Catch-Up: A Data Aggregation Scheme for VANETs

Conference： ACM VANET'08

Authors： Bo Yu, Jiayu Gong, Cheng-Zhong Xu

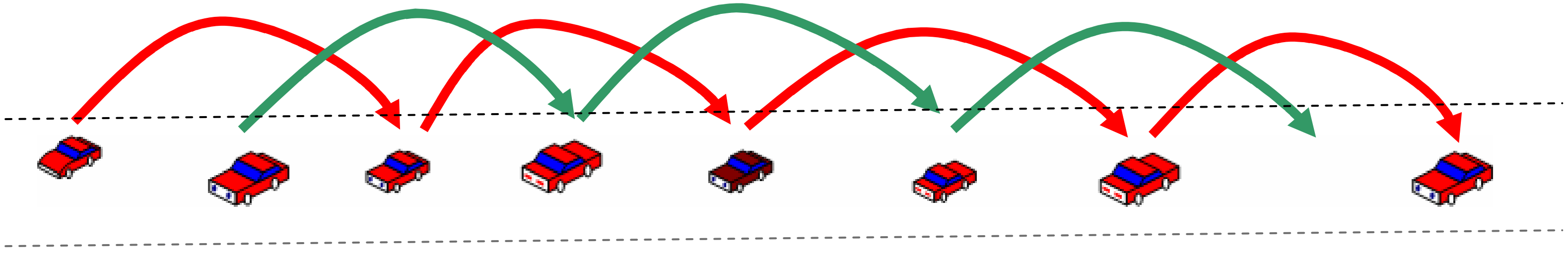
Presenter： 劉哲維

Outline

- Introduction
- Related Work
- Distributed coordinated aggregation
- Analysis for upper bound of overview report
- Simulation
- Conclusion

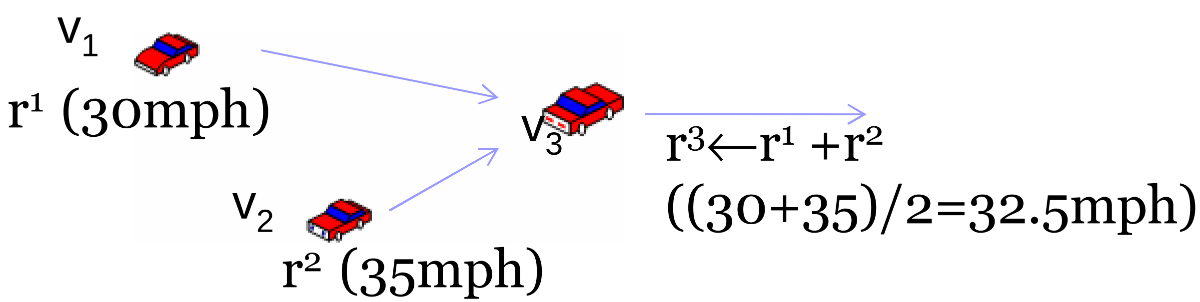
Introduction

- Traffic Information Dissemination
  - Each vehicle periodically detects the traffic conditions around it, and then, forwards the information to vehicles following behind it
  - Like congestion detection
- Redundant Data & Limited Bandwidth
  - Multiple redundant copies for the same traffic status
  - Consuming a considerable amount of bandwidth



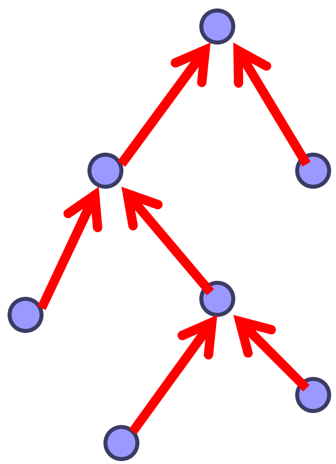
Related Work

- Two aspects
  - Routing-related (our focus)
    - How two reports can meet each other at the same time at the same node
  - Data-related
    - Coding, calculation, and compression of aggregatable data



Routing-related

- Structured Aggregation
  - A routing structure, forwarding tree, is maintained to ensure reports can be forwarded to the same node at the same time
  - Widely used in sensor networks, but infeasible in VANETs



