Chapter 8: Deployment & Data Access MAC Scheduling Protocols on VANETs

Optimal Placement of Gateways in Vehicular Networks

Yuguang Fang, Xiaoxia Huang, Pan Li, and Phone Lin

Department of Electrical and Computer Engineering, University of Florida

Department of Computer Science and Information Entineering, National Taiwan University

IEEE Transactions on Vehicular Technology, TVT 2007

Introduction

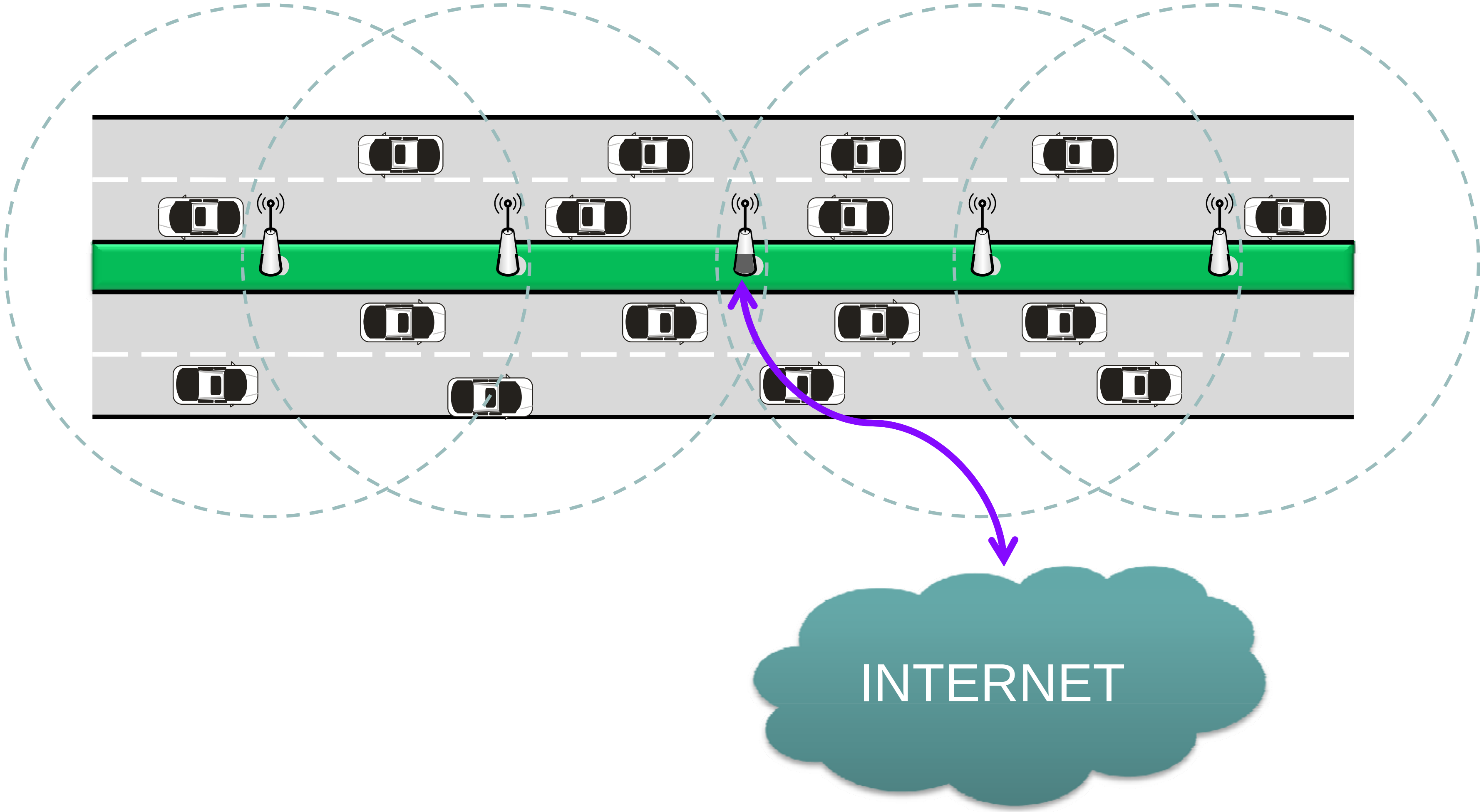
- Mobile users in vehicles require to access the Internet.
- How to place gateways to link mobile nodes to the Internet ?

Goals

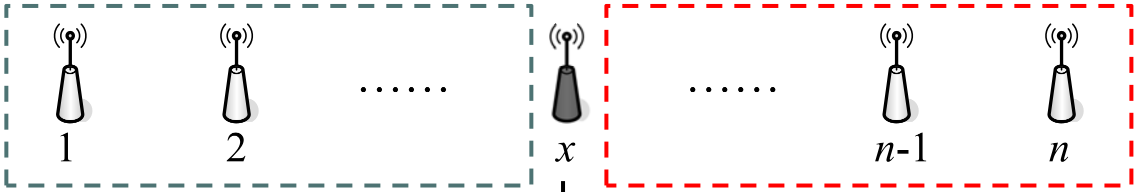
- Optimally placing gateways.
  - Minimize the average number of hops from APs to gateways.
  - Minimize the total power consumption.
  - Maximize the average capacity of each AP.

1-D Vehicular Networks

Assume that several APs have been deployed.
Which AP should play the role of gateway?



One Gateway



$$H(x)=\frac{1}{n}[(x-1)+(x-2)+\ldots+1+0+1+2+\ldots+(n-x)]$$

Average number of hops

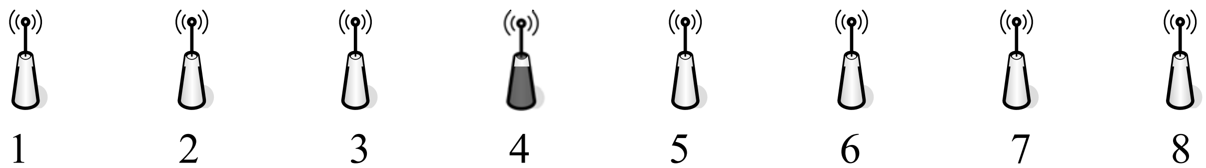
$$H(x)=\frac{1}{n}\left[\left(x-\frac{n+1}{2}\right)^2+\frac{n^2-1}{4}\right]$$

To minimize *H(x), we have*

$$x=\left\lfloor\frac{n+1}{2}\right\rfloor \quad or \quad x=\left\lceil\frac{n+1}{2}\right\rceil$$

One Gateway

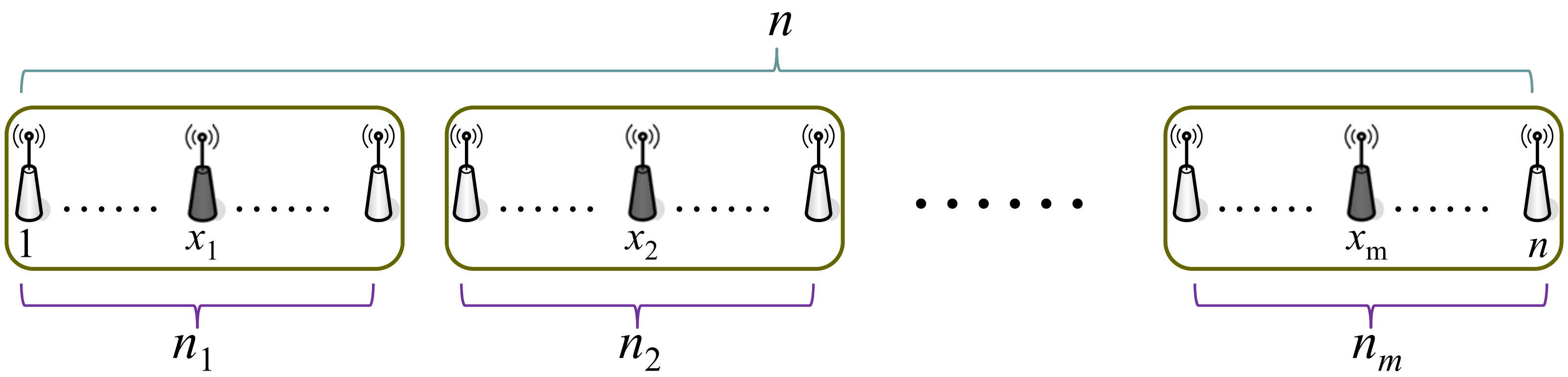
$$n = 9 \Rightarrow x = \left\lfloor \frac{n+1}{2} \right\rfloor = 5$$



$$n = 8 \Rightarrow x = \left\lfloor \frac{n+1}{2} \right\rfloor = 4$$

Multiple Gateways

How to select *m* gateways from *n* APs?



$$H_i(x_i) = \frac{1}{n_i}\left[\left(x_i - \frac{n_i + 1}{2}\right)^2 + \frac{n_i^2 - 1}{4}\right]$$

Average number of hops of each group.

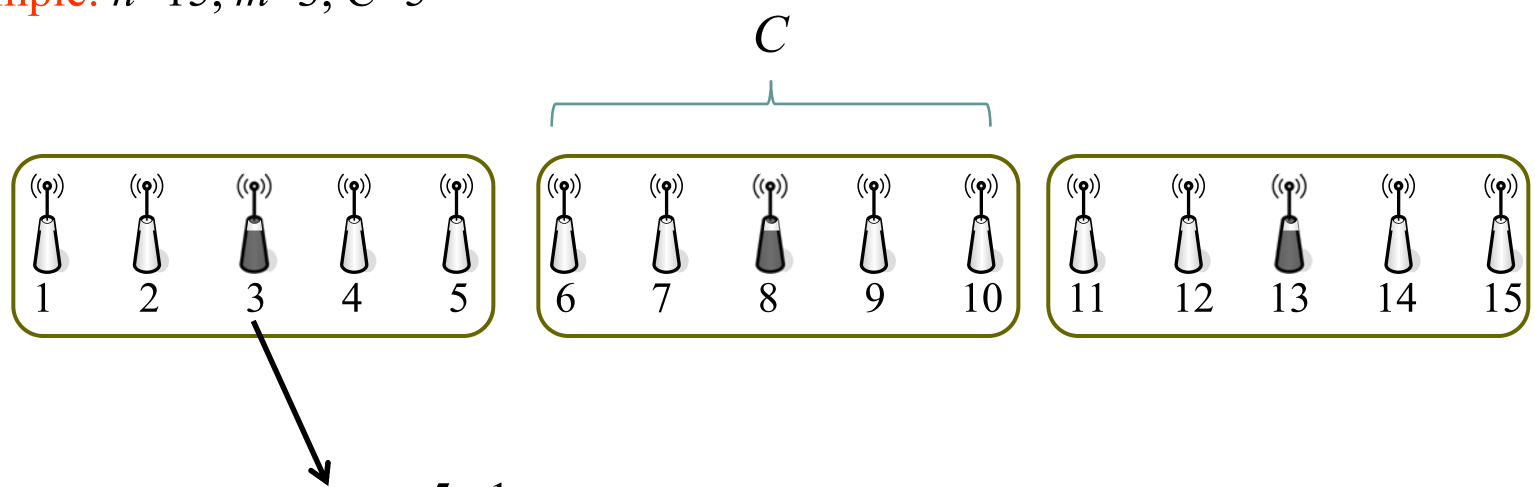
Multiple Gateways

Let $C = \frac{n}{m}$ be the number of APs in a group

Case1: C is an odd integer:

