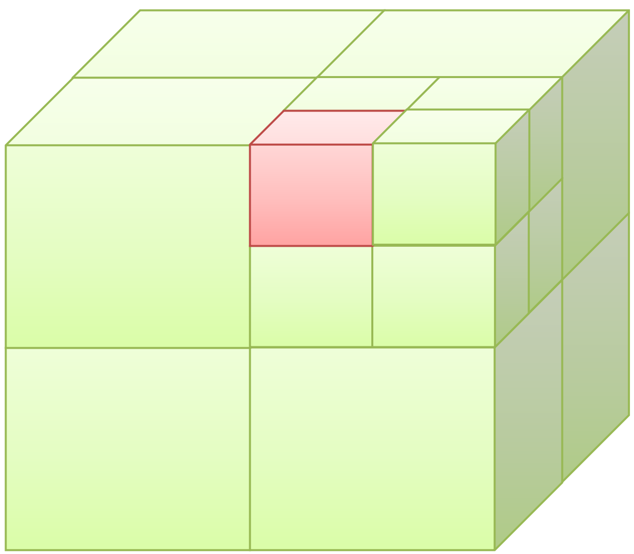
Compactly Representing Data

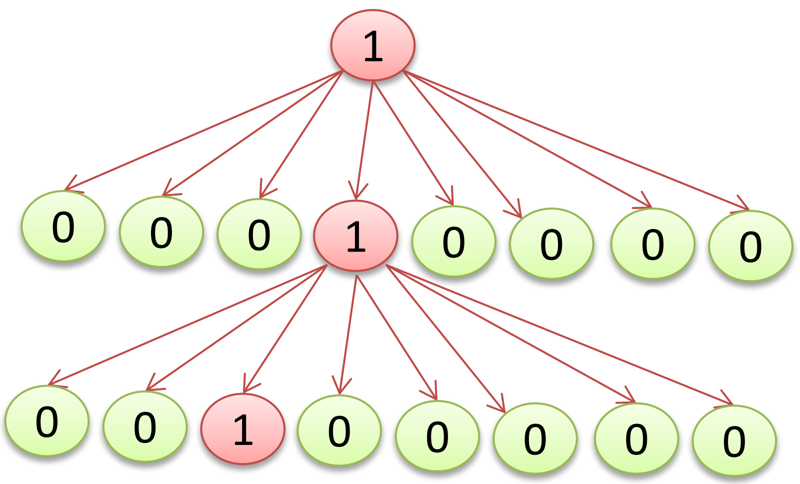
- Level 1 has 8 bits where 0-empty, 1-occupied
- None of 0 nodes need to be expanded



Compactly Representing Data

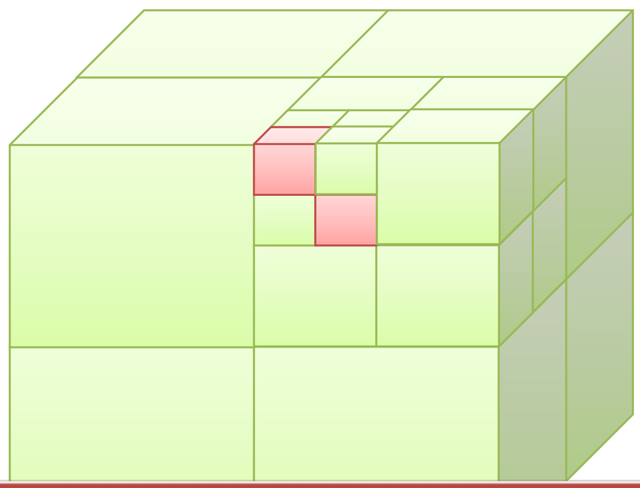
- Level 1 has 8 bits where 0-empty, 1-occupied
- None of 0 nodes need to be expanded
- Expand 1 node to see inside at more resolution

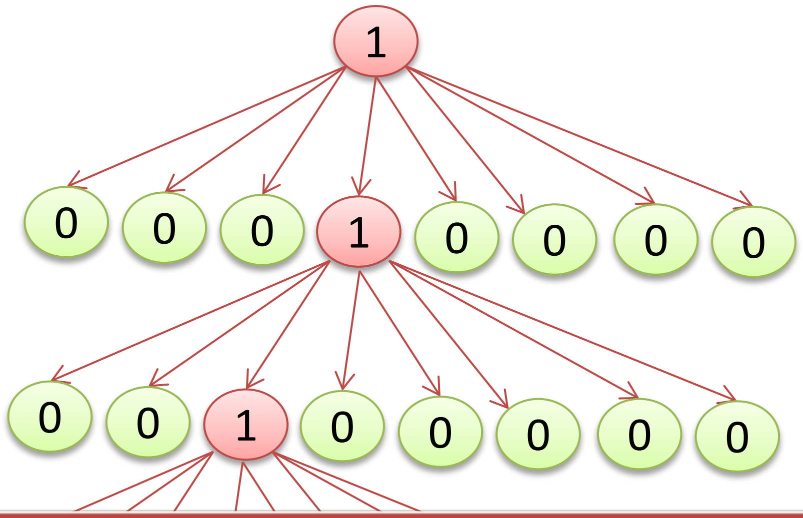




Compactly Representing Data

- Level 1 has 8 bits where 0-empty, 1-occupied
- None of 0 nodes need to be expanded
- Expand 1 node to see inside at more resolution





Tree representation compresses data efficiently

1. What is “content” and how do we represent it?

2. How do we disseminate this content?

How do we disseminate this content?

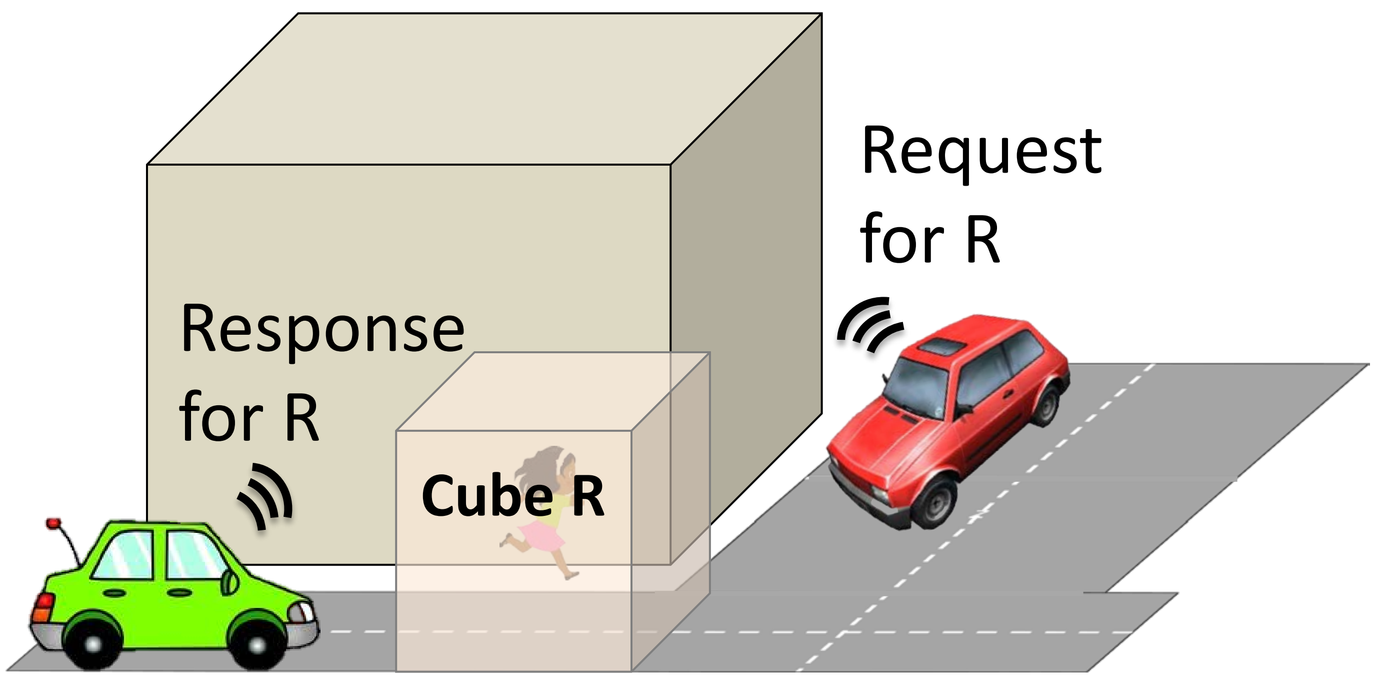
Autonomous cars collect huge amount of data

→ Cannot flood medium with all their data

A Request-Response Approach

A Request-Response Approach

- Car requests only data of interest
  - E.g. at blind spots, intersections, etc.
- Cars which sense the data, may respond

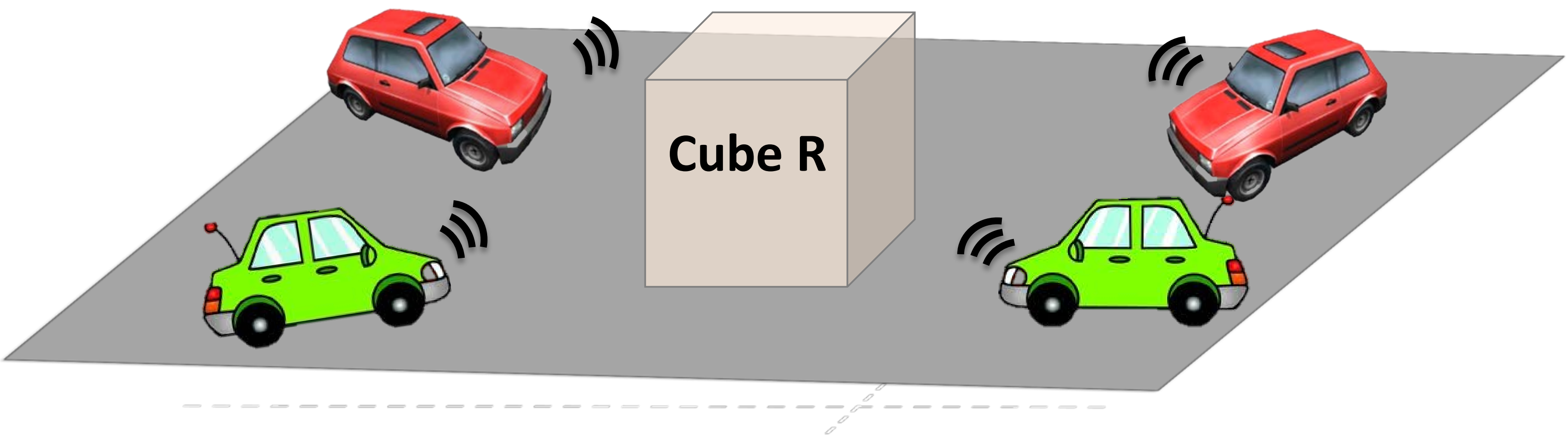


Challenge 1: Who Should Respond?