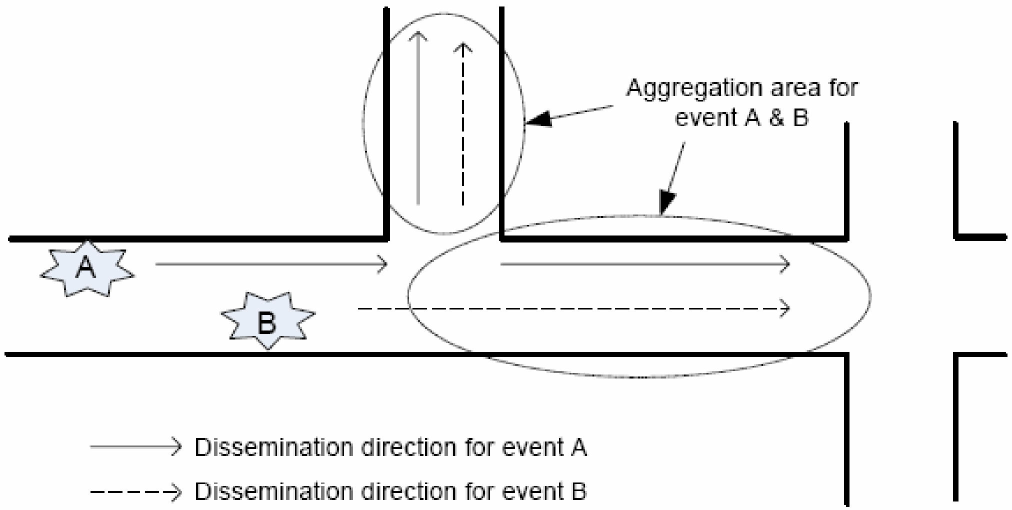
- Structureless Aggregation
  - Randomized Waiting
    - Wait for a random period before forwarding to the next hop
    - During the waiting period, more reports can be received and aggregated
  - Periodical Waiting
    - Wait for a fixed period before forwarding to the next hop
- We adaptively change the forwarding delay of individual reports

Distributed coordinated aggregation

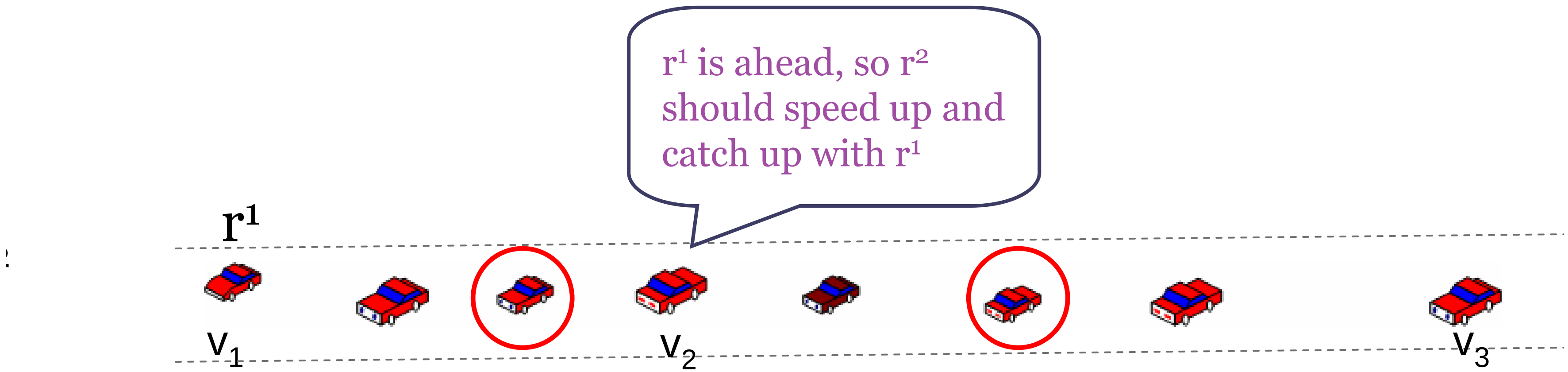
- Two Properties
  - Channel Eavesdropping
    - Every node is able to receive reports being transmitted in the channel and log them into its local database
  - Traffic Information is delay-insensitive
    - Even a delay of tens of seconds is still acceptable
    - Give us an opportunity to trade off increased delay for reduced communication overhead