$$x_i = \frac{C+1}{2} + (i-1)C$$

Example: n=15, m=3, C=5



$$x_1 = \frac{5+1}{2}$$

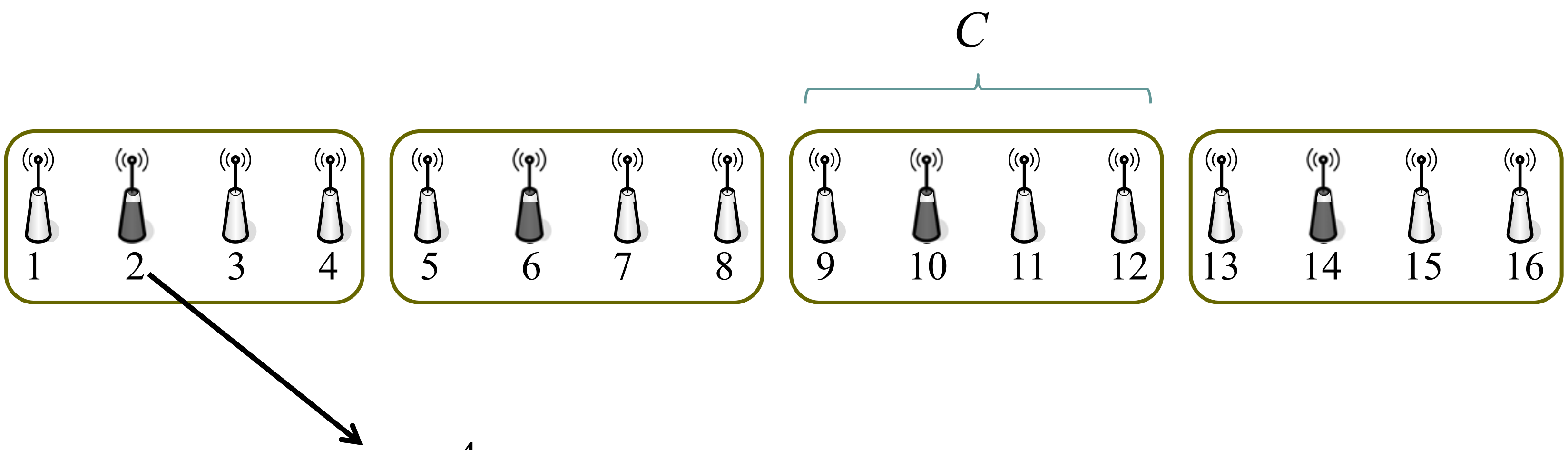
Multiple Gateways

Let $C = \frac{n}{m}$ be the number of APs in a group

Case2: C is an even integer:

$$x_i = \frac{C}{2} + (i-1)C$$

Example: n=16, m=4, C=4



$$x_1 = \frac{4}{2}$$

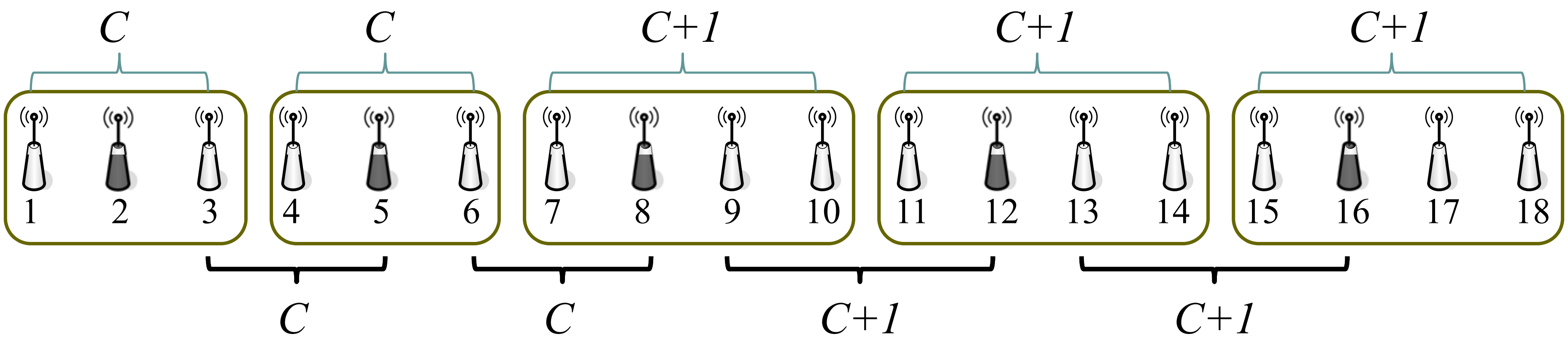
Multiple Gateways

Let $C = \frac{n}{m}$ be the number of APs in a group

Case3: C is not an integer but exists $k(1 \leqq k \leqq m)$ such that $C=(n-k)/m$ is an odd integer :

$$x_i = \begin{cases} \frac{C+1}{2} + (i-1)C & , if\ 1 \le i \le m-k+1 \\ x_{m-k+1} + [i-(m-k+1)](C+1) & , if\ m-k+2 \le i \le m \end{cases}$$

Example: n=18, m=5, C=3



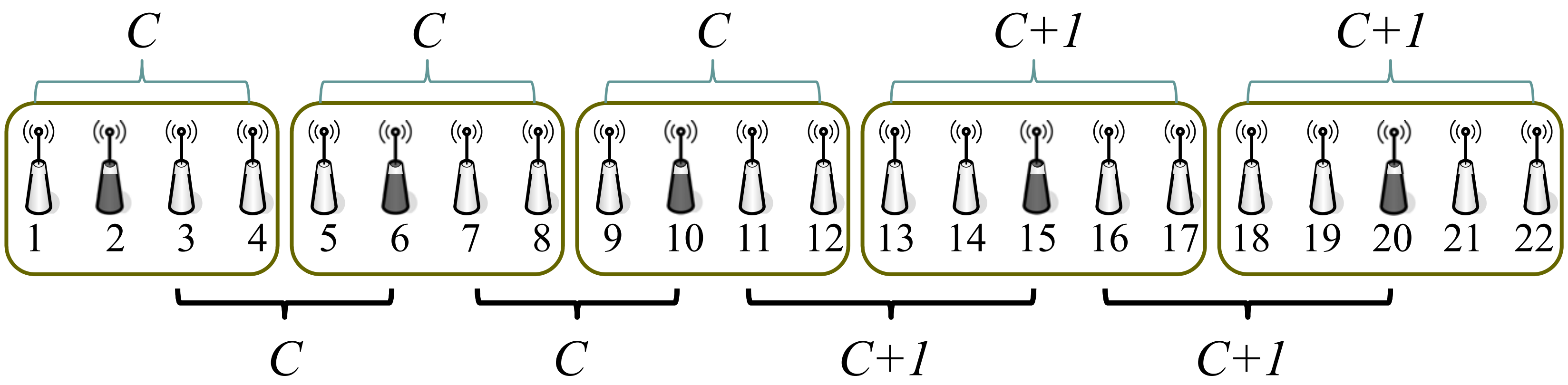
Multiple Gateways

Let $C = \frac{n}{m}$ be the number of APs in a group

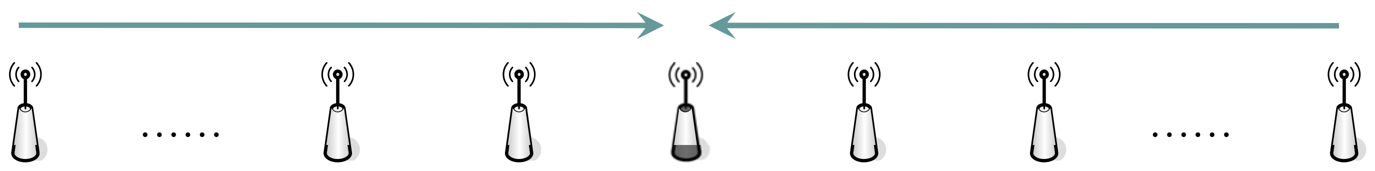
Case3: C is not an integer but exists $k(1 \leqq k \leqq m)$ such that $C=(n-k)/m$ is an even integer :

$$x_i = \begin{cases} \frac{C}{2} + (i-1)C & , if\ 1 \le i \le m-k \\ x_{m-k} + [i-(m-k)](C+1) & , if\ m-k+1 \le i \le m \end{cases}$$

Example: n=22, m=5, C=4



Energy Balance



越靠近 Gateway的AP所要代傳封包量越多，耗費越多電量

Energy Balance

T. Rappaport, *Wireless Communications: Principles and Practice*, 2nd ed. Prentice-Hall PTR, 2002.

$$P_r = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 L d^\gamma}$$

P_t : transmitted power

P_r : received power

G_t : antenna gains for the transmitter

G_r : antenna gains for the receiver

$L (L \geq 1)$: the system loss factor

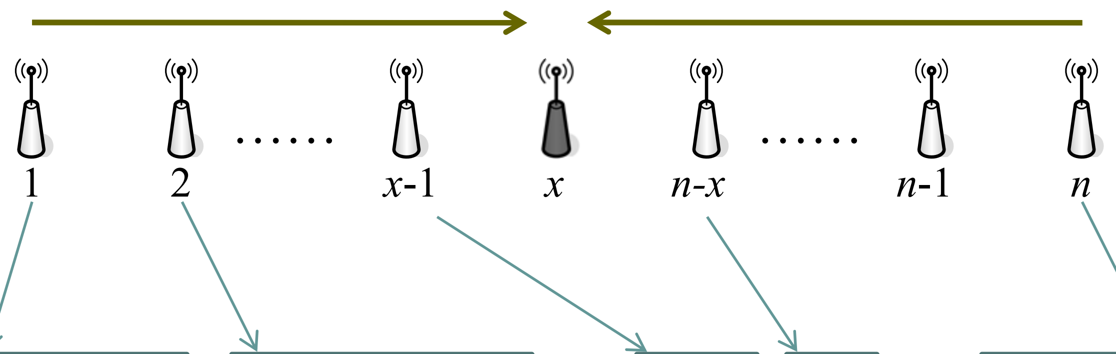
λ : the wavelength

γ : the path loss exponent

Energy Balance

Given n APs and the gateway has been known,
how to determine the distance of neighboring APs for achieving energy balance?

$$P_r = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 L d^\gamma} \Rightarrow P_t = \boxed{P_r \frac{(4\pi)^2 L}{G_t G_r \lambda^2}} d^\gamma = T d^\gamma$$



$$P = \boxed{(d_1^\gamma + d_2^\gamma + \ldots + d_{x-1}^\gamma)T} + \boxed{(d_2^\gamma + \ldots + d_{x-1}^\gamma)T} + \ldots + \boxed{d_{x-1}^\gamma T} + \boxed{d_x^\gamma T} + \ldots + \boxed{(d_x^\gamma + d_{x+1}^\gamma + \ldots + d_{n-1}^\gamma)T}$$

越靠近 Gateway的AP所要代傳封包量越多，耗費越多電量➔ AP 間距離越小!