Challenge 1: Who Should Respond?

Naïve solution 1: Simply let all cars respond

→ A lot of redundant data



Challenge 1: Who Should Respond?

Naïve solution 2: One car respond; others who hear it suppress their response

→ Responder leaves before sending all packets

→ Different cars have different perspectives

Need to balance data diversity with data overlap

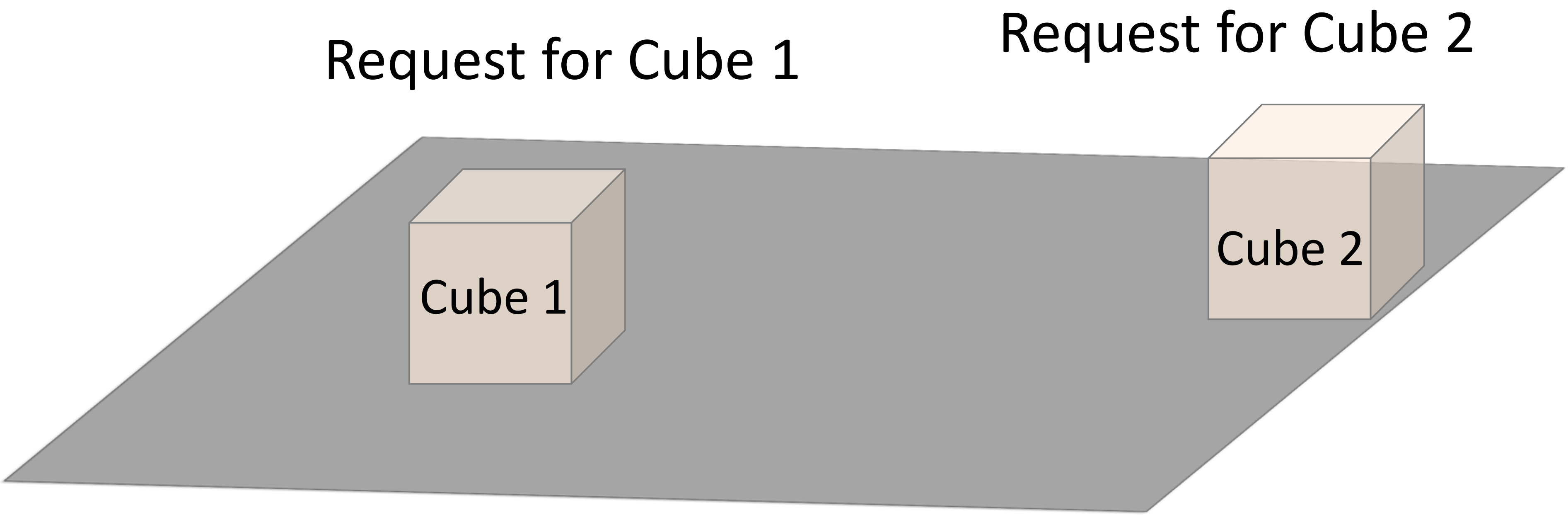
Solution: Random Walks

- Content of the cube (i.e., its subtree) is divided into packets
- Each car uses a different random walk to transmit packets

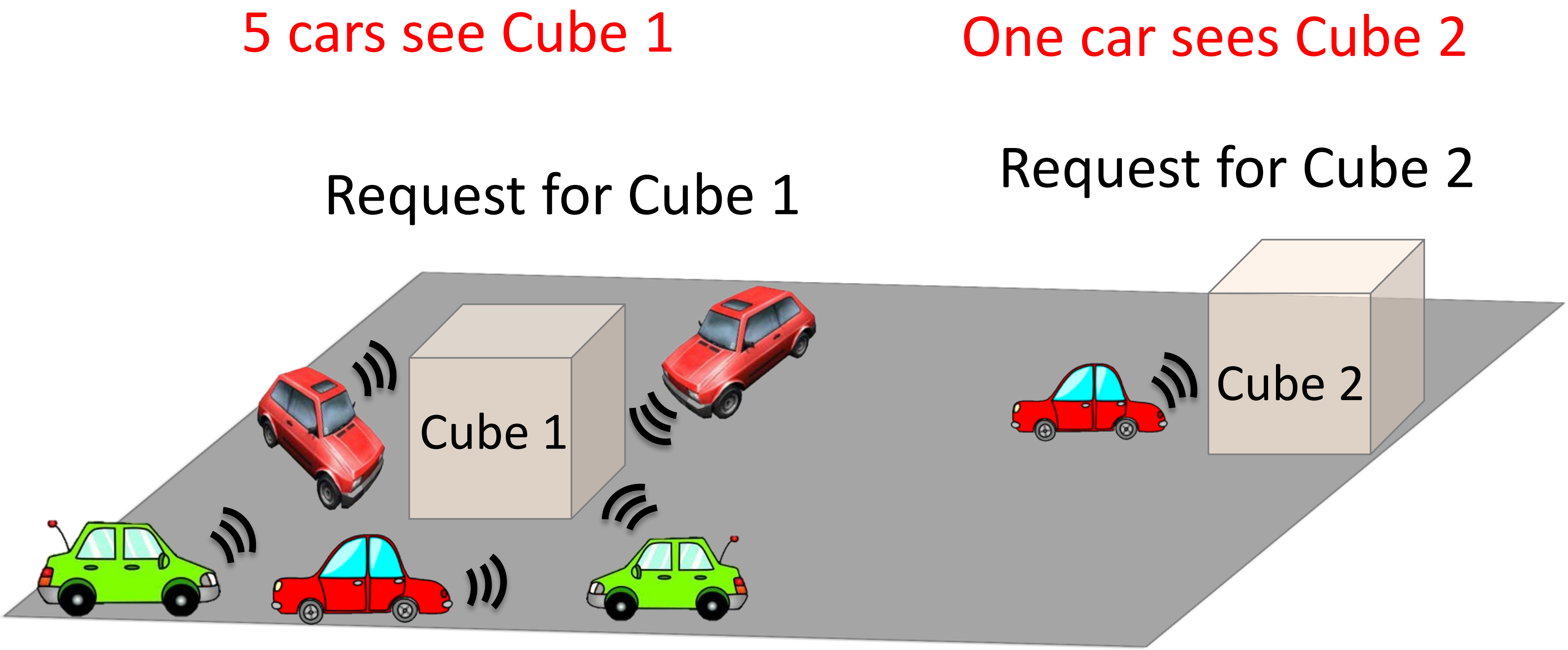
If one car transmits → Eventually finishes walk

If multiple cars transmit → Overlap is minimum

Challenge 2: Ensure medium is shared fairly by requested content



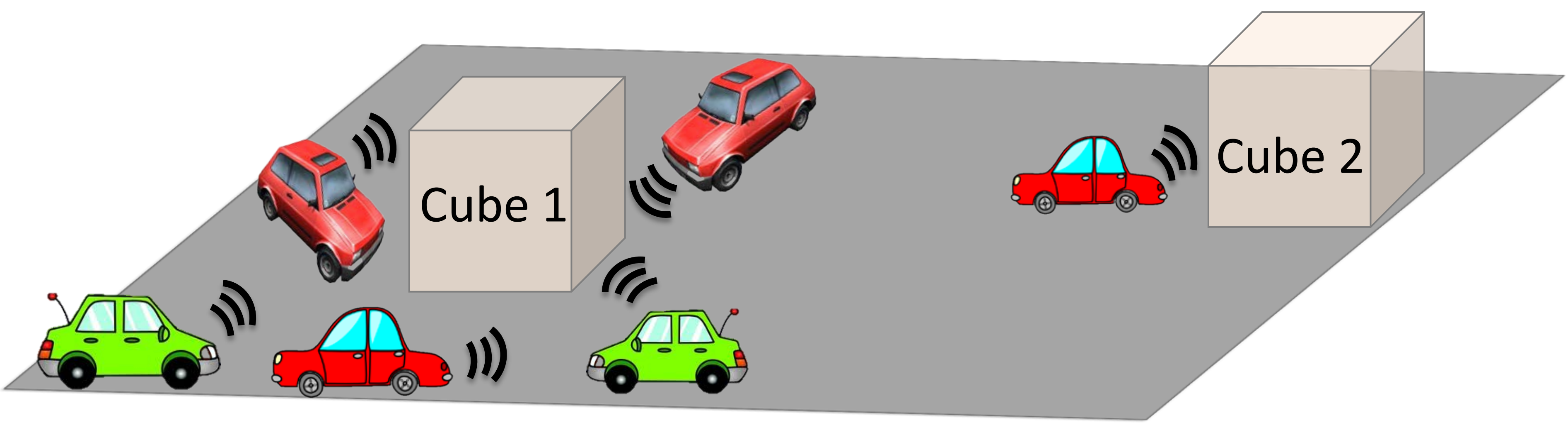
Challenge 2: Ensure medium is shared fairly by requested content