System Model

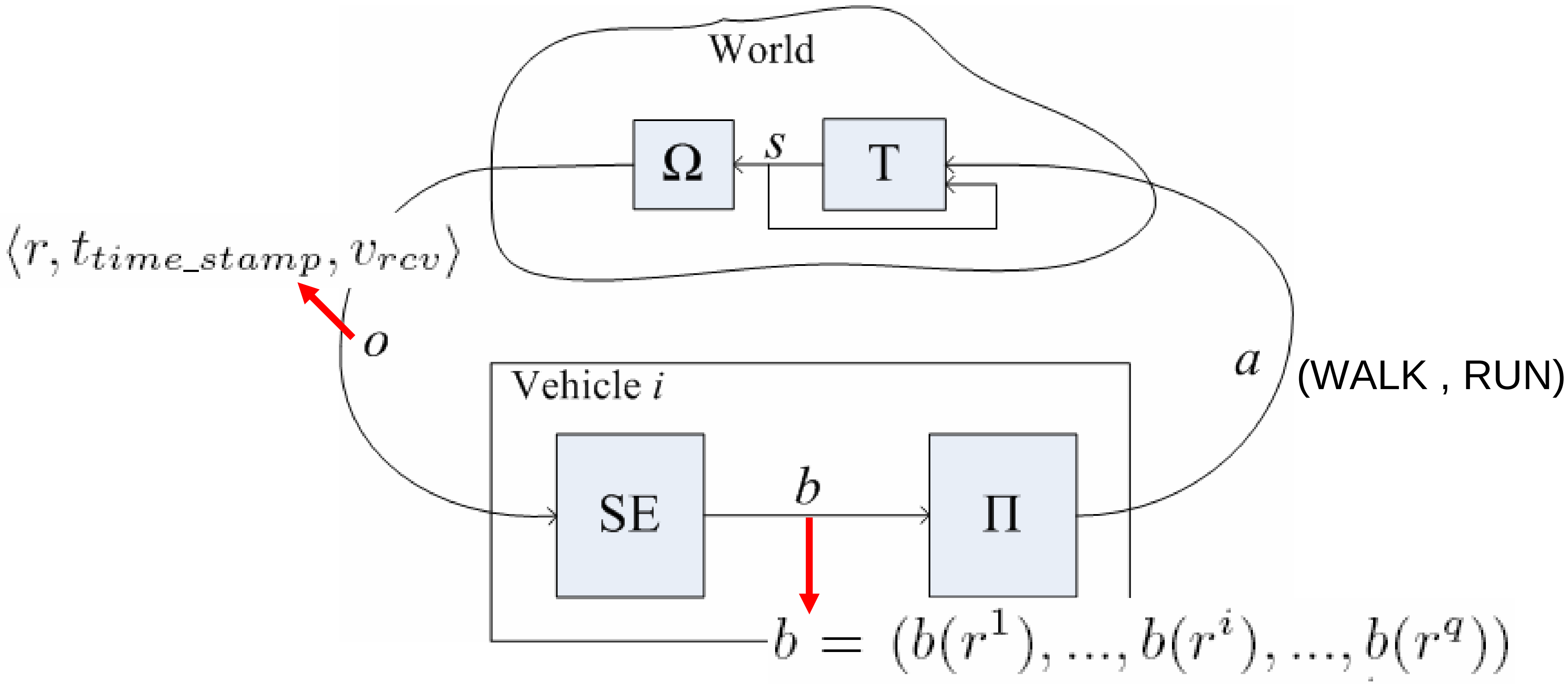
- Aggregatable Reports
  - Generated by a change in traffic condition
  - Define event frame as $< p_1, p_2, t_1, t_2 >$
  - Generate an overview report for all aggregatable reports
- Data Dissemination