One Gateway

$$P = (d_1^\gamma + d_2^\gamma + \ldots + d_{x-1}^\gamma)T + (d_2^\gamma + \ldots + d_{x-1}^\gamma)T + \ldots + d_{x-1}^\gamma T + d_x^\gamma T + \ldots + (d_x^\gamma + d_{x+1}^\gamma + \ldots + d_{n-1}^\gamma)T$$

$$P = \{[d_1^\gamma + 2d_2^\gamma + \ldots + (x-1)d_{x-1}^\gamma] + [d_{n-1}^\gamma + 2d_{n-2}^\gamma + \ldots + (n-x)d_x^\gamma]\}T$$

$$P = \left(\sum_{i=1}^{x-1} i d_i^\gamma + \sum_{j=1}^{n-x} j d_{n-j}^\gamma \right) T$$

One Gateway

算術平均數 ≥ 幾何平均數

$$\frac{\sum_{i=1}^{x-1} id_i^{\gamma} + \sum_{j=1}^{n-x} jd_{n-j}^{\gamma}}{n-1} \geq \left\{\left(\prod_{i=1}^{x-1} id_i^{\gamma}\right)\left(\prod_{j=1}^{n-x} jd_{n-j}^{\gamma}\right)\right\}^{\frac{1}{n-1}}$$

$$P \geq (n-1)\left\{\left(\prod_{i=1}^{x-1} id_i^{\gamma}\right)\left(\prod_{j=1}^{n-x} jd_{n-j}^{\gamma}\right)\right\}^{\frac{1}{n-1}} T$$

The equality holds when $id_i^{\gamma} = jd_{n-j}^{r}$ for all $i \in [1, x-1]$ and $j \in [1, n-x]$

$$P = \left(\sum_{i=1}^{x-1} id_i^{\gamma} + \sum_{j=1}^{n-x} jd_{n-j}^{\gamma}\right) T$$

One Gateway

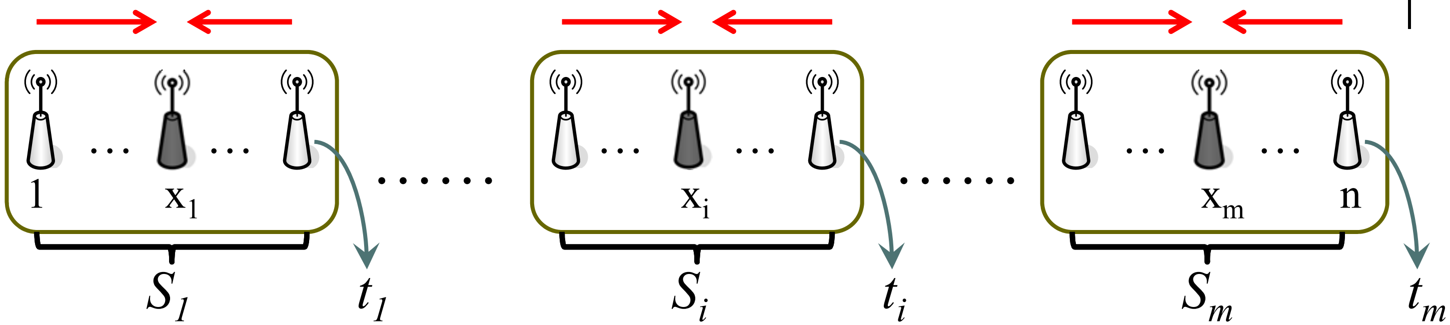
Since $\sum_{i=1}^{x-1} d_i + \sum_{j=1}^{n-x} d_j = L$

The minimum value of P can be achieved when

$$d_i = \begin{cases} \left(\sum_{i=1}^{x-1} i^{-\frac{1}{\gamma}} + \sum_{i=x}^{n-1} (n-i)^{-\frac{1}{\gamma}} \right)^{-1} L , & for\ i = 1 \\ i^{-\frac{1}{\gamma}} d_1 & , for\ i \in [2, x-1] \\ (n-i)^{-\frac{1}{\gamma}} d_1 & , for\ i \in [x, n-1] \end{cases}$$

越靠近 Gateway的AP所要代傳封包量越多，耗費越多電量➔ AP 間距離越小!

Multiple Gateways



$$P=\{[d_1^\gamma+2d_2^\gamma+\ldots+(x_1-1)d_{x_1-1}^\gamma]+[d_{t_1-1}^\gamma+2d_{t_1-2}^\gamma+\ldots+(t_1-x_1)d_{x_1}^\gamma]\}T$$

$$+\{[d_{t_1+1}^\gamma+2d_{t_1+2}^\gamma+\ldots+(x_2-1)d_{x_2-1}^\gamma]+[d_{t_2-1}^\gamma+2d_{t_2-2}^\gamma+\ldots+(t_2-x_2)d_{x_2}^\gamma]\}T$$

$$+\ldots$$

$$+\{[d_{t_{m-1}+1}^\gamma+2d_{t_{m-1}+2}^\gamma+\ldots+(x_m-1)d_{x_m-1}^\gamma]+[d_{n-1}^\gamma+2d_{n-2}^\gamma+\ldots+(n-x_m)d_{x_m}^\gamma]\}T$$

$$P=\sum_{k=1}^{m}\left(\sum_{i=1}^{x_k-1} i d_{t_{k-1}+i}^\gamma+\sum_{j=1}^{t_k-x_k} j d_{t_k-j}^\gamma\right)T$$

Multiple Gateways

$$P \geq (n-1)\prod_{k=1}^{m}\left(\left(\prod_{i=1}^{x_k-1} id_{t_{k-1}+i}^{\gamma}\right)\left(\prod_{j=1}^{t_k-x_k} jd_{t_k-j}^{\gamma}\right)\right)^{\frac{1}{n-1}} T$$

The equality holds when $id_{t_{k-1}+i}^{\gamma} = jd_{t_l-j}^{r}$ for all $i \in [1, x_k - 1]$, $j \in [1, t_k - x_k]$, and $k, l \in [1, m]$

$$P = \sum_{k=1}^{m}\left(\sum_{i=1}^{x_k-1} id_{t_{k-1}+i}^{\gamma} + \sum_{j=1}^{t_k-x_k} jd_{t_k-j}^{\gamma}\right) T$$

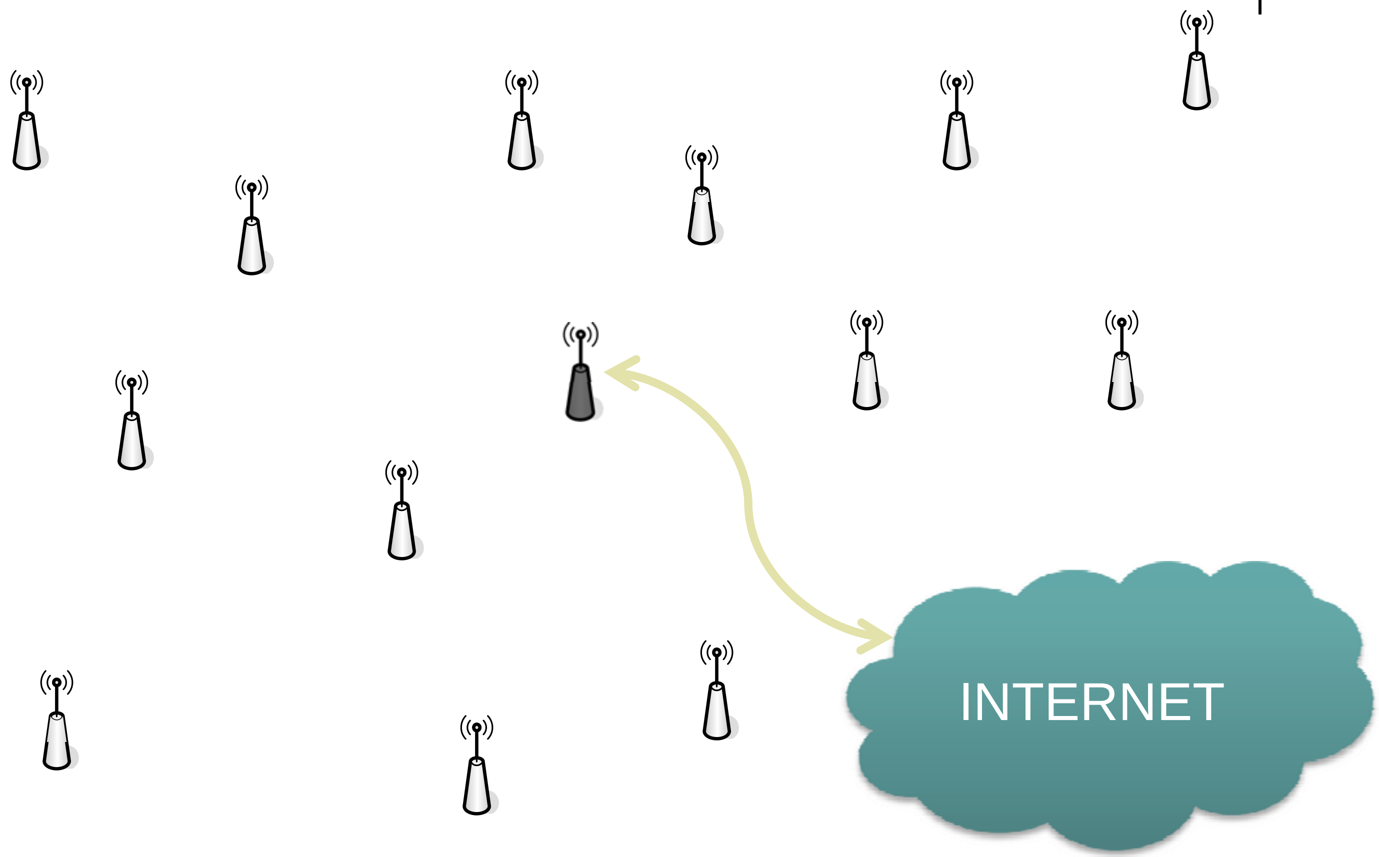
Multiple Gateways

$$\text{Since } \sum_{k=1}^{m}\left(\sum_{i=1}^{x_k-1} d_{t_{k-1}+i} + \sum_{j=1}^{t_k-x_k} d_{t_k-j}\right) = L$$

