School of Information Systems

LARC Living Analytics Research Centre

SMU Singapore Management University

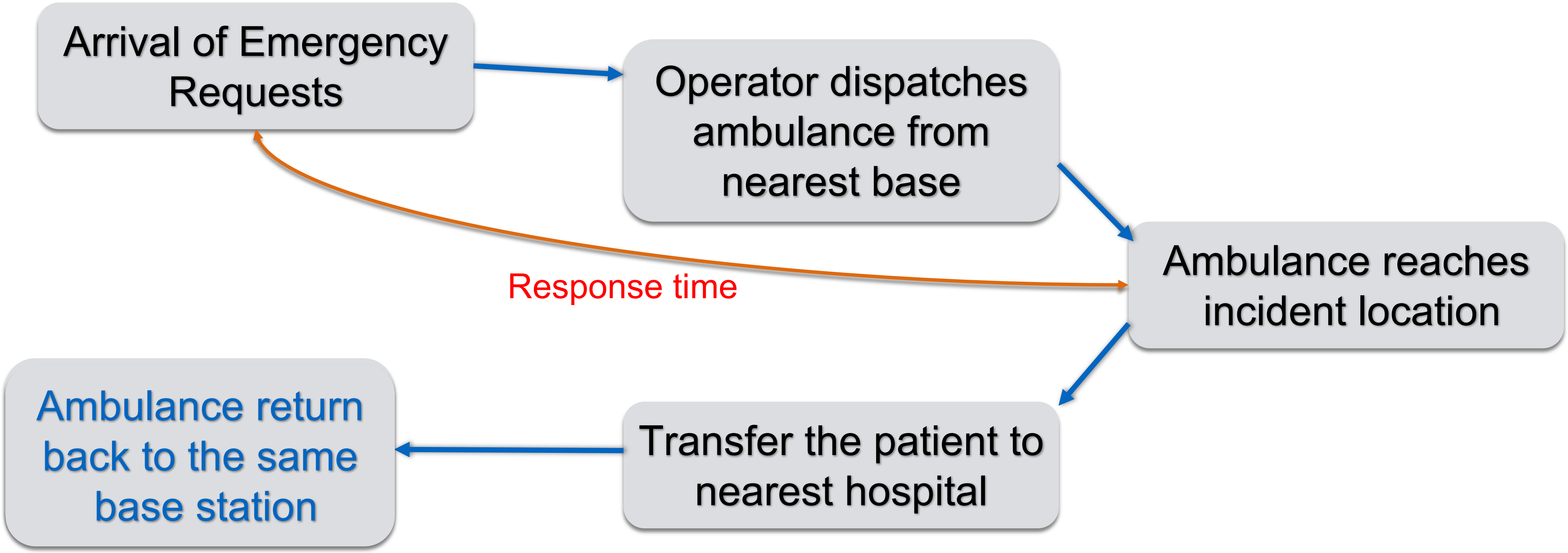
Strategic Planning for Setting up Base Stations in Emergency Medical Systems

Supriyo Ghosh and Pradeep Varakantham

School of Information Systems
Singapore Management University

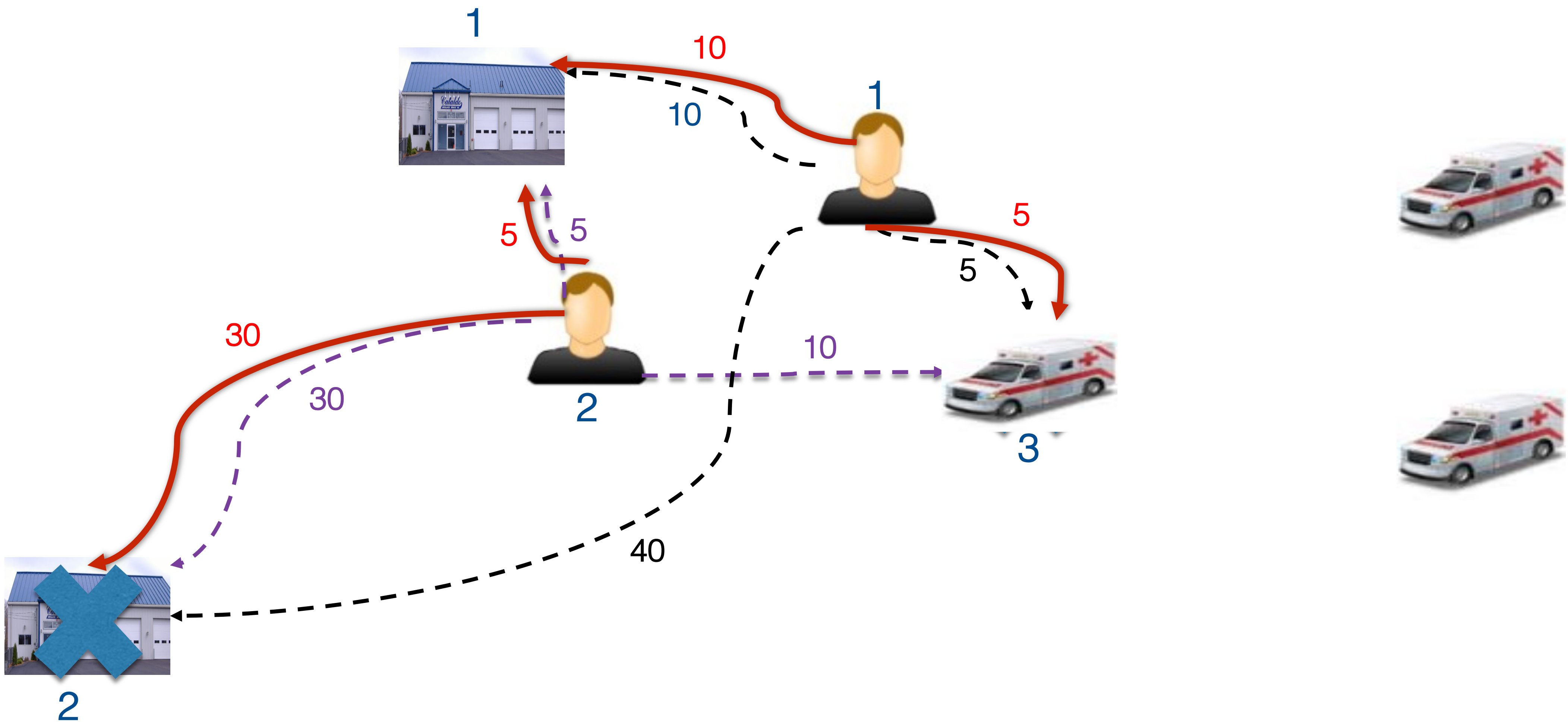
Motivation: Emergency Medical Systems

- Emergency Medical Systems:
  - Integral part of public health-care
  - Response time is the key factor
  - Placement of resources have major impact



Motivating Example

- Response times with base 1 & 2 are 10 and 30 minutes.
- Response times with base 1 & 3 are 5 and 5 minutes
  - Total response time reduces by 30 minutes
  - Both requests are served within 5 minutes



Challenges & Objectives

- Strategic planning in EMS is computationally challenging
  - Demand is dynamic & stochastic
  - Exponentially large action space
  - Direct impact on ambulance allocation problem
  - Budget for resources (#bases & #ambulances) is dynamic
  - Extension of k-center facility location problem (NP-Hard Problem)

- Goal: Strategic planning to optimize EMS performance metrics.
  - **Bounded time response:** Maximise the number of requests that are served within a given threshold time (e.g., 15 minutes)
  - **Bounded risk response:** Minimise the response time for a fixed percentage (e.g., 80%) of requests

Background & Contribution

- Operational Planning:
  - Ambulance allocation and dynamic redeployment
    - [Yue *et. al.,* 2012; Siasubramanian *et. al.,* 2015; Maxwell *et. al.,* 2010]
    - Presume a fixed set of bases are given
- Strategic planning for rare large-scale disaster response
  - [Sylvester *et. al*., 1857; Huang *et. al*., 2010]
  - Not efficient for day-to-day decision making in EMS

- Our contributions:
  - A data-driven greedy algorithm – add bases incrementally
  - Use faster lazy greedy to optimize widely used metrics in EMS
  - Evaluate our approach on a simulation build on real-world data sets

Solution Overview

Inputs: Possible bases $\mathcal{B}$, fleet of ambulances, request logs for a certain period
Initialization: Set resulting base *E* as null

Find the gain $\mu_s, \; \forall s \in \mathcal{B}$ for adding base s to *E*, by optimally allocating ambulances to $E \cup \{s\}$

Choose the base s^* with highest marginal gain μ_{s^*}

Add s^* to *E* and remove it from $\mathcal{B}$, if budget is available

Ambulance Allocation Problem

- Input: Ambulance allocation problem are defined using tuple:

$$< \mathcal{R}, \mathcal{B}, \mathcal{A}, \mathbf{T}, L >$$

Each request $r \in \mathcal{R}$ is tagged with tuple $< t, s, h >$

$$L_{rl} = \begin{cases} 1 & \text{if } T_{l,r.s} \leq 15 \text{ minutes} \\ 0 & \text{Otherwise} \end{cases}$$

Bounded time response objective

- Output: Number of ambulances, a_l allocated to each bases $l \in \mathcal{B}$
- Objective: Maximize number of requests served within 15 minutes.
- Decision variables:

$$x_{rl} = \begin{cases} 1 & \text{if request } r \text{ is served from base } l \\ 0 & \text{Otherwise} \end{cases}$$

MILP for Optimizing Bounded Response Time

$$\max_{a,x} \sum_{r \in \mathcal{R}} \sum_{l \in \mathcal{B}_r} x_{rl} L_{rl}$$

Maximize bounded response time

$$s.t. \sum_{l \in \{\mathcal{B}_r \cup \perp\}} x_{rl} = 1, \qquad \forall r \in \mathcal{R}$$

Serve request from one base only

$$x_{rl} + \sum_{j \in P_r^l} x_{jl} \leq a_l, \qquad \forall r \in \mathcal{R}, l \in \mathcal{B}_r$$

Assigned ambulances at any point is bounded by allocated ambulance

$$\sum_{l \in \mathcal{B}} a_l = |\mathcal{A}|$$

$$a_l \geq 0, x_{rl} \in \{0, 1\}$$

Set of parent requests for ***r***

Ensures all the ambulances are allocated

+ Similarly an MILP is used to optimize bounded risk response objective

Submodularity

Objective function $F : 2^{\mathcal{B}} \rightarrow \mathbb{R}$ is submodular if

$$\Delta(A|b) - \Delta(B|b) \geq 0 \qquad \forall b \in \mathcal{B} \setminus B$$

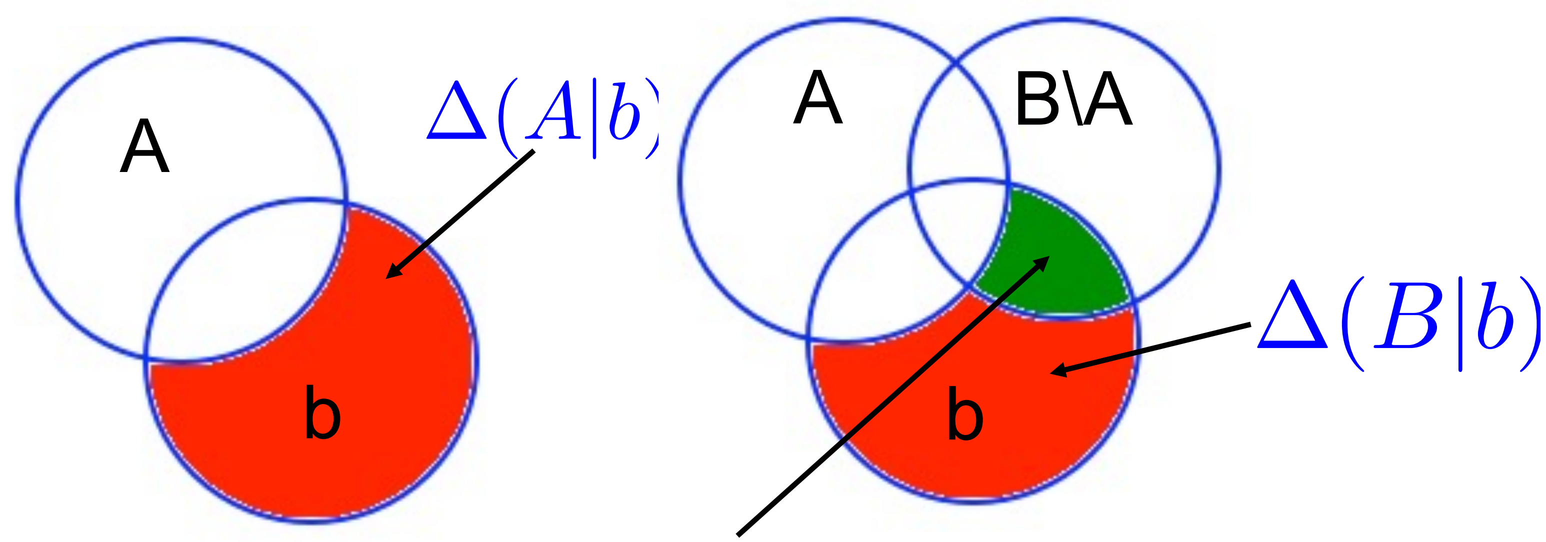
where, $A \subset B \subseteq \mathcal{B}$ and $\Delta(A|b) = F(A \cup \{b\}) - F(A)$

Proposition 1: *F* function is monotone submodular for bounded time response objective. Therefore, greedy approach provides $(1 - \frac{1}{e})$ approximation guarantee

Proof: Let $S_i \in \mathcal{R}$ is set of requests served from base *i*

$$F(A) = |\cup_{i \in A} S_i|$$

F Supports all properties of sets



$\boxed{\Delta(A|b) - \Delta(B|b) \geq 0}$ *(Proved)*

School of Information Systems

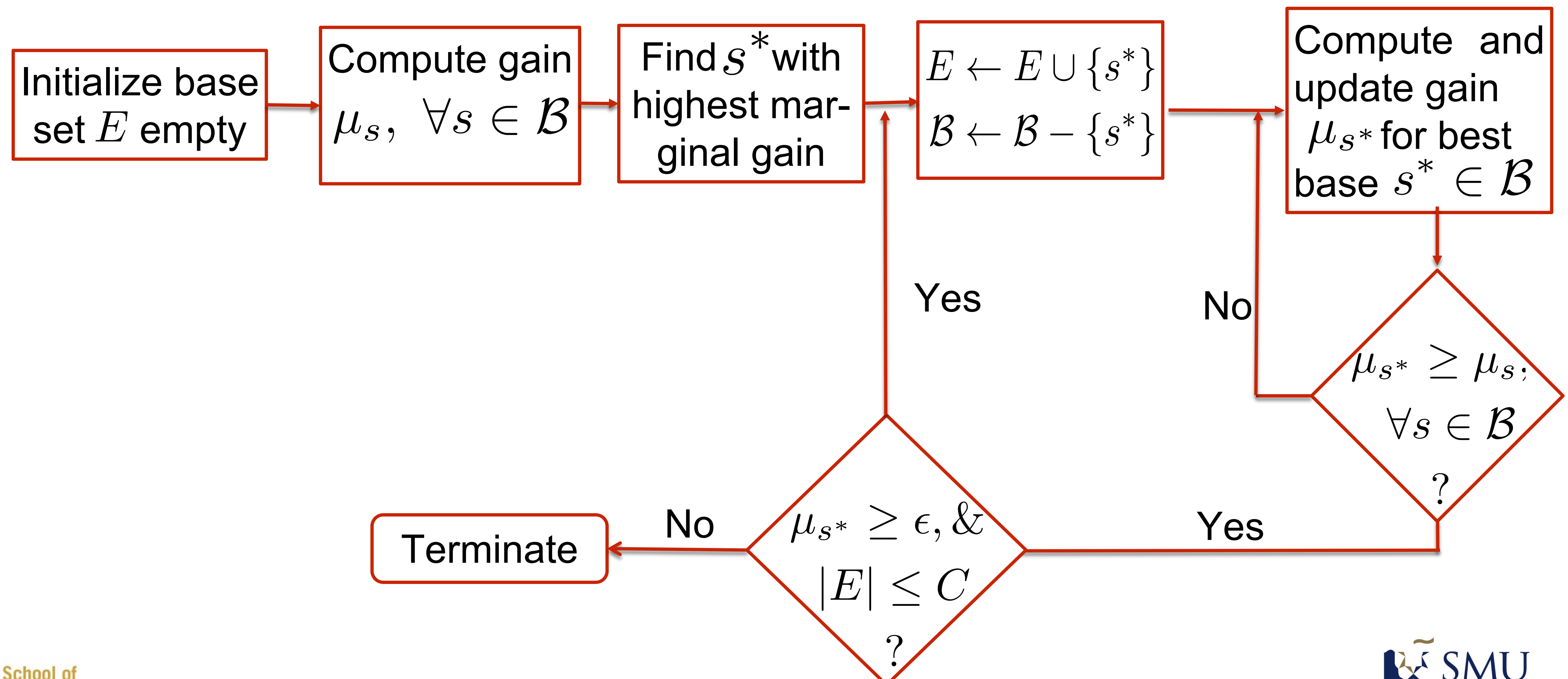
SMU SINGAPORE MANAGEMENT UNIVERSITY

Lazy Greedy Algorithm

Proposition 2: **For a placement of bases $E \in \mathcal{B}$ and for each available base $s \in \mathcal{B} \setminus E$, let $\Delta_s = F(E \cup s) - F(E)$ then:**

$$\max_{\mathcal{B},\mathcal{A},\mathcal{R}} F(\mathcal{B}) \leq F(E) + \sum_{s \in \{\mathcal{B} \setminus E\}} \Delta_s$$

+ Lazy Greedy Approach



School of Information Systems

SMU SINGAPORE MANAGEMENT UNIVERSITY

Experimental Setup

- Data set: Real EMS data set from a large Asian city
  - 58 bases, 58 ambulances
  - 1500 weeks of request samples - divided into training, validation and test set
- Benchmark Approaches
  - Baseline – one ambulance in each base
  - Bounded Time Response Optimization [BTRO] (Yue *el. al.*, 2012)
  - Risk Based Optimization [RBO] (Saisubramanian *et. al.*, 2015*)*
- Event-driven Simulation (Yue *et. al.*, 2012):

Create event list ξ with requests $\mathcal{R}$ sorted in arrival order