- Insert a delay before forwarding a report to the next hop by local observation



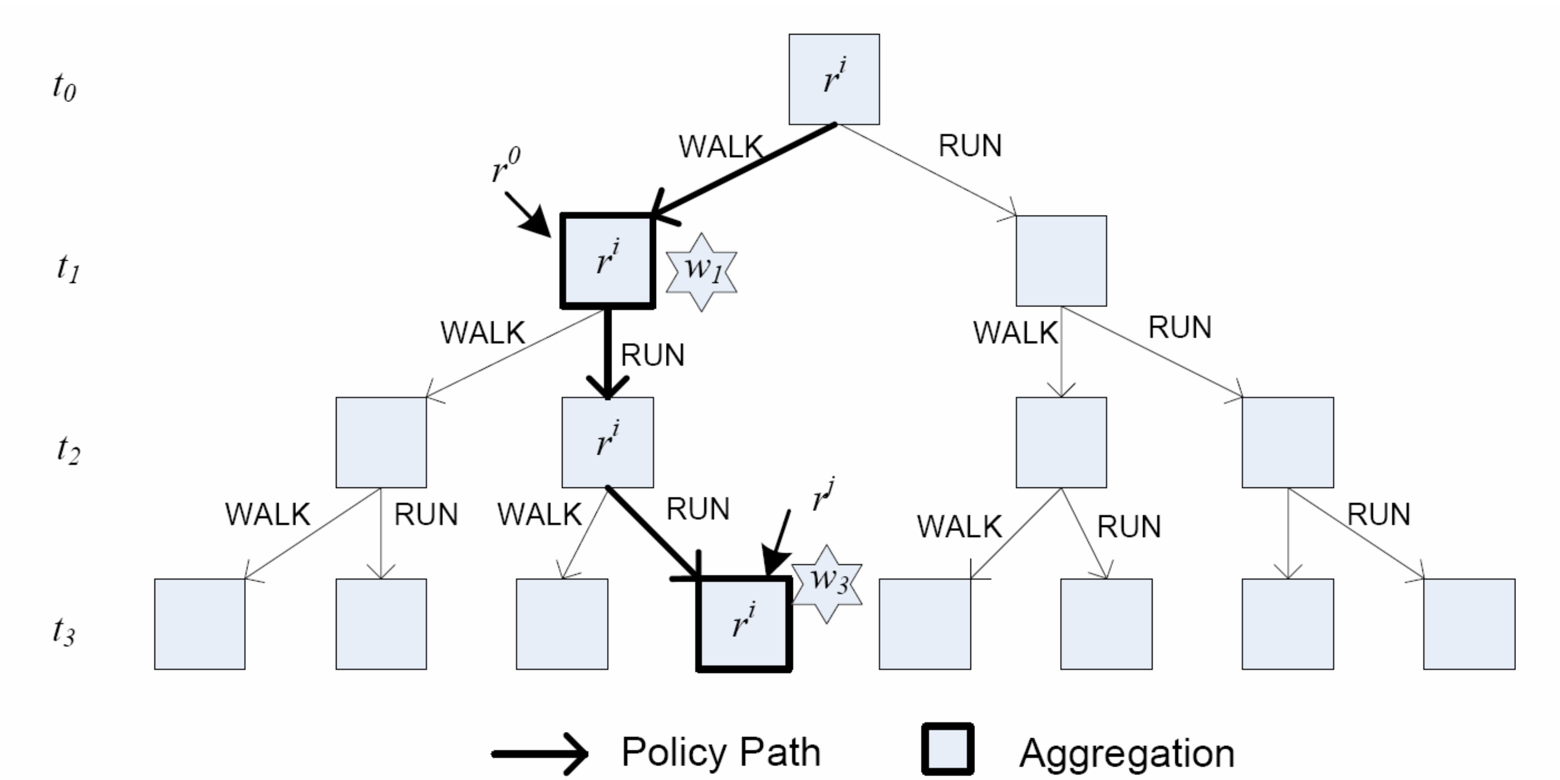
Distributed MDP Model



- s – world state
- a – action
- Π – Decision Maker

Decision Tree

- To find the optimal policy π =(a_0,a_1,a_2,…),a_i = (WALK , RUN)



Expected Future Reward

- Objective
  - To find a policy which maximizes the expected future reward
- Policy π
  - Sequence of actions to be performed for a given report in the future
  - π = (a_1, a_2, …, a_t, …) ,each element is an action of (WALK,RUN)
- Expected Future Reward
  - $$R_b(\pi) = \sum_{t \geq 1} \gamma^t \frac{\sum_{1 \leq j \leq q} b(r^j, p_t) w_t(r^i, r^j)}{q}$$
  - where γ is a future discount factor and w_t means the expected reward at time t (the saved communication overhead due to aggregation of the reports)