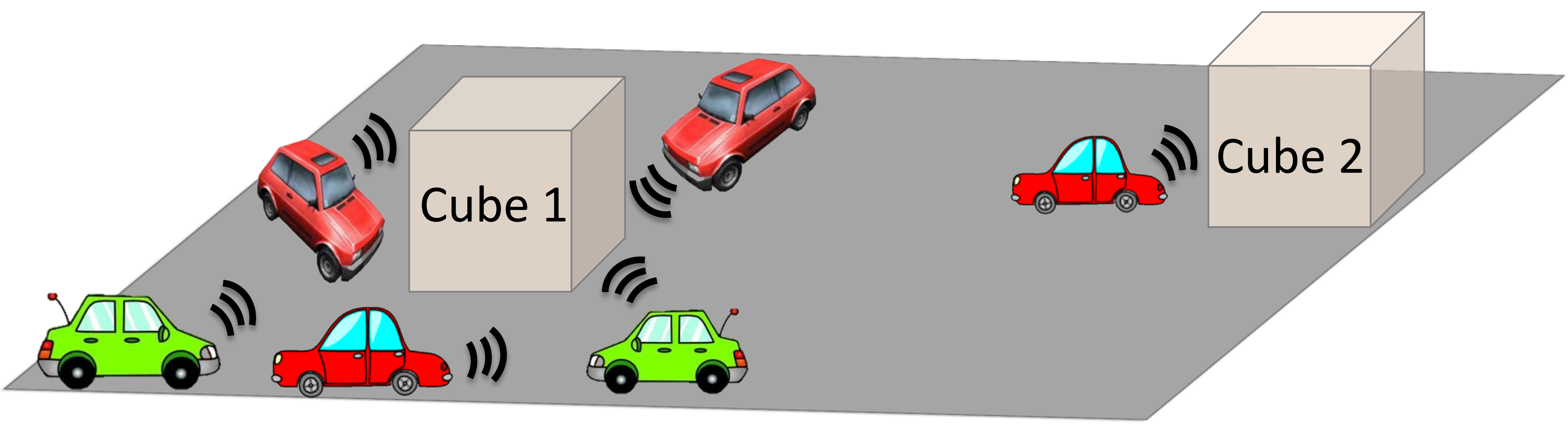
Challenge 2: Ensure medium is shared fairly by requested content

802.11 shares medium between senders
→ Cube 1's share = 5 x Cube 2's share

Ideally, we want a MAC where
→ Cube 1 share = Cube 2's share



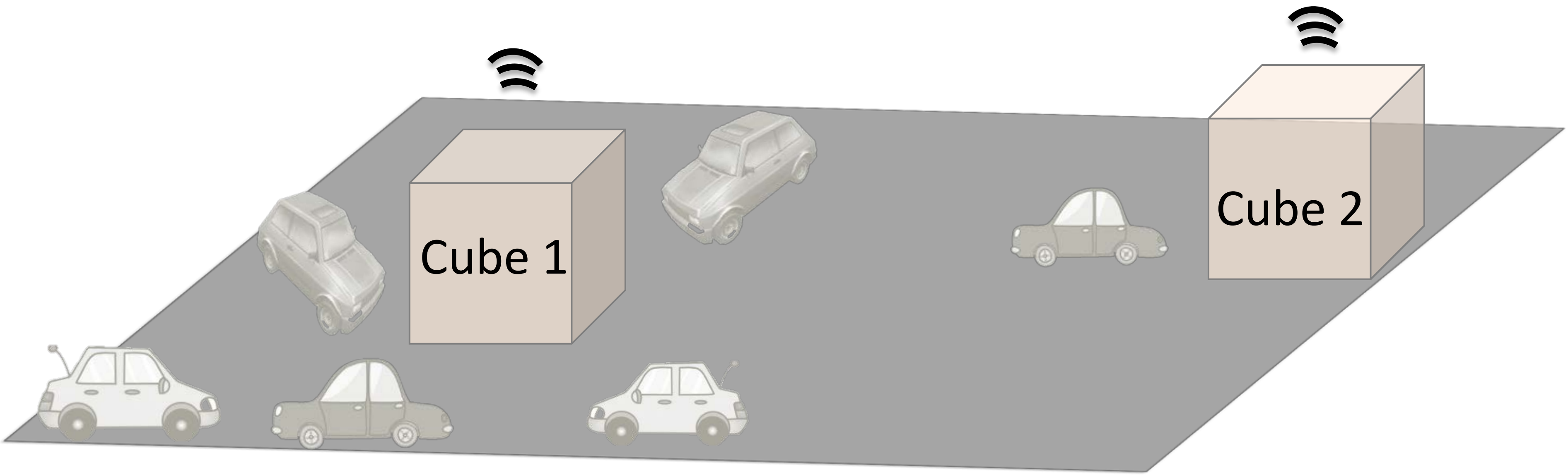
Solution: Replace sender-contention by content-contention



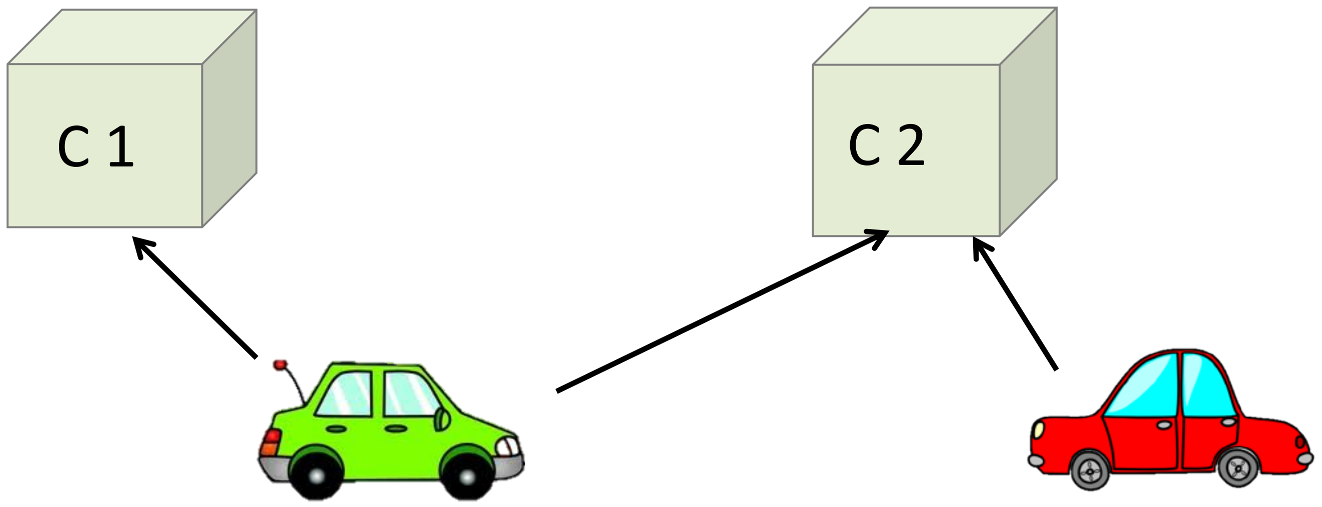
Solution 2: Replace sender-contention by content-contention

- Instead of senders, cubes contend for the medium
  → Requested cubes get equal share of medium

- But cubes are virtual entities
  → Cars viewing a cube contend on its behalf



But how can a car compute its medium share?

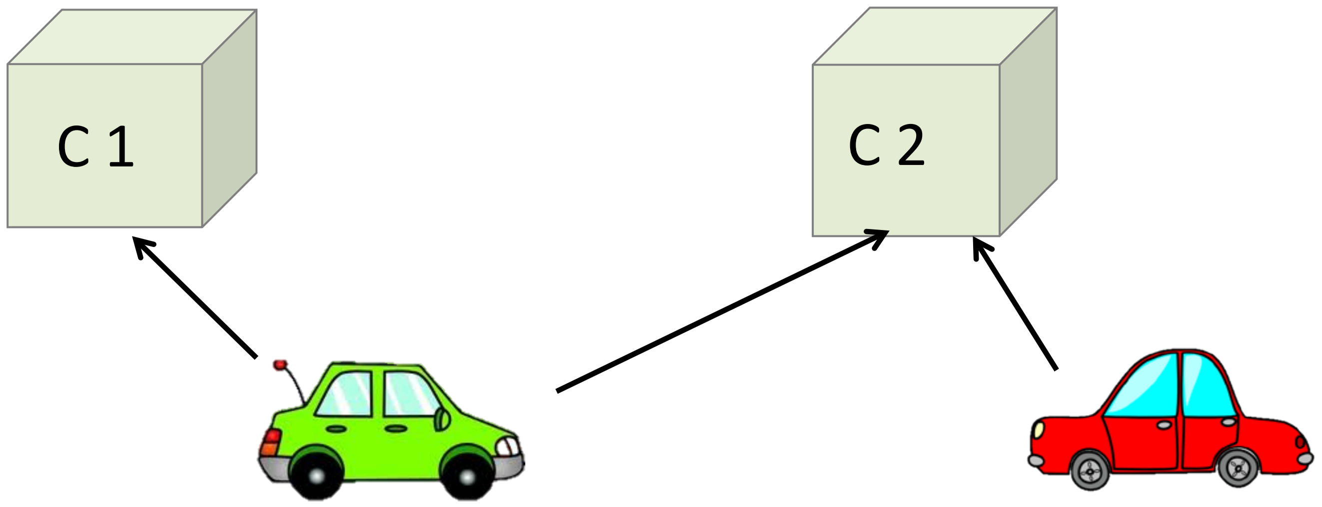


Each cube should get a share of 1/2

Green car share of the medium 3/4

red car share of the medium 1/4

But how can a car compute its medium share?