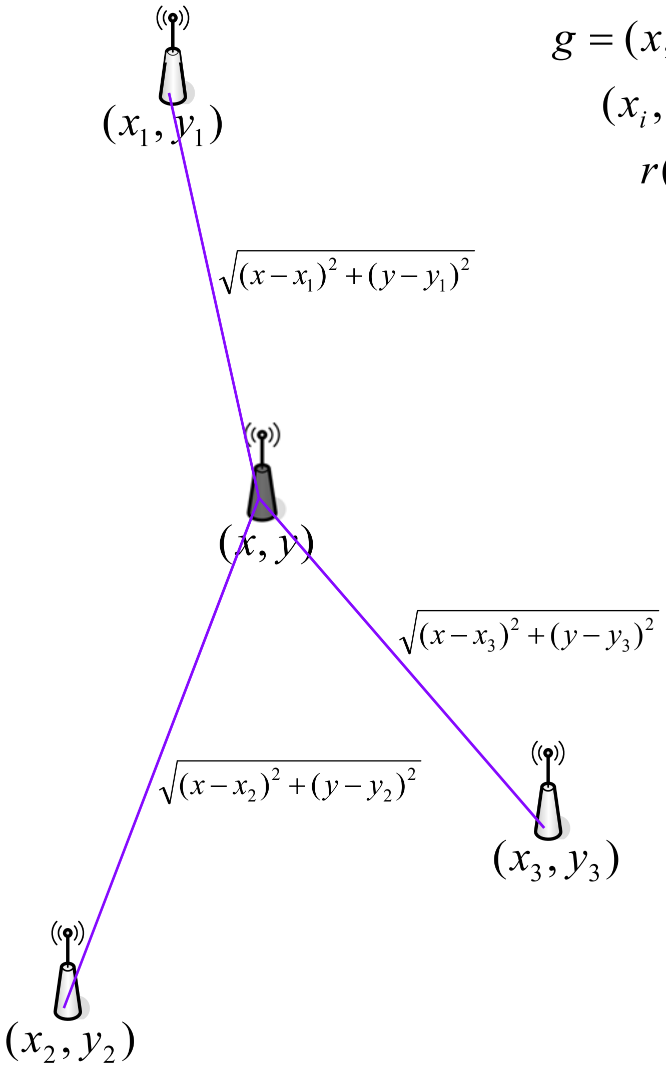
The minimum value of P can be achieved when

$$d_i = \begin{cases} \left(\sum_{k=1}^{m}\left(\sum_{i=1}^{x_k-1} i^{-\frac{1}{\gamma}} + \sum_{j=x_k}^{t_k-x_k} (t_k - j)^{-\frac{1}{\gamma}}\right)\right)^{-1} L \text{ , } for\ i = t_k + 1 \\ i^{-\frac{1}{\gamma}} d_1 \text{ , } for\ i \in [t_k + 2, x_{k+1} - 1] \\ (t_{k+1} - i)^{-\frac{1}{\gamma}} d_1 \text{ , } for\ i \in [x_{k+1}, t_{k+1} - 1] \end{cases}$$

2-D Vehicular Networks



One Gateway



$g=(x,y)$ denotes the location of the gateway

(x_i, y_i) denotes the location of AP i

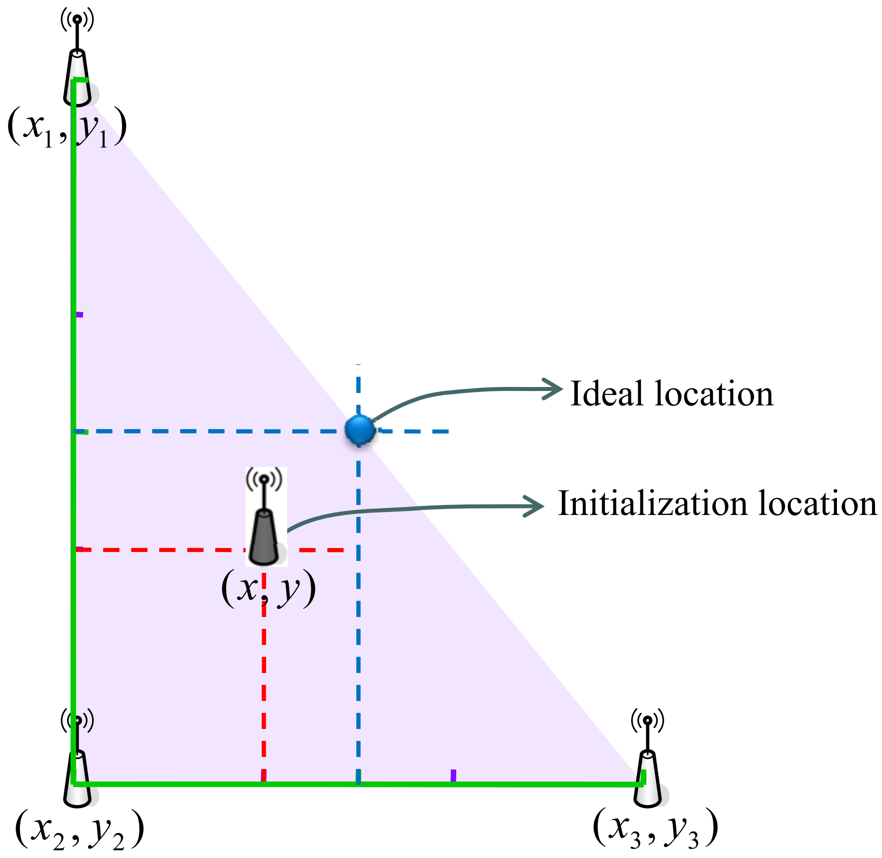
$r(n)$ denotes the common transmission range for all n APs

The number of hops from AP_i to gateway $\geq \left\lceil \frac{\sqrt{(x-x_i)^2+(y-y_i)^2}}{r(n)} \right\rceil$

To minimize *H(g), we have*

$$\text{Minimize} \left\{ f(x,y) = \sum_{i=1}^{n} \sqrt{(x-x_i)^2+(y-y_i)^2} \right\}$$

One Gateway



1. Initialization:

$$x = \frac{1}{n}\sum_{i=1}^{n} x_i$$

$$y = \frac{1}{n}\sum_{i=1}^{n} y_i$$

→ Not the ideal location

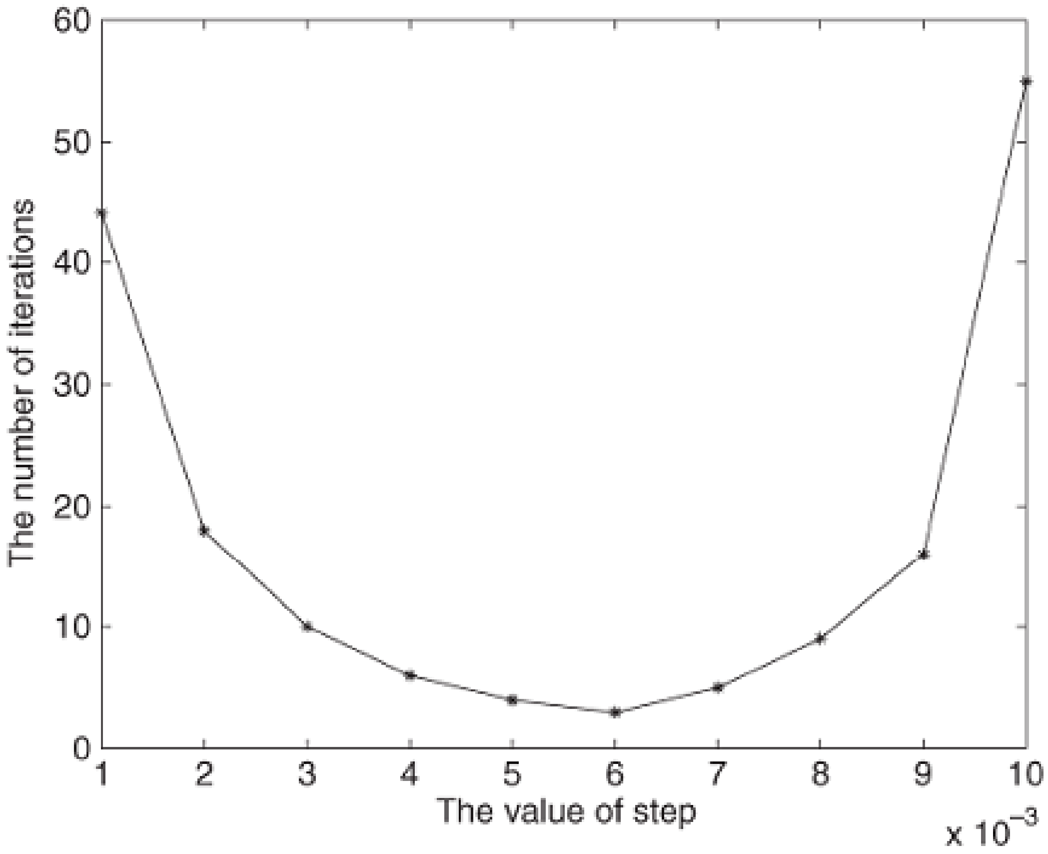
2. Iteration: calculate *Scos* and *Ssin*, where

$$Scos = \sum_{i=1}^{n} \frac{(x_i - x)}{\sqrt{(x-x_i)^2 + (y-y_i)^2}}$$

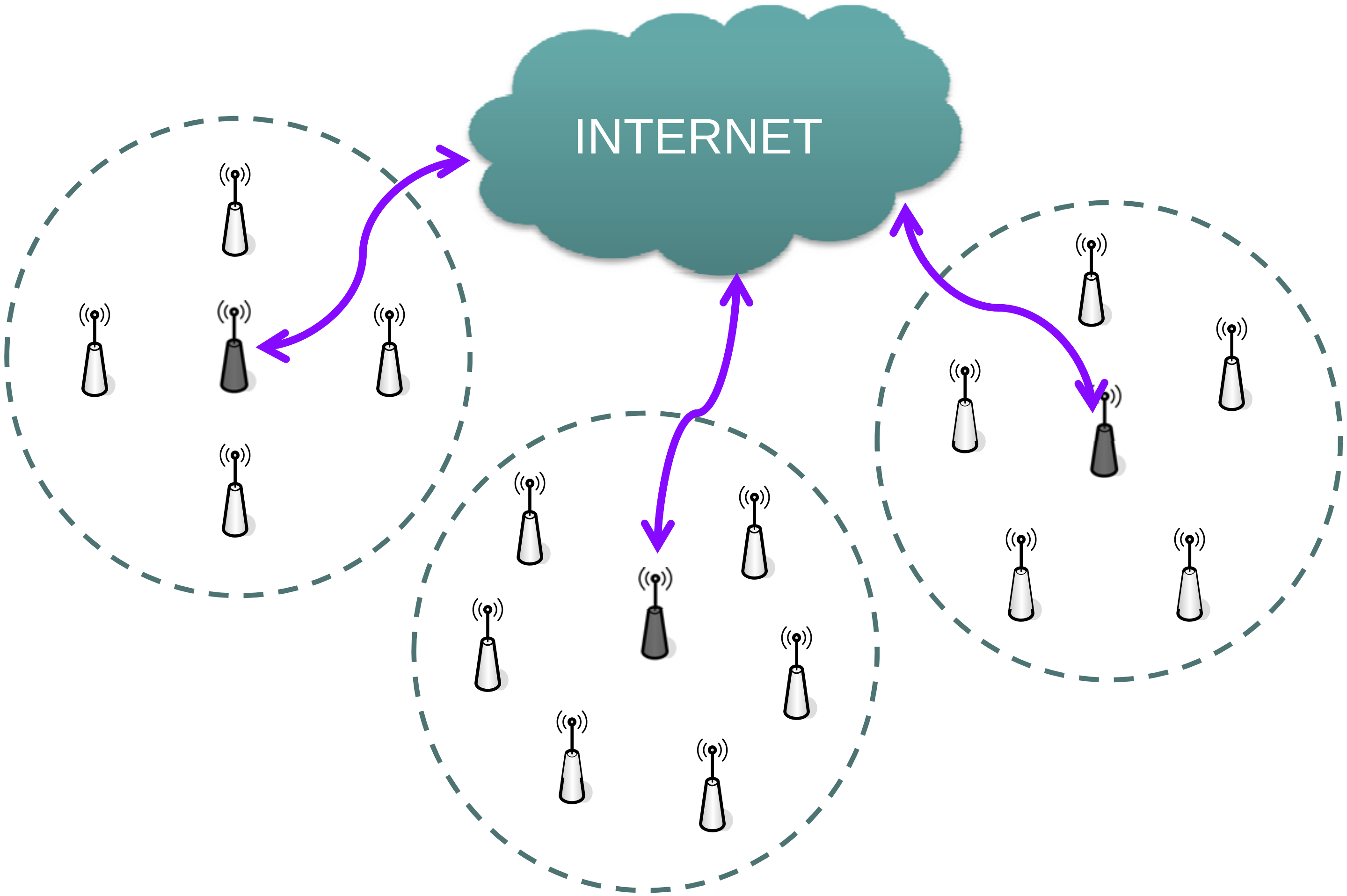
$$Ssin = \sum_{i=1}^{n} \frac{(y_i - y)}{\sqrt{(x-x_i)^2 + (y-y_i)^2}}$$

3. IF |*Scos*|≤*Threshold* and |*Ssin*|≤*Threshold*
4. Go to 9.
5. Else
6. *x*=*x*+*step***Scos* and *y*=*y*+*step***Ssin*
7. Go to 2.
8. END IF
9. Finish

One Gateways



Multiple Gateways



Discussion-Power

$$P = P_t H(G)$$

P_t is the transmission power, which is the same for all APs,
G is the gateways in the network
$H(G)$ is the average number of hops from an AP to a gateway

When $H(G)$ is minimized, the average transmission power P is also minimized.