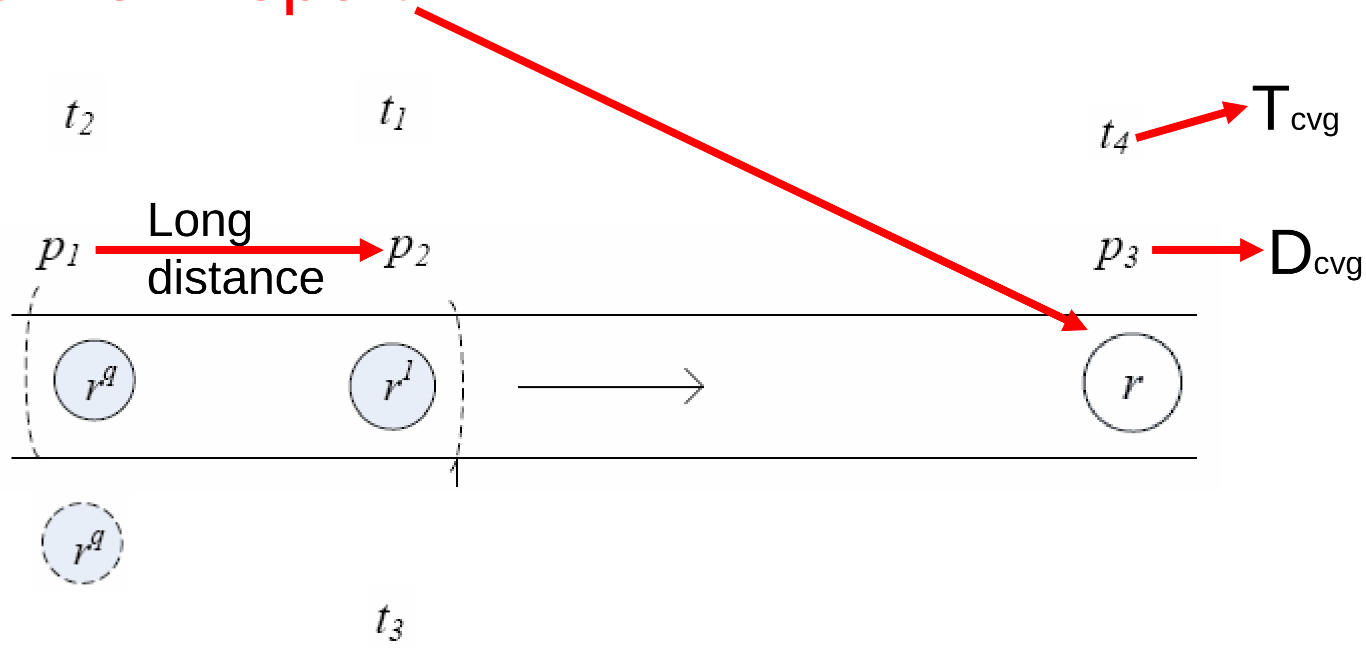
- Virtual Report – r^0
  - To encourage some reports to slow down (WALK)
  - Probability that report r^i and r^0 meet at time t

$$b(r^0, t) = \int_{t_2 \in [t, t+1]} arr(\frac{v_{RUN} - \bar{v}}{v_{RUN}} t_2) \mathbf{d} t_2$$

- Internal States
  - $(r^0, r^1, r^2, r^2, \ldots)$

Analysis

- All reports from a given road section and from a given time period can be aggregated into an overview report



- The convergence time upper bound:

$$T_{cvg} \leq \frac{(p_2 - p_1)v_{RUN} + (t_2 - t_1)v_{RUN}v_{WALK}}{v_{WALK}(v_{RUN} - v_{WALK})}$$

- The convergence distance upper bound:

$$D_{cvg} \leq \frac{(p_2 - p_1)v_{RUN} + (t_2 - t_1)v_{RUN}v_{WALK}}{v_{WALK}(v_{RUN} - v_{WALK})} \cdot v_{WALK}$$