Car listens to how many cars respond for a particular cube

Share-per-cube = 1 / # cars responding

Car's total share = (Σ share-per-cube) / # requested cubes

Set contention window using car's share on off-the-shelf cards

Empirical Results

CarSpeak Implementation

- Implemented in Robot OS (ROS)
- Integrated with MIT’s Path Planner from DARPA challenge
- MAC implemented in the ath9k driver for Atheros WiFi cards

Compared Schemes

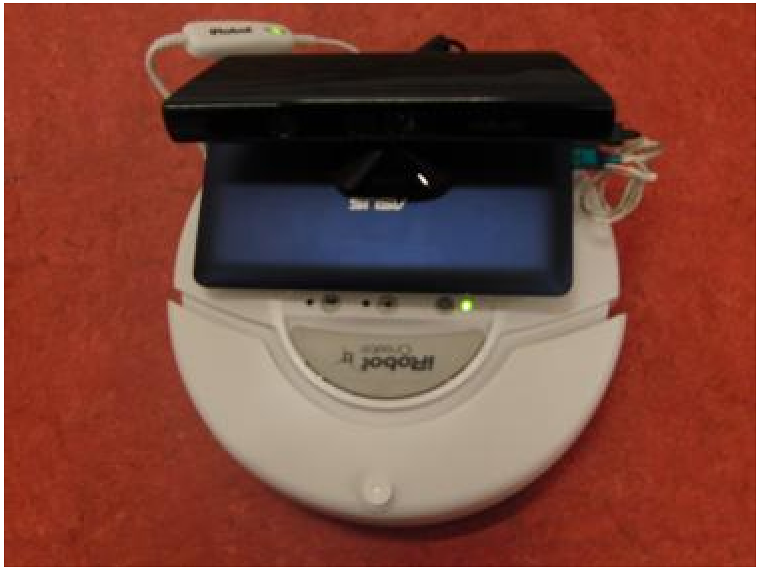
- CarSpeak
- 802.11 – Request & Response