Discussion-Capacity

$$\lambda \cdot H(G) \cdot n \leq C$$

λ denotes the AP generates traffics.
C denotes the capacity of the network.
H(*G*) is the average number of hops from an AP to a gateway.
n is the total number of APs in the network.

Therefore, the capacity available to each node, λ, is bounded by

$$\lambda \leq \frac{C}{nH(G)} \leq \frac{C}{nH_{\min}(G)}$$

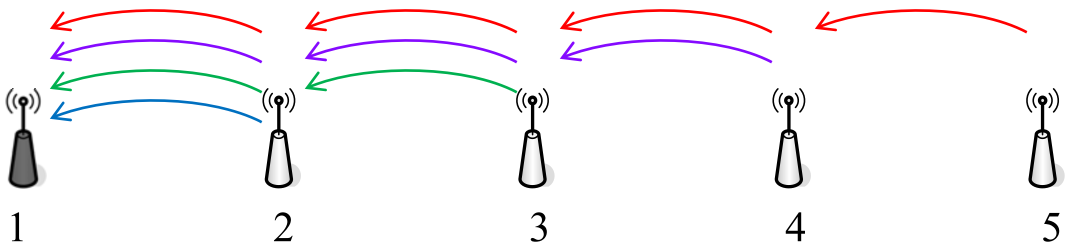
When *H(G)* is minimized, the capacity of APs can be maximized.

Discussion-Capacity

$C = 60$ Mbps
$n = 5$

$$H(G) = 1.2 \;\Rightarrow n \times H(G) = 6$$

$$\Rightarrow \lambda = 10 \text{ Mbps}$$



$$H(G) = 2 \;\Rightarrow n \times H(G) = 10$$

$$\Rightarrow \lambda = 6 \text{ Mbps}$$

When *H(G)* is minimized, the capacity of APs can be maximized.

Conclusion

- Optimally placing gateways.
  - Minimize the average number of hops from APs to gateways.
  - Minimize the total power consumption.
  - Maximize the average capacity of each AP.

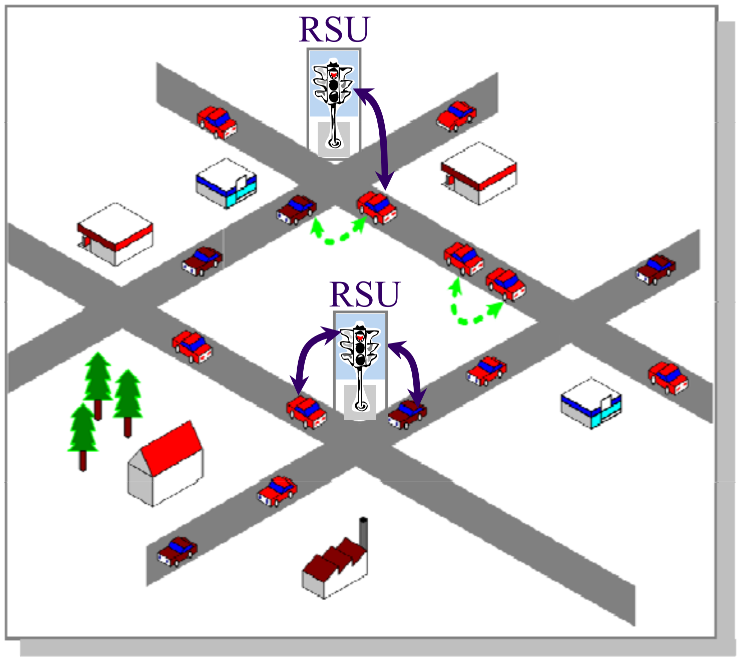
On Scheduling Vehicle-Roadside Data Access

Yang Zhang , Jing Zhao and Guohong Cao
Department of Computer Science & Engineering
The Pennsylvania State University

ACM VANET 2007

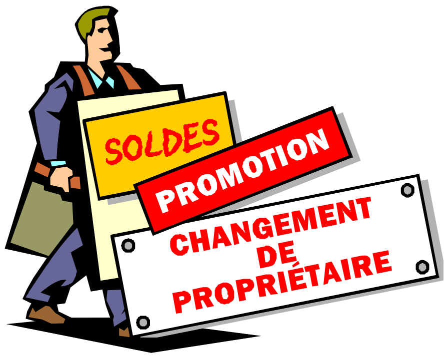
Introduction – The Architecture of Vehicle (Roadside service)

- **V**ehicular **A**d-hoc **NET**works includes
  - Moving Vehicles, and
  - Roadside Units (RSU).
    - Local broadcasting information
    - IEEE 802.11 access point



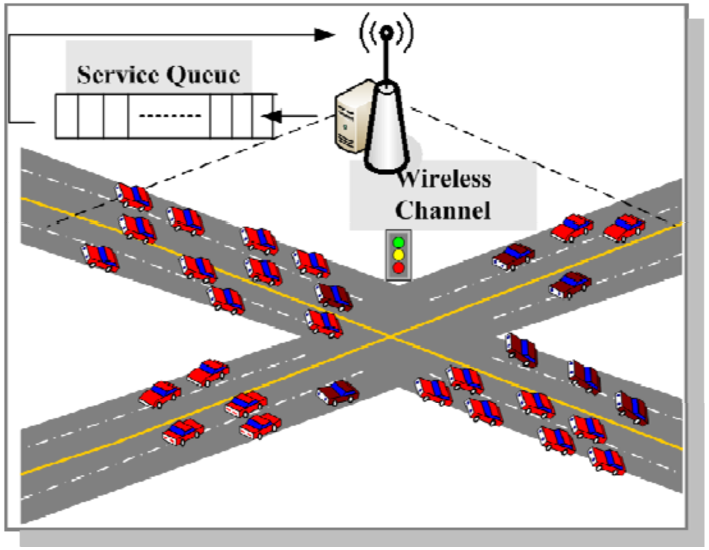
Introduction – The Architecture of Vehicle (Roadside service)

- Applications of the RSU
  - Commercial Advertisement
  - Real-Time Traffic
  - Digital Map Downloading



Introduction – Challenges

- Bandwidth Competition
  - All requests (upload/download) compete for the same limited bandwidth
- Time Constraint
  - Vehicles are moving and they only stay in the RSU area for a short period of time
- Data Upload / Download
  - The miss of upload leads to data staleness



Problem

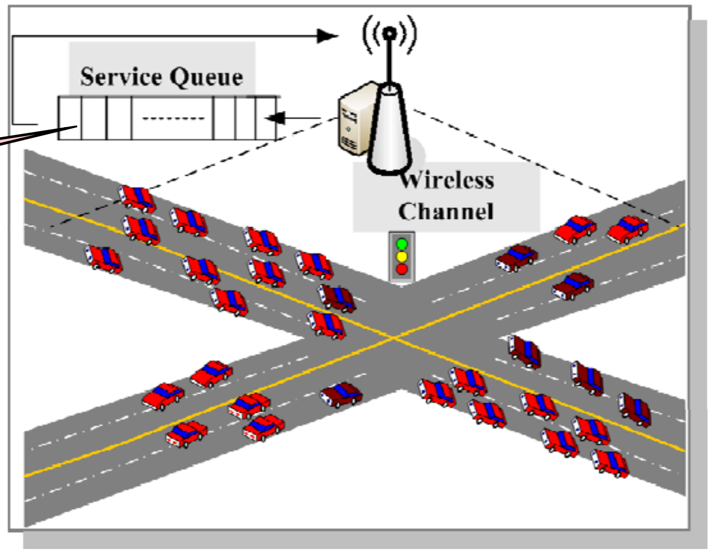
- How to develop a scheduling algorithm for data access?

- Performance Metrics (Tradeoff)
  - Service Ratio (下載資訊服務的成功率)
    - Ratio of the number of requests served before the service deadline to the total number of arriving requests.
  - Data Quality (車子上傳資訊更新資料)
    - Percentage of fresh data access

System Model

- Each request is characterized by
  - *v-id* : the Vehicle-ID
  - *d-id* : the Data-ID
  - *op* : upload / download
  - *deadline* : Time Constraint

(*v-id* , *d-id* , *op* , *deadline*)