- The total latency upper bound:

$$T_{dism} \leq \max(T_{cvg}) + \frac{D_{dism} - \max(D_{cvg})}{v_{FLY}}$$

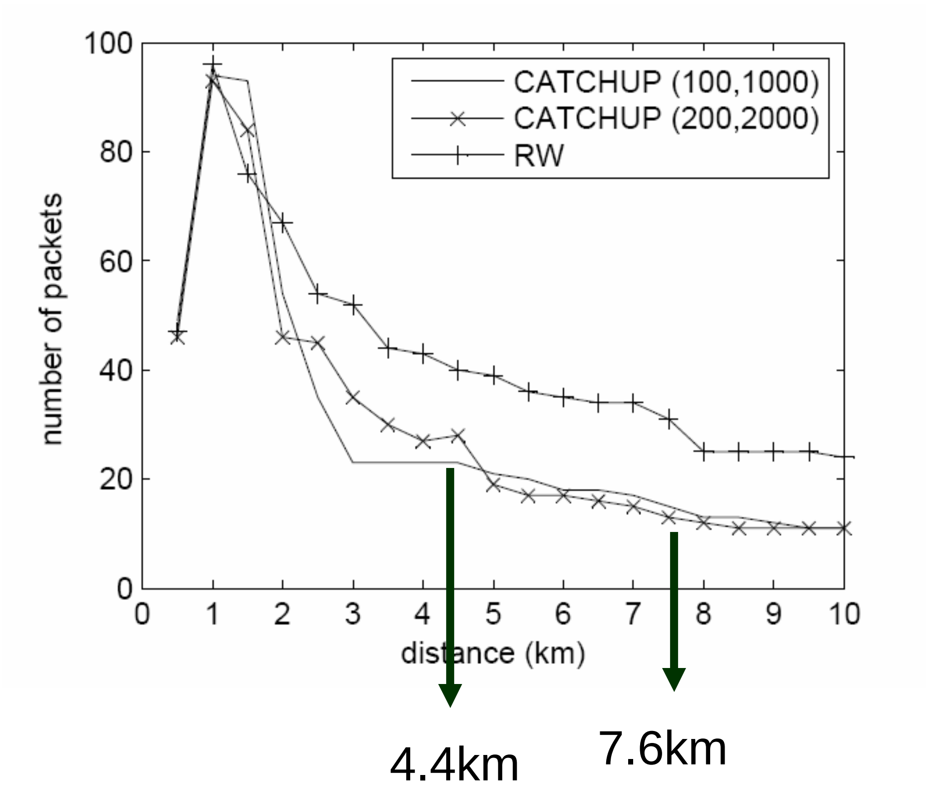
Simulation

- Based on NS2 and GrooveNet
- Compared to Randomized Waiting (RW)

Table 1: Simulation Configurations

System Parameter	Value
Road Length	10km
Vehicle Number	300
Road Section Length	1km
Time Frame	30s
Communication Range	250m
MAC layer	802.11
Mobility Model	StreetSpeedModel[2]
Trip Model	DjikstraTripModel[2]