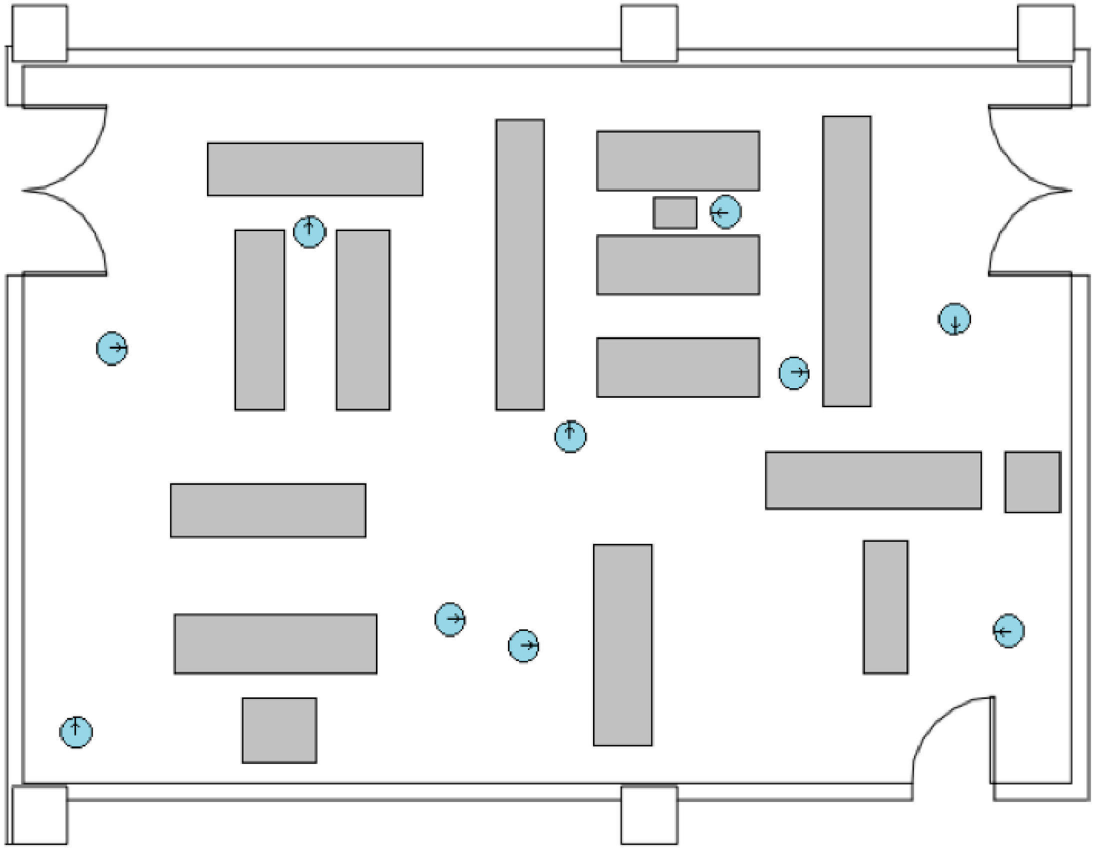
Experiments

- Indoor Testbed with Robots
- Outdoor Testbed with Autonomous Car

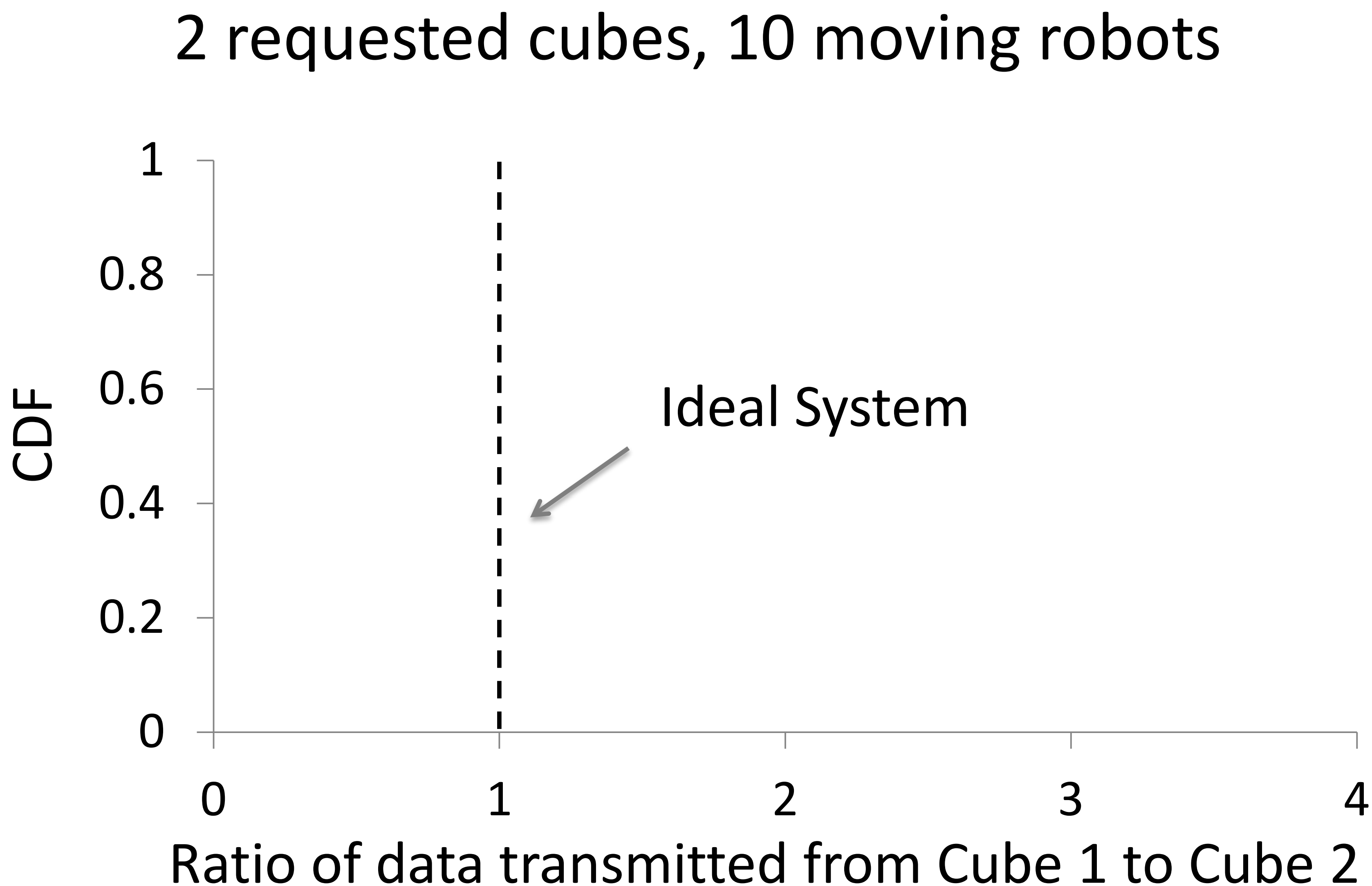
Indoor Testbed

- 10 Roomba robots with Kinect
- Navigate environment with obstacles

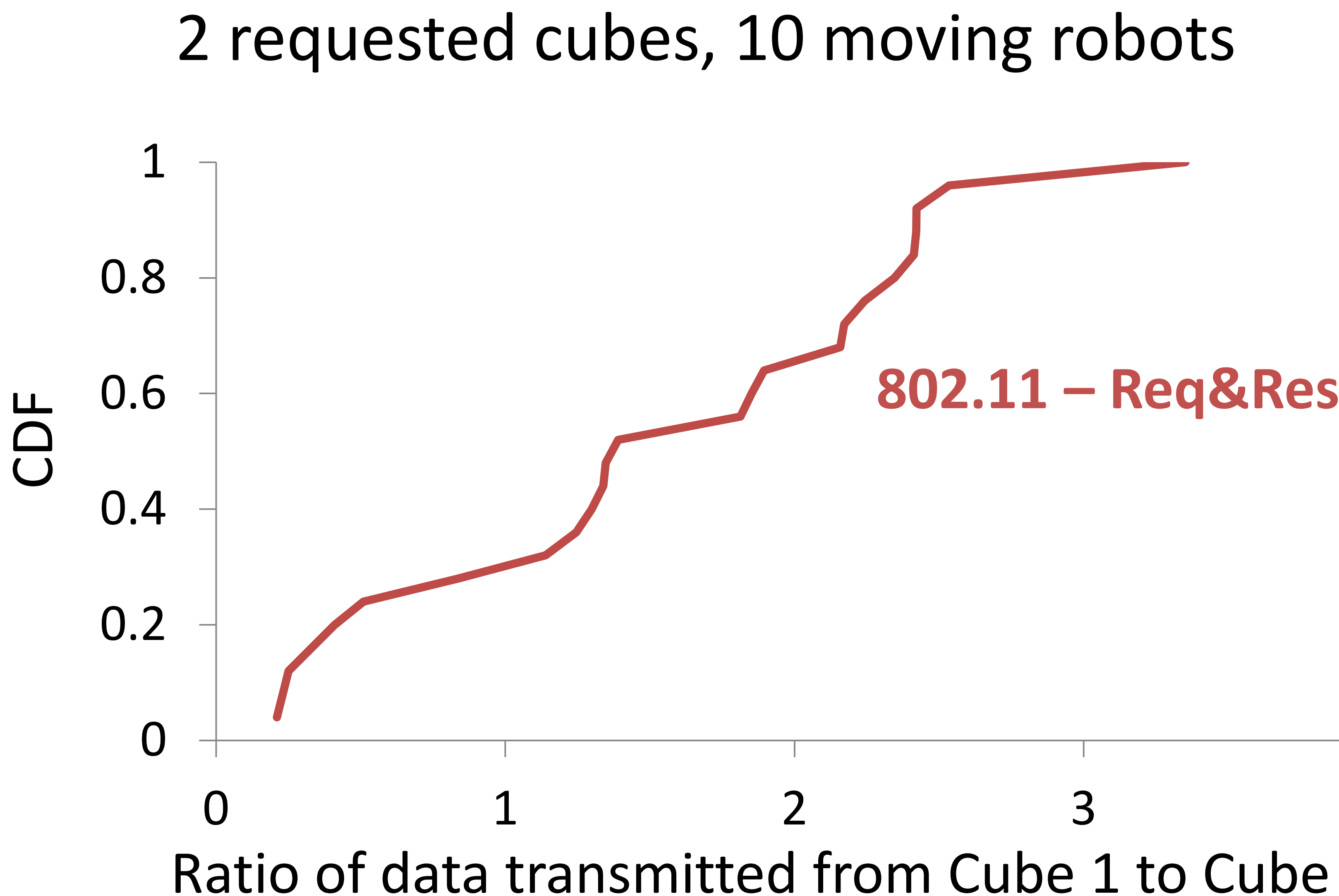




Can CarSpeak's MAC assign fair share to content?

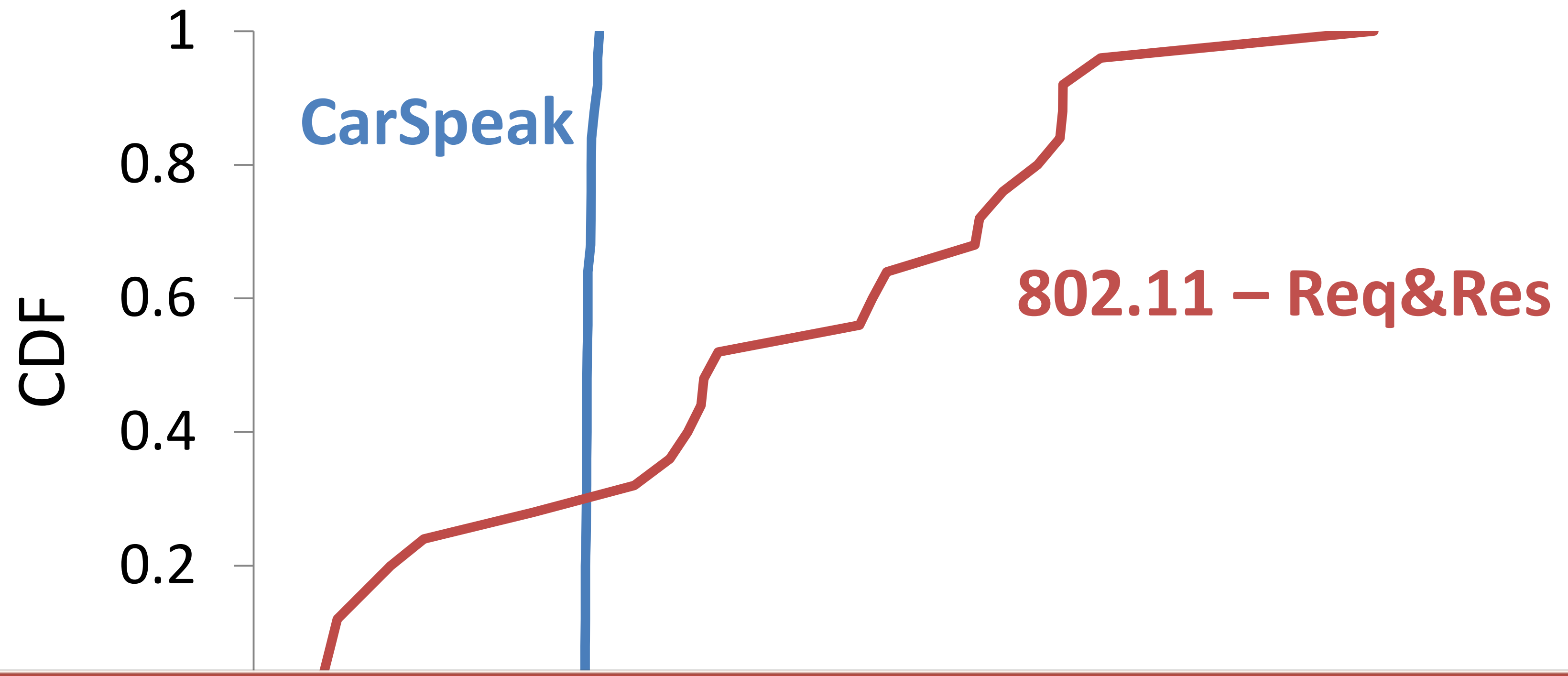


Can CarSpeak's MAC assign fair share to content?



Can CarSpeak's MAC assign fair share to content?

2 requested cubes, 10 moving robots

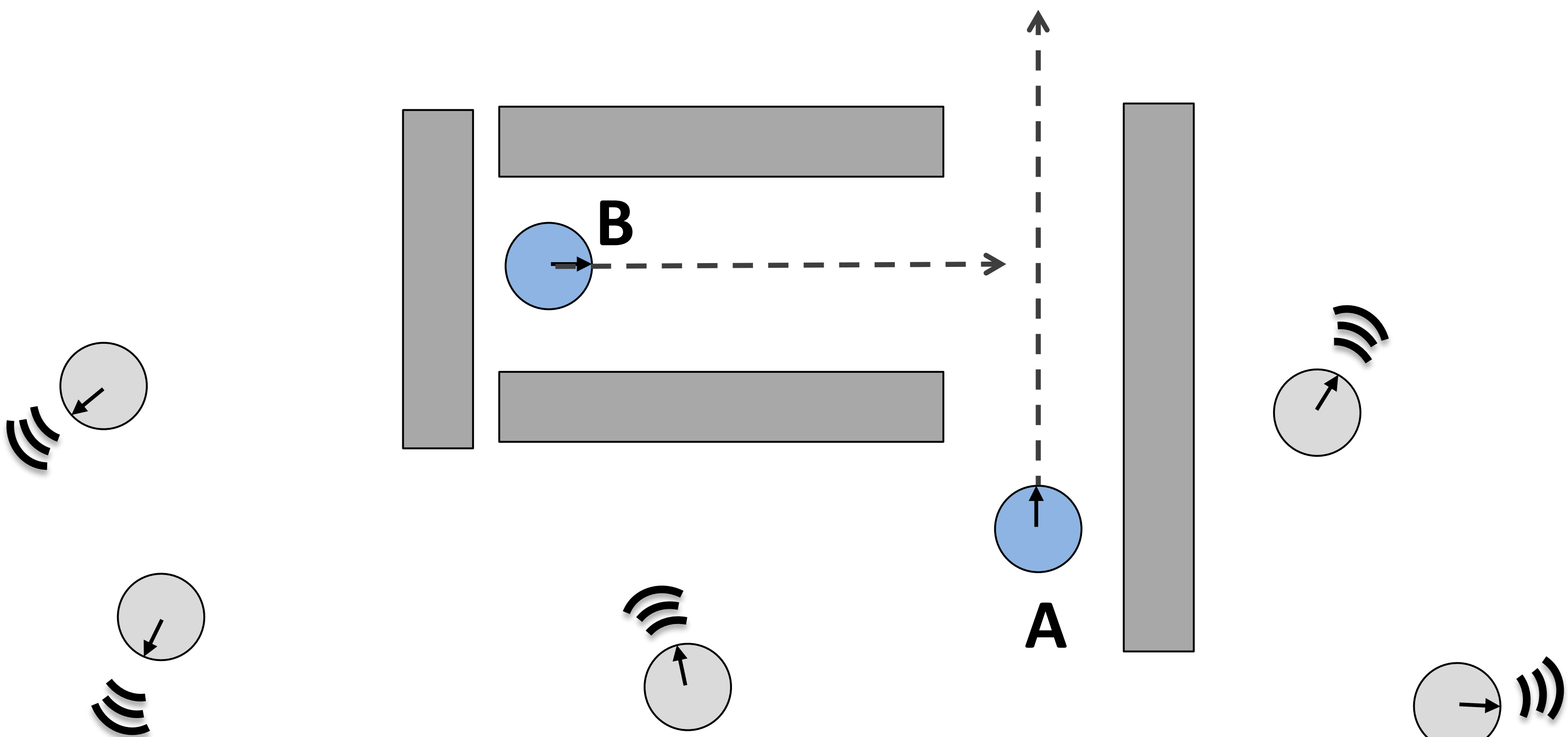


CarSpeak's content-centric MAC divides the medium fairly between requested content

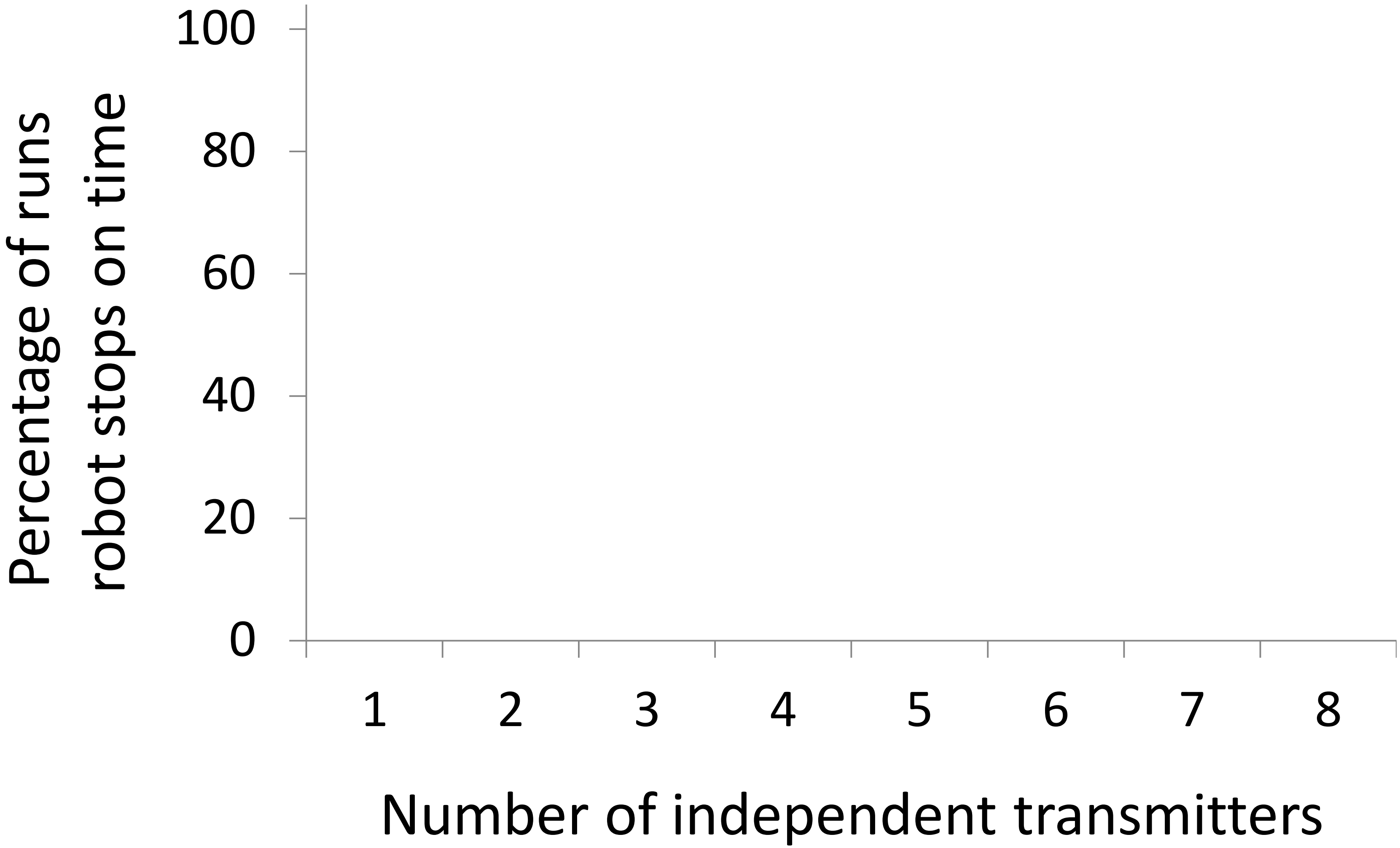
Does CarSpeak Improve Reaction to Objects in Blind Spots?

A drives on road while B pulls out of occluded driveway

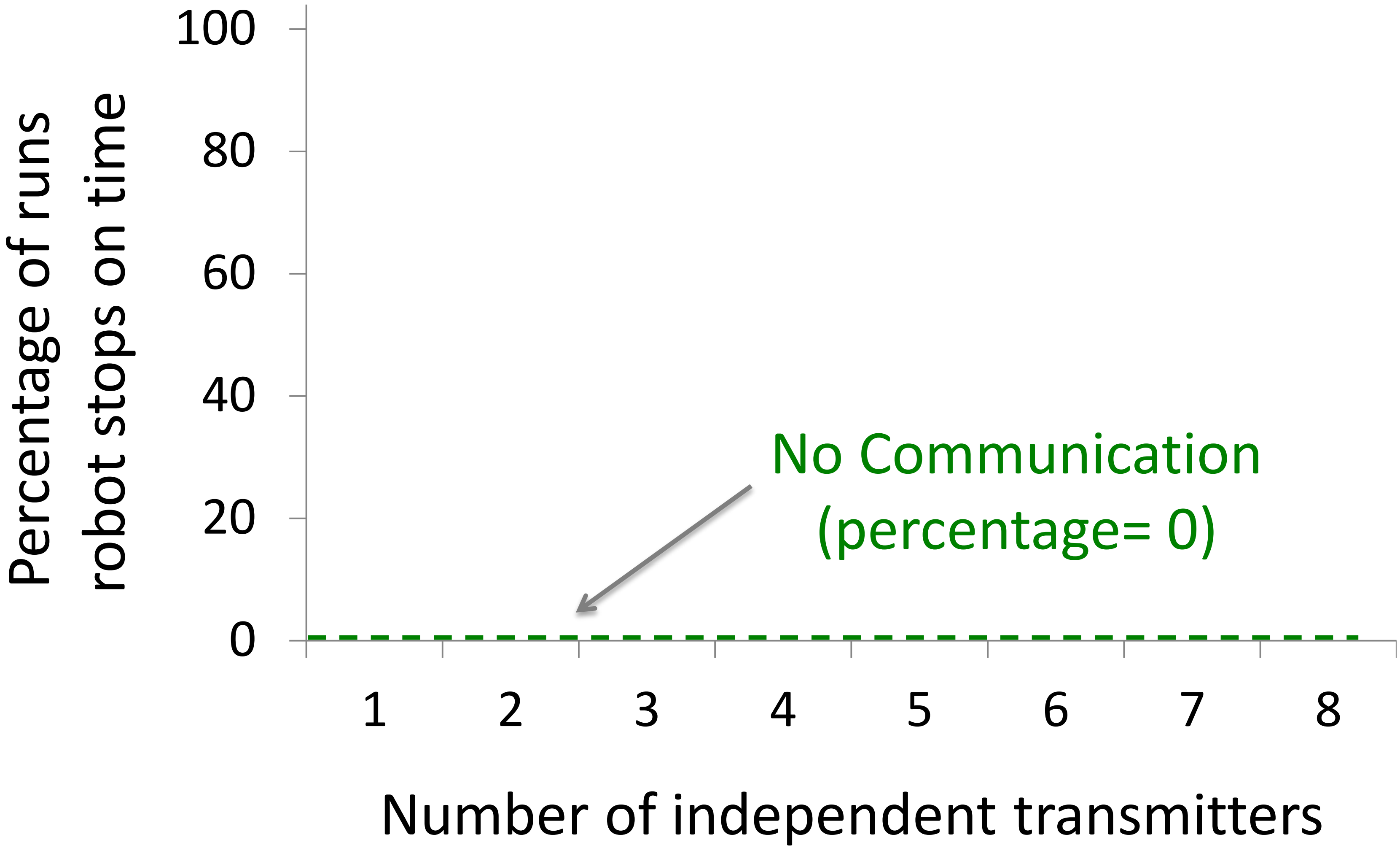
Timely communication can help avoid collisions



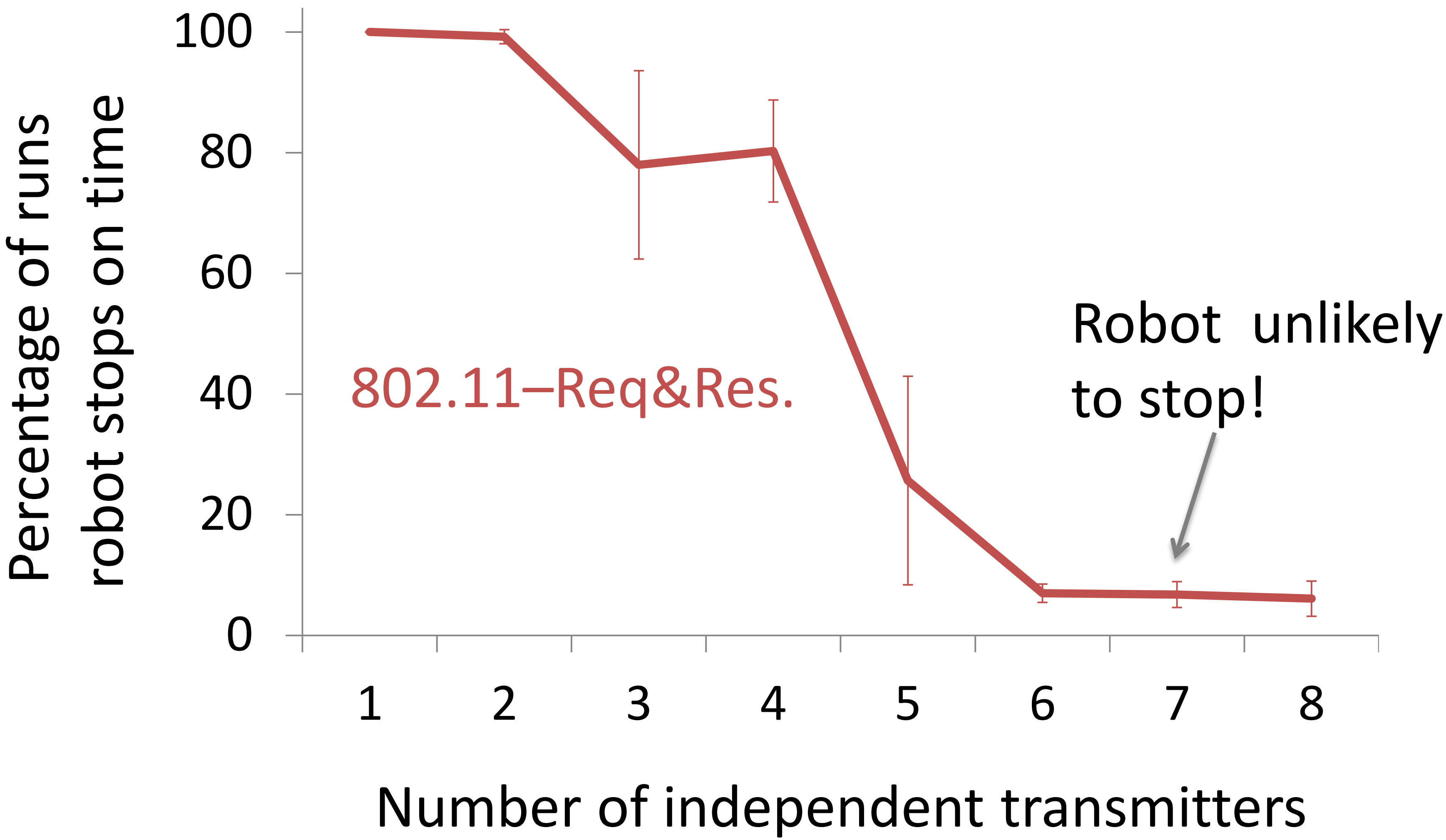
Reaction to Objects in Blind Spots



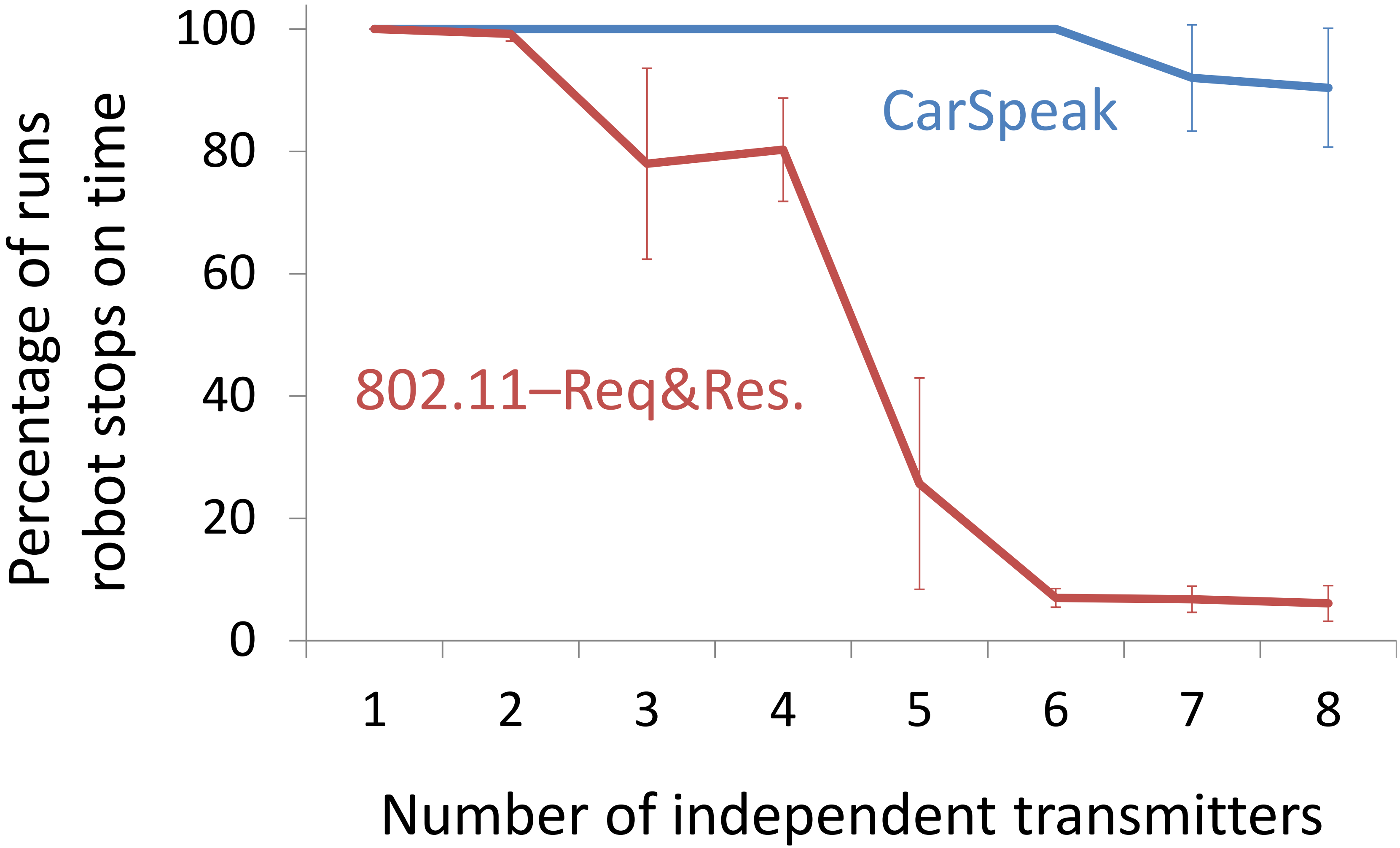
Reaction to Objects in Blind Spots



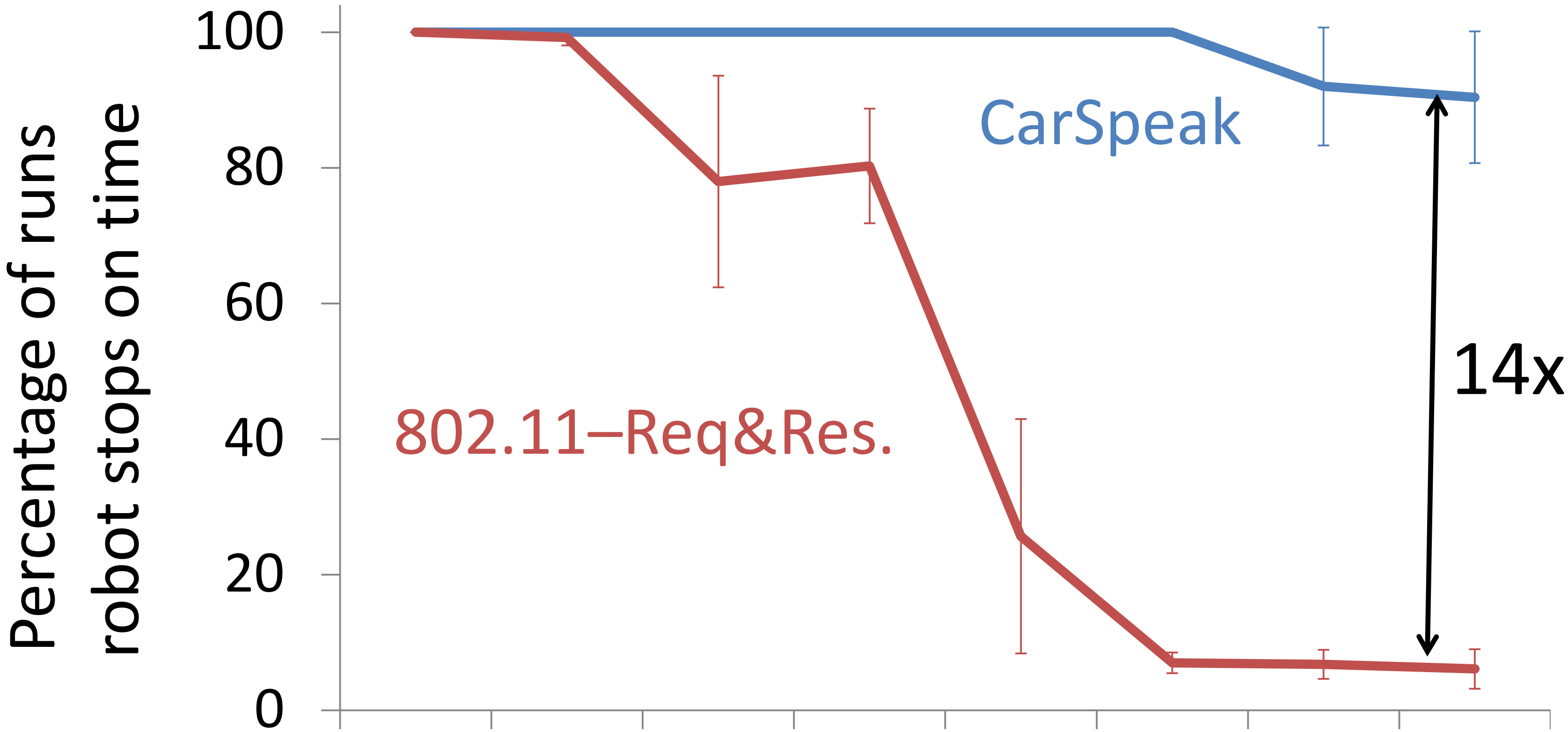
Reaction to Objects in Blind Spots



Reaction to Objects in Blind Spots



Reaction to Objects in Blind Spots



CarSpeak enables vehicles to better deal with hidden objects in blind spots

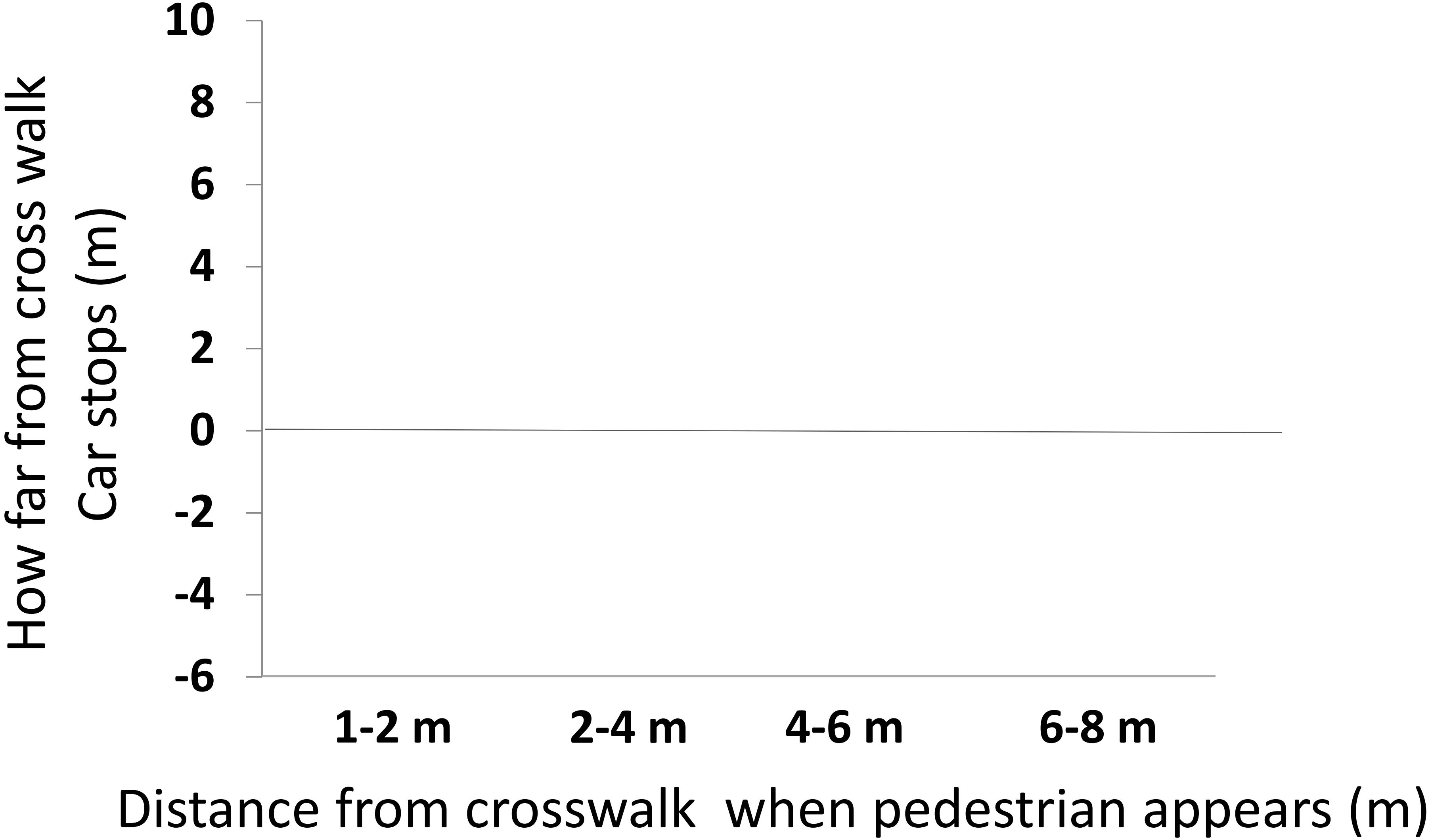
Outdoor Testbed

- Instrumented Yamaha car with laser sensors
- Pedestrian crosswalk in campus-like environment

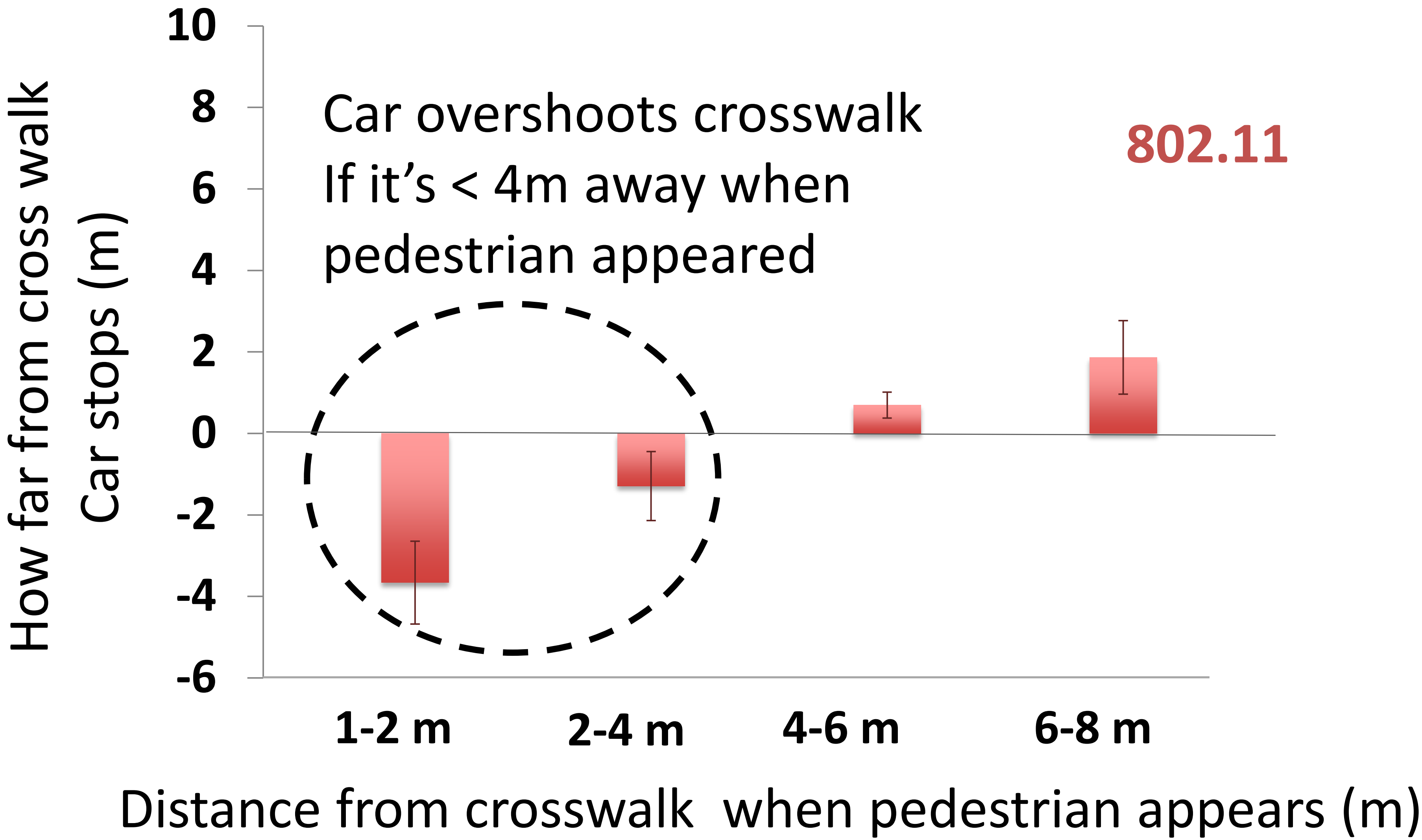
Detecting a Pedestrian in Blind Spot

- Pedestrians emerge from lobby
- Lobby is a blind spot
- Infrastructure sensors, some can send view of lobby

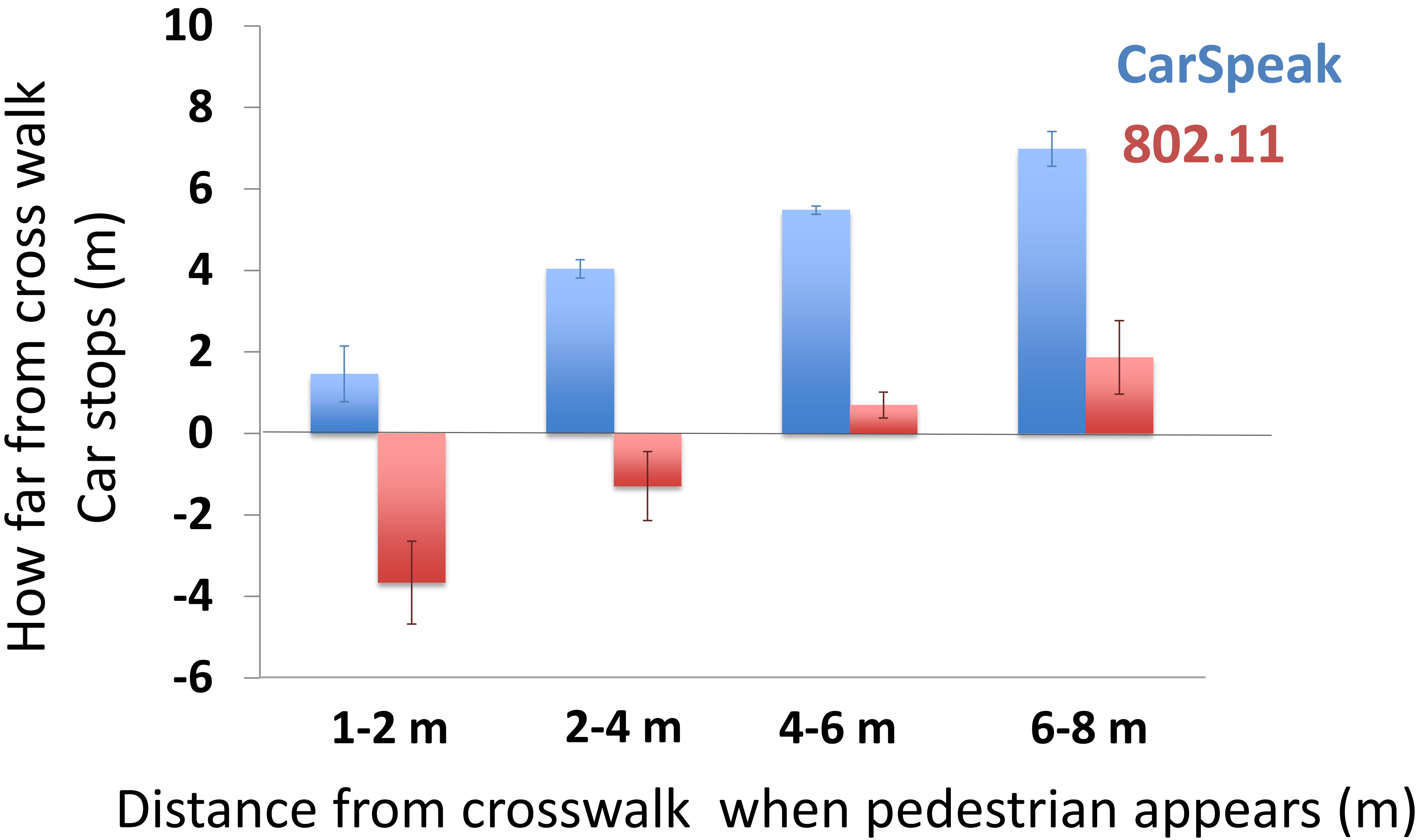
Outdoor Results



Outdoor Results



Outdoor Results



Conclusion

- A communication system fully integrated with autonomous driving
- Content-Centric approach to data access & MAC
- Generally applies to collaborative robotics, virtual reality and virtual games