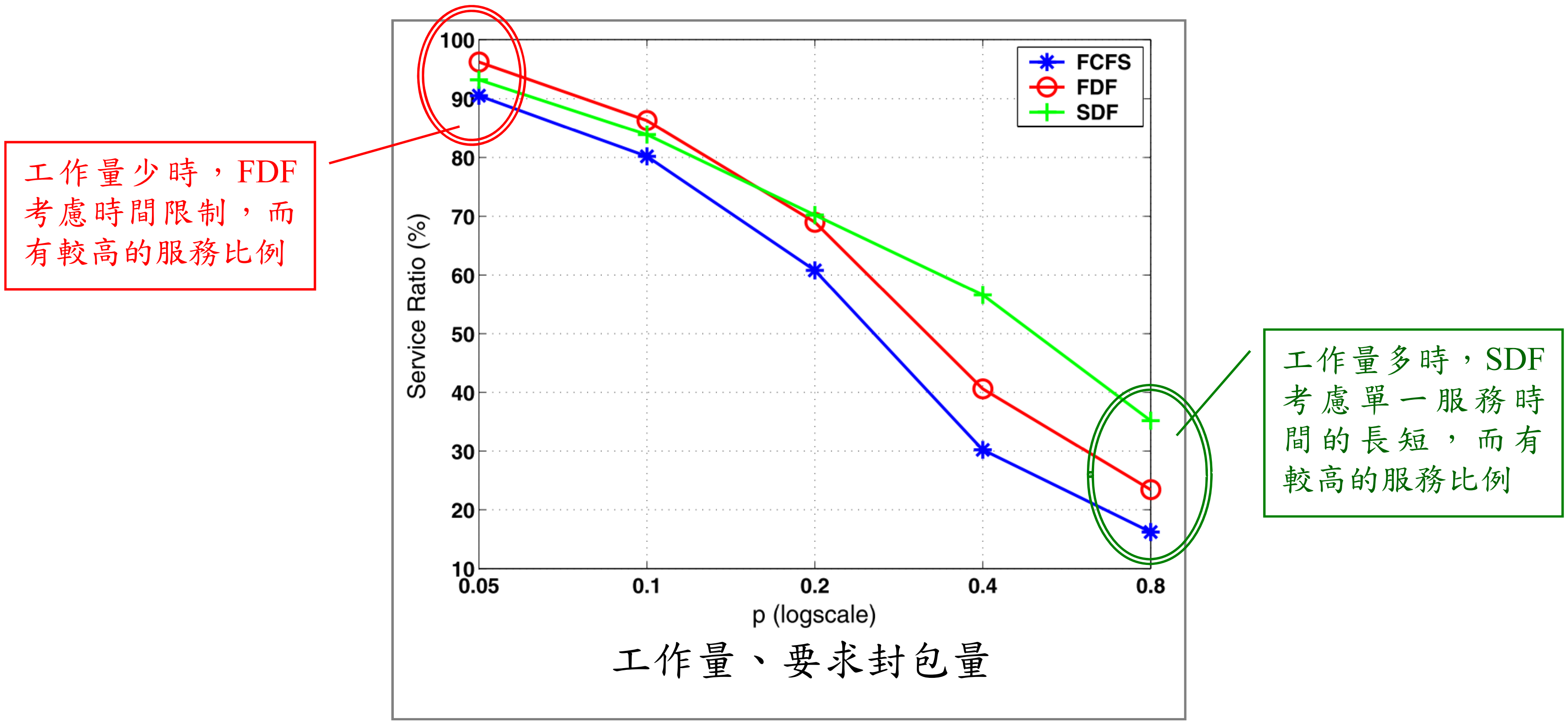


System Model

- Assumptions
  - Each vehicle knows the service deadline of its request.
    - Location-aware (GPS)
    - Deadline-aware (driving velocity)
  - The RSU maintains a service cycle
  - Service **non-preemptive**

Naive Scheduling Policies

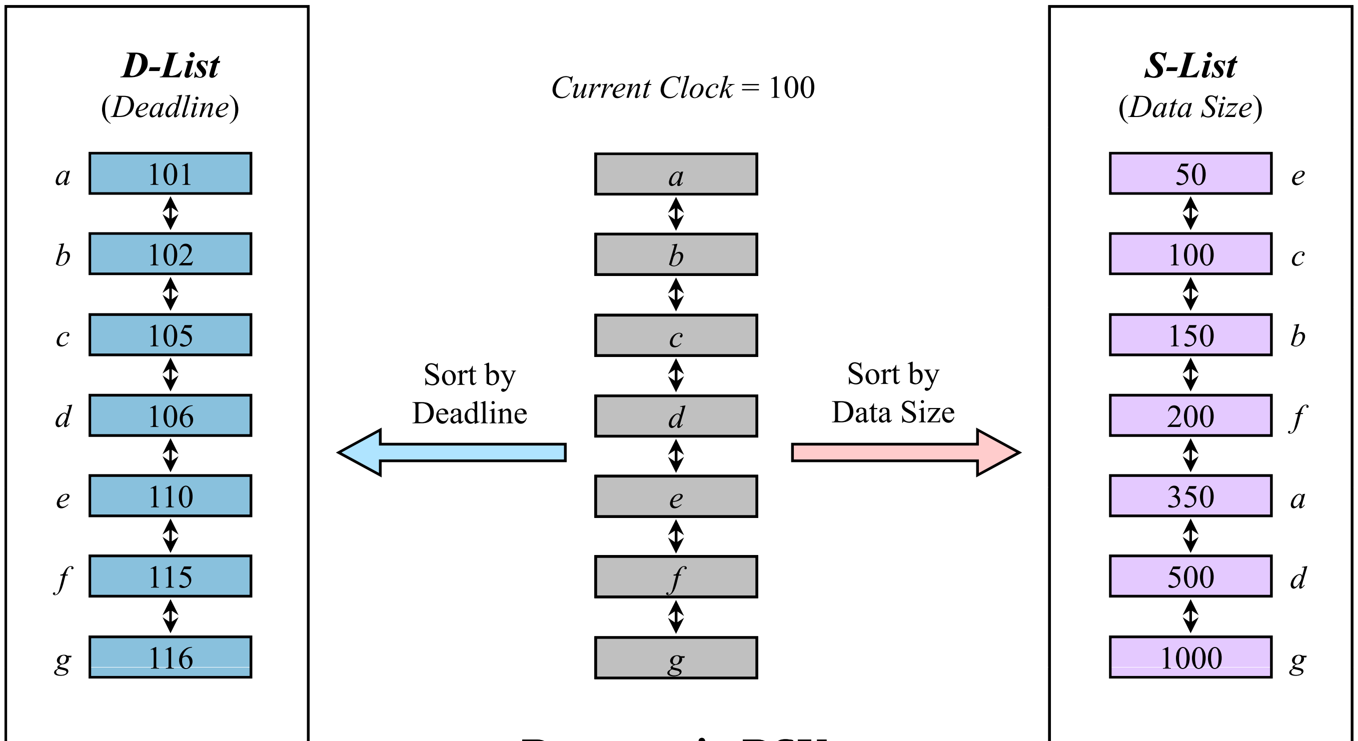
- First Come First Serve (**FCFS**)
- First Deadline First (**FDF**)
- Smallest Datasize First (**SDF**)



Scheduling Scheme I

– *D*S* Scheduling

- *D*S* Scheduling
  - D : D-List (Deadline-list)
  - S : S-List (DataSize-list)



Requests in RSU

Scheduling Scheme I

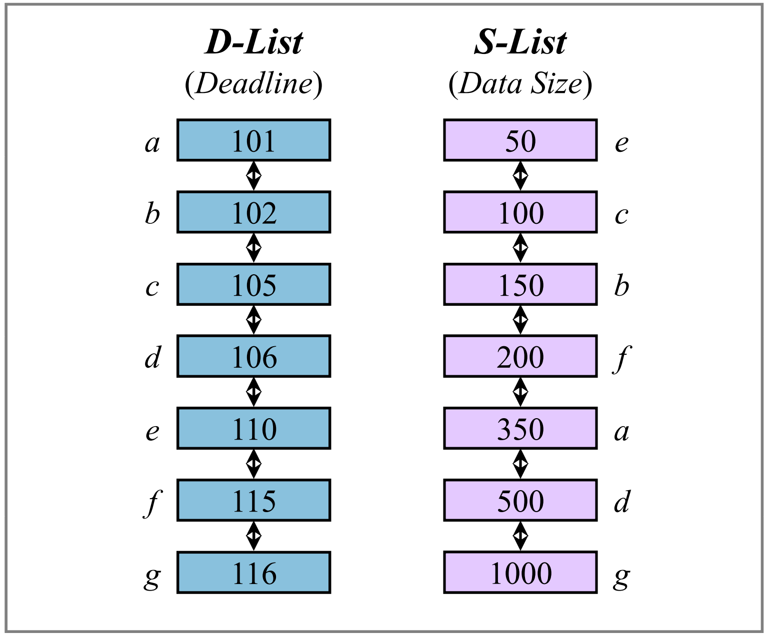
– *D*S* Scheduling

- Basic Idea
  - Given two requests with the same deadline, the one asking for a small size data should be served first.
  - Given two requests asking for the data items with same size, the one with an earlier deadline should be served first.
  - Assign each arrival request a *service value* based on its deadline and data size, called ***DS_value*** as its service priority weight

$$DS\_value = (Deadline - CurrentClock) * DataSize$$

Scheduling Scheme I
– *D*S* Scheduling Implementation

Information in RSU



Current Clock **= 100**

Scheduling Scheme I
– *D*S* Scheduling Implementation

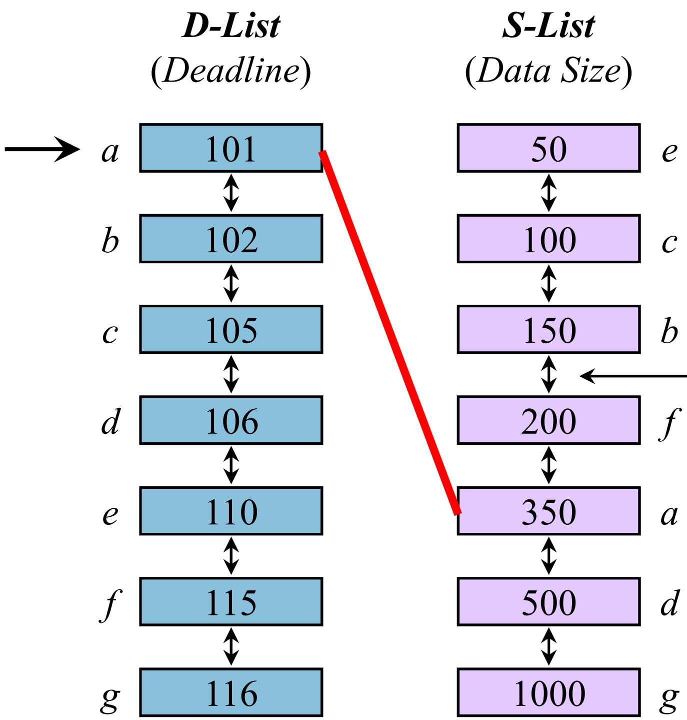
Step 1

MinDS
= (*Deadline* - *CurrentClock*) * *DataSize*
= (101-100) * 350 = 350

Current Clock = 100

目前最小的 *DS_Value* 為 350，
而 *D-List* 次要緊急的 entry 為 request *b*；
若 request *b* 要有更小的 *DS_Value*，
則與其配對的 *DataSize* 必須小於 *MinS*

也就是 ***MinS* = 350 / (102-100) = 175**



MinS = 175

Scheduling Scheme I

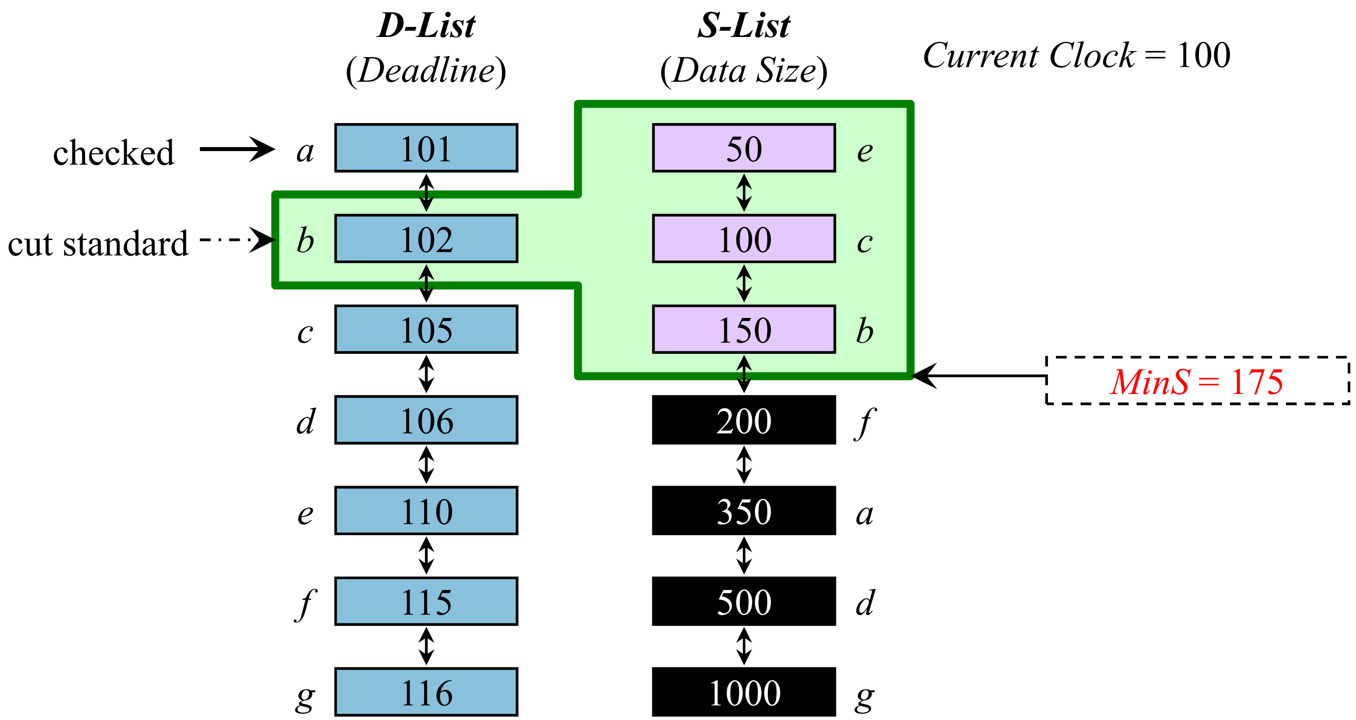
– *D*S* Scheduling Implementation

Step 1

MinDS

= (*Deadline* - *CurrentClock*) * *DataSize*

= (101-100) * 350 = 350



Scheduling Scheme I
– *D*S* Scheduling Implementation

Step 2

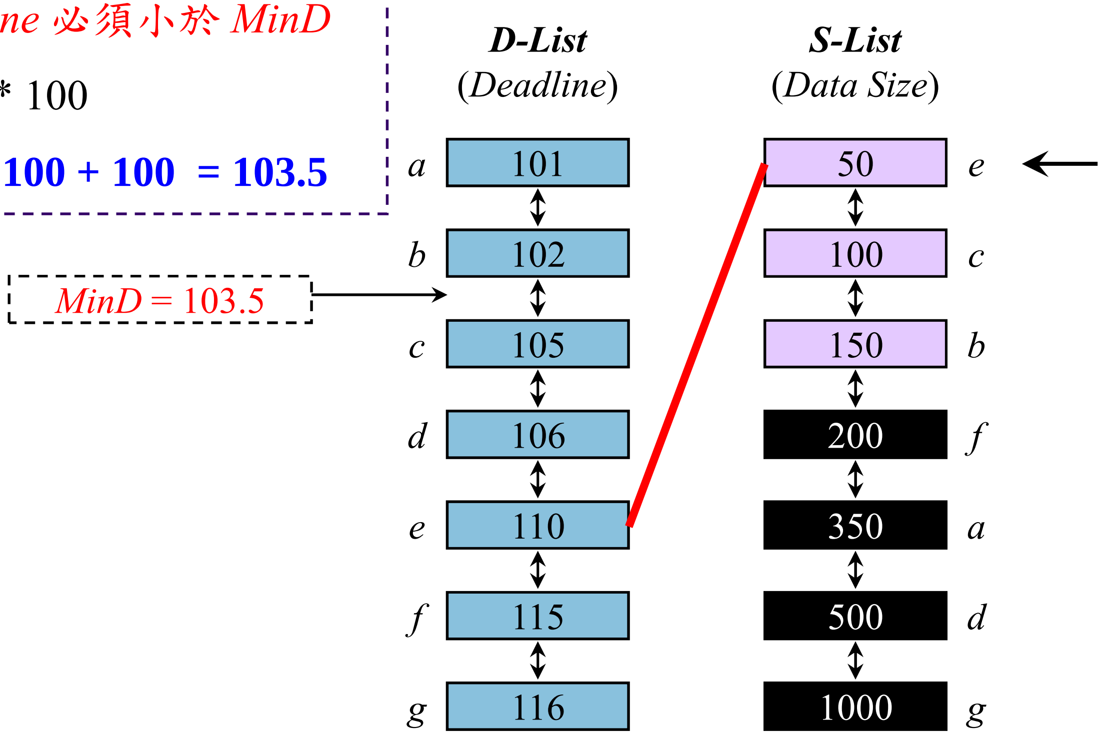
Current Clock = 100

目前最小的 *DS_Value* 依然為 350，
而 *S-List* 次大的 entry 為 request *c*；
若 request *c* 要有更小的 *DS_Value*，
則與其配對的 *Deadline* 必須小於 *MinD*

350 > (*MinD* – 100) * 100

也就是 ***MinD* = 350 / 100 + 100 = 103.5**

***DS_Value* (request *e*)**
= (*Deadline* - *CurrentClock*) * *DataSize*
= (110-100) * 50 = 500 > ***MinDS*** = 350

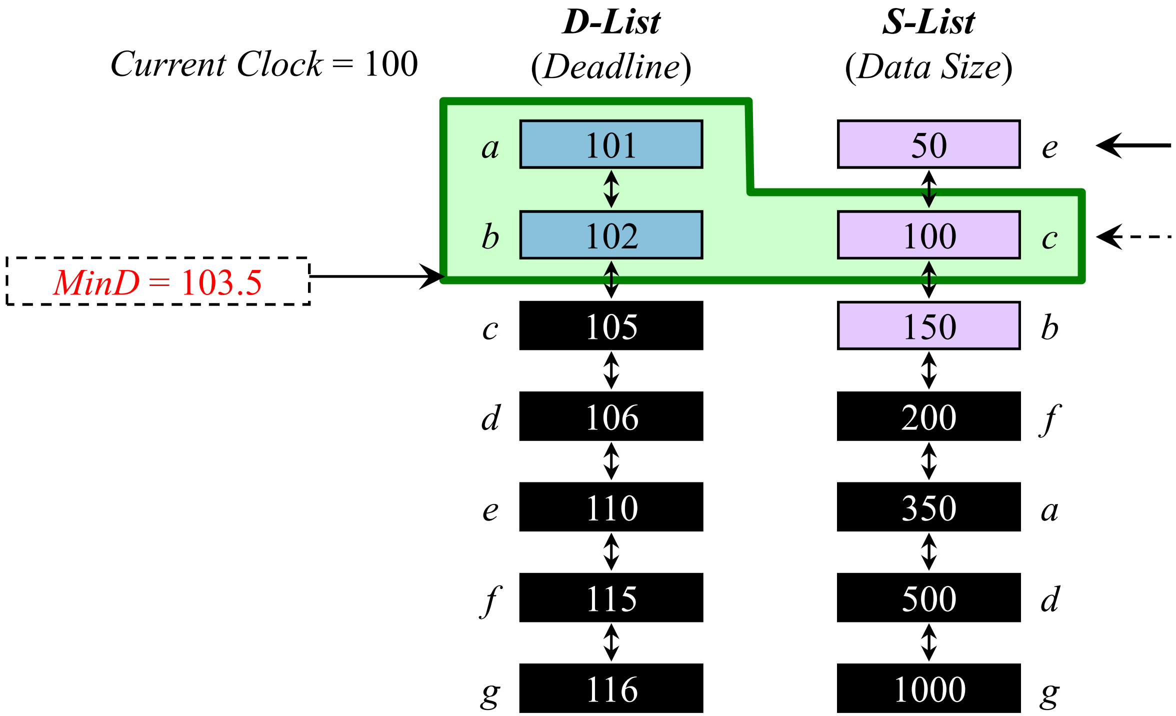


Scheduling Scheme I

– *D*S* Scheduling Implementation

Step 2

***DS_Value* (request *e*)**
= (*Deadline* - *CurrentClock*) * *DataSize*
= (110-100) * 50 = 500 > ***MinDS*** = 350



Scheduling Scheme II

– Download Optimization: Broadcasting

- Observation
  - several requests may ask for downloading the same data item
  - wireless communication is broadcast in nature
- Basic Idea
  - delay some requested data and broadcast it before the deadlines, then several requests may be served via a single broadcast
  - the data with **more** pending requests should be served first
  - *DSN_value = (Deadline − CurrentClock) * DataSize/Number*

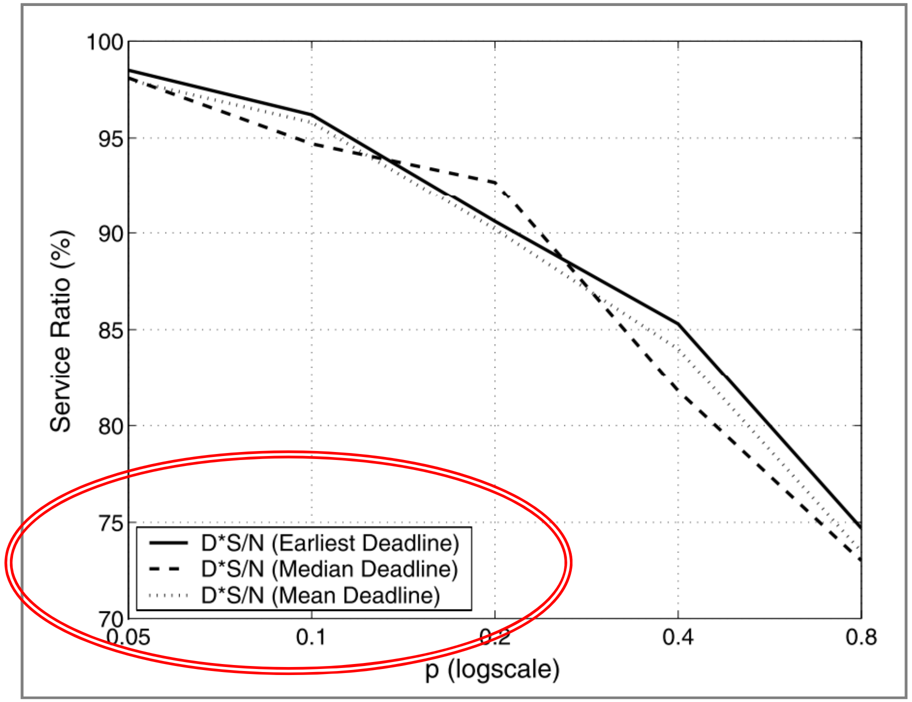
The number of pending requests for the same data (N)

Scheduling Scheme II

– Download Optimization: Broadcasting

- When calculating their *DSN value*, we need to assign each pending request group a single deadline to estimate the urgency of the whole group.

Not too much impact !



Scheduling Scheme II
– The Problem of DSN

- Data Quality will be blocked

  DSN_value = (Deadline − CurrentClock) * DataSize / Number

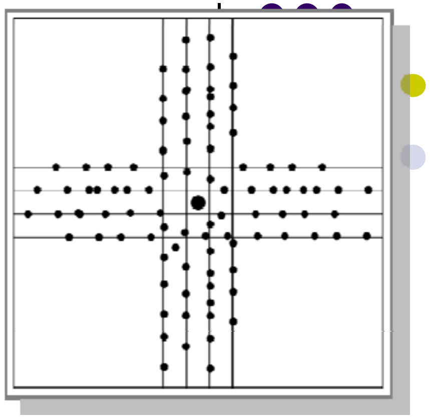
  - For upload request, it is not necessary to maintain several update requests for one data item since only the last update is useful.
  - Number value of update requests is always 1, which makes it not fair for update requests to compete for the bandwidth .
  - D*S/N can improve the system service ratio but block the service opportunity of update requests, which degrades the data quality for downloading.

Scheduling Scheme III

– Upload Optimization (2Step Scheduling)

- Basic Idea
  - two priority queues: one for the update requests and the other for the download requests.
  - the data server provides two queues with different bandwidth (i.e., service probability)

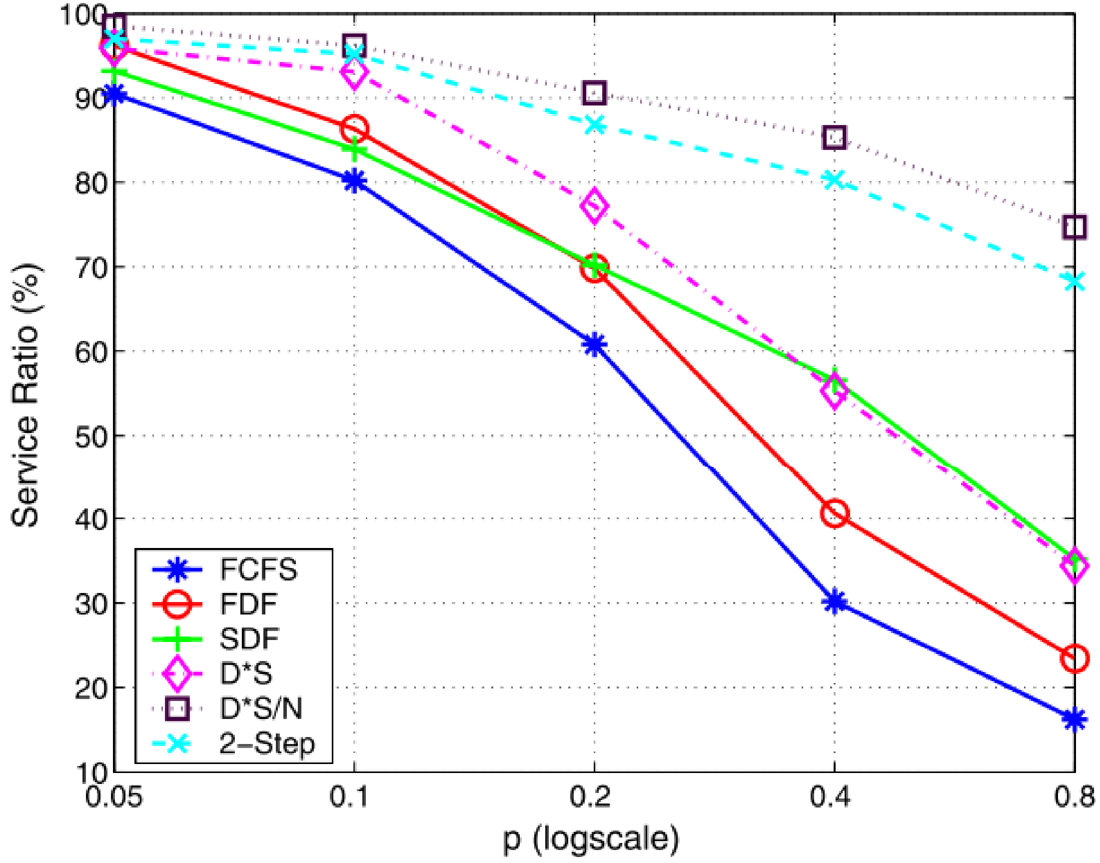
Simulation Setup



- NS-2
- 400m*400m square street scenario
- One RSU server is located at the center of two 2-way roads
- 40 vehicles randomly deployed on each lane
- Each vehicle issues request with a probability *p*
- Access pattern of each data item follows *Zipf* distribution

Parameter	**Value**
Simulation Time	900s
Transmission Rate	$5Mbit/s = 625Kbyte/s$[11]
Vehicle Velocity	15m/s
Wireless Coverage	200 m
Data size	50K $\sim$ 5M, average 2.5M
Vehicle-Vehicle Space	20m
Data set size	25
$Zipf$ Parameter θ	0.8
Update Percentage	10%
Adaptation Window	40s

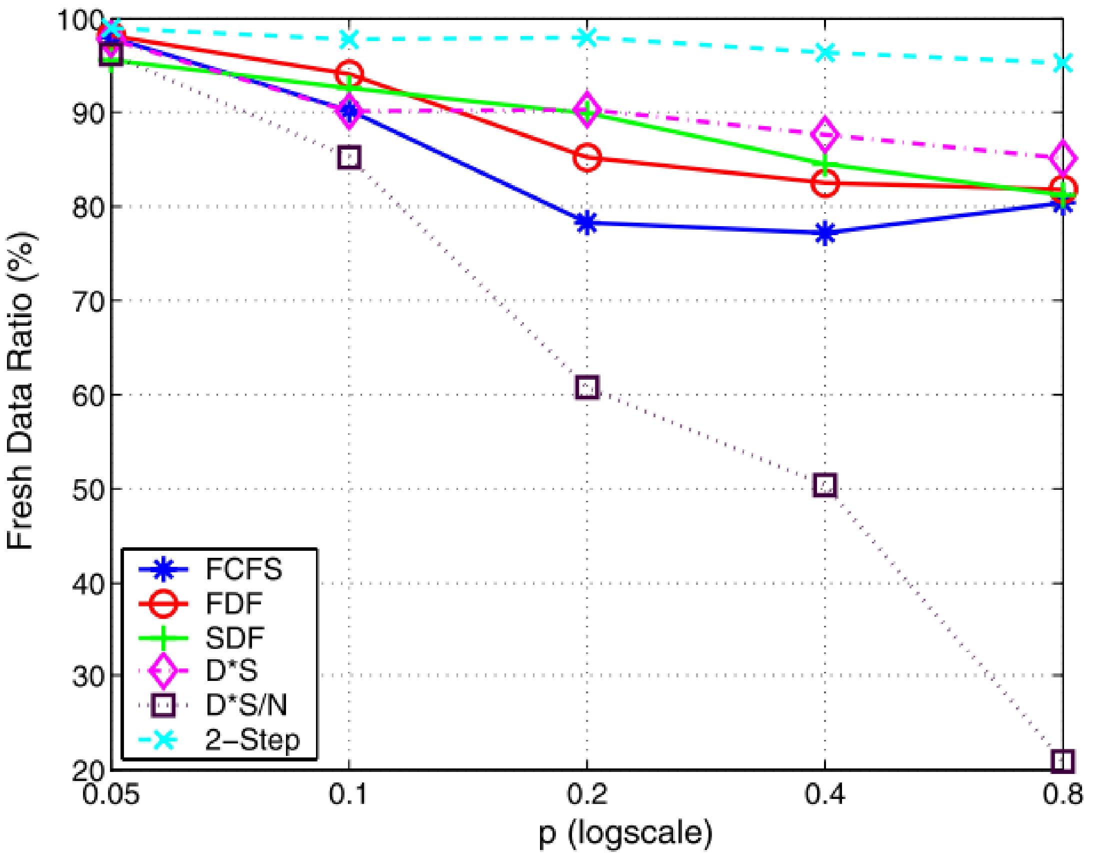
Effect of Workload



(a) Service Ratio

As workload increases, *D*S/N* can achieve the highest service ratio while its data quality degrades dramatically

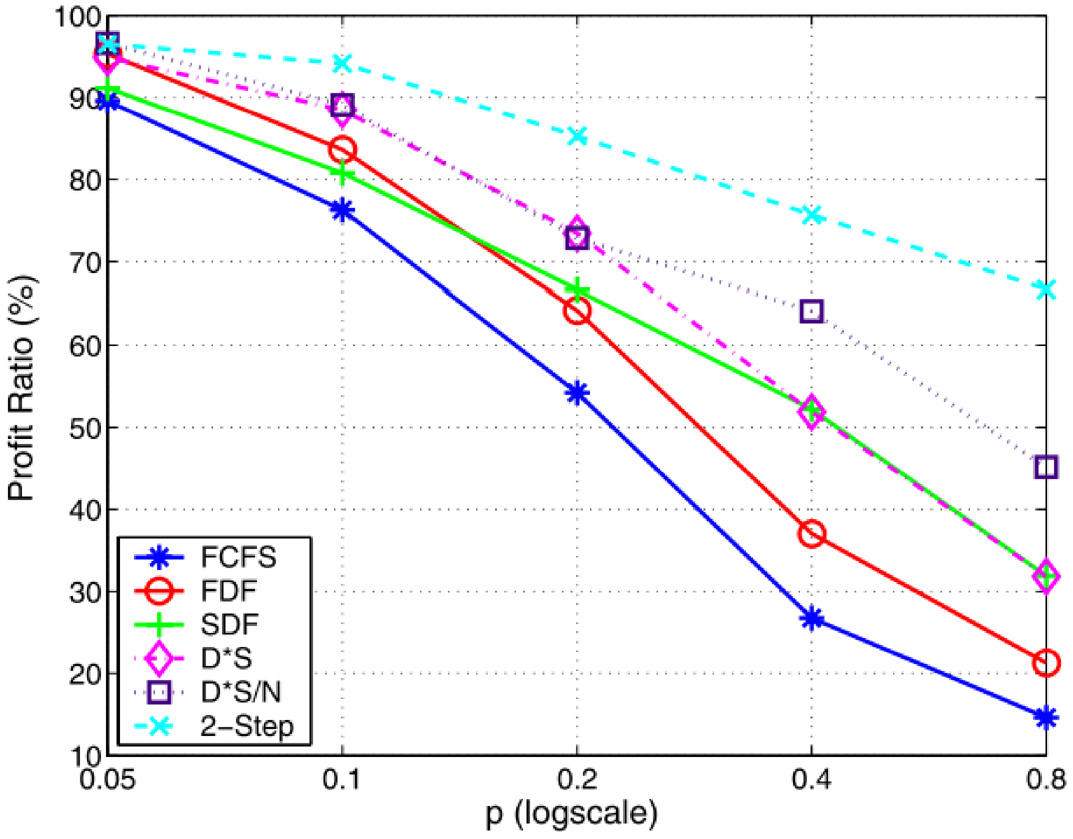
Effect of Workload



(b) Fresh Data Ratio

As workload increases, *D*S/N* can achieve the highest service ratio while its data quality degrades dramatically

Effect of Workload



(c) System Profit ($\alpha = 0.5$)

As workload increases, *D*S/N* can achieve the highest service ratio while its data quality degrades dramatically

Conclusion

- This paper proposed a basic scheduling scheme called *D*S* to consider both service deadline and data size when making scheduling decisions.
- To make use of the wireless broadcasting, this paper proposed a new scheduling scheme called *D*S/N* to serve multiple requests with a single broadcast.
- This paper also identified the effects of upload requests on data quality, and proposed a *Two-Step* scheduling scheme to provide a balance between serving download and update requests.