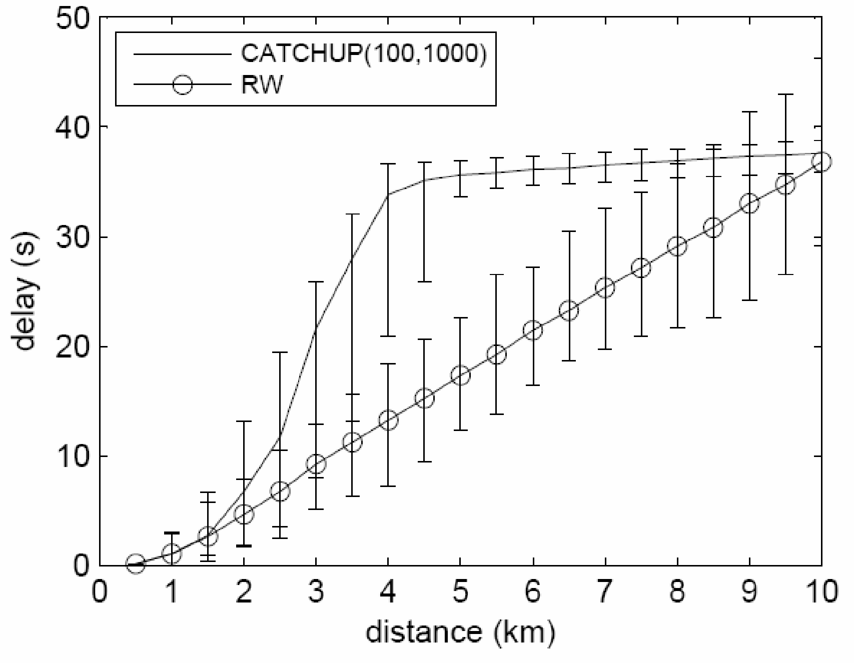
Results



CATCHUP(100,1000) - walking speed at 100m/s, running speed at 1000m/s

CATCHUP(200,2000) - walking speed at 200m/s, running speed at 2000m/s

Protocol	TotalTrans	MaxCvgDist	MaxCvgTime
CATCHUP(100,1000)	1204	4.4km	44s
CATCHUP(200,2000)	1231	7.6km	38s
Randomized Waiting	1944	-	-



- CATCHUP trades increased delay for reduced communication overhead
- For CATCHUP, the delay mainly reside within the first 4 km
- For Randomized Waiting, the delay is linear

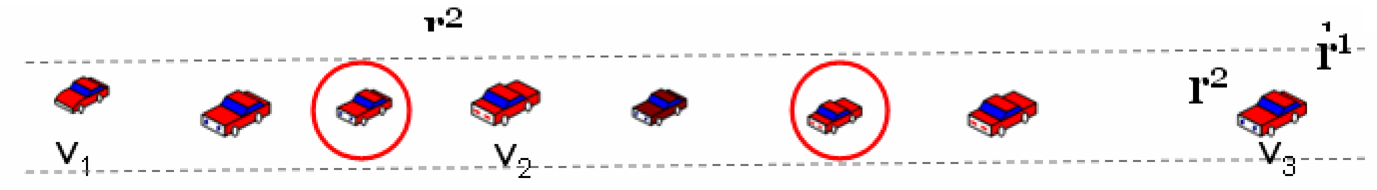
Conclusion

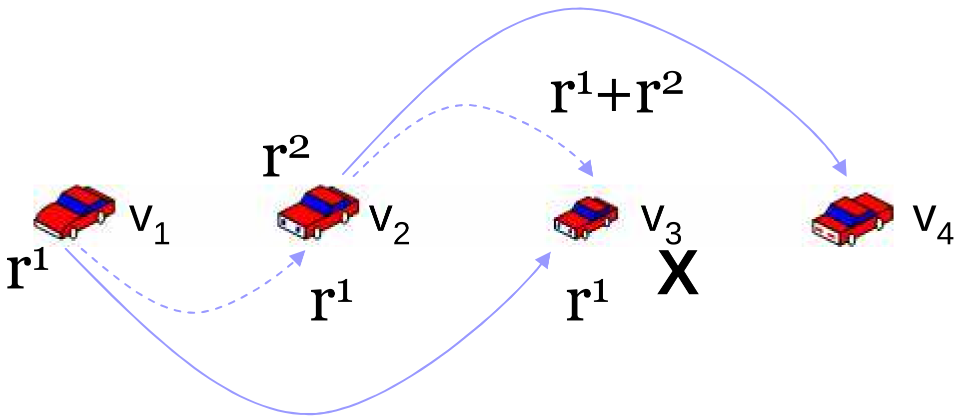
- We introduced a idea of adaptively controlling forwarding delay in data aggregation in VANETs
- Aggregation is a tradeoff between delay and communication overhead

- One problem is what are the optimal delay values for WALK and RUN for our scheme

Other Issues

- Report Overtaking





- How to judge whether a report is contained by another aggregated report?
  - $r^1 \in r^3$? where $r^3=r^1+r^2$
- Bloom Filter
  - A space-efficient probabilistic data structure that is used to test whether an element is a member of a set
  - Each report is attached with a bloom filter table, which includes information of all merged report

Some problems

- Each vehicle can only obtain a partial observation of the world
- We cannot introduce much extra communication overhead for node coordination

- Action a : WALK, RUN
  - The propagation speed of a report
  - Two different delays before forwarding to the next hop
- Observation o : eavesdropped reports
  - Tuple <report, time_stamp, action(WALK/RUN)>
- Internal state b : estimated position of reports
  - $b(r, p_t)$: Probability that report r is at position p at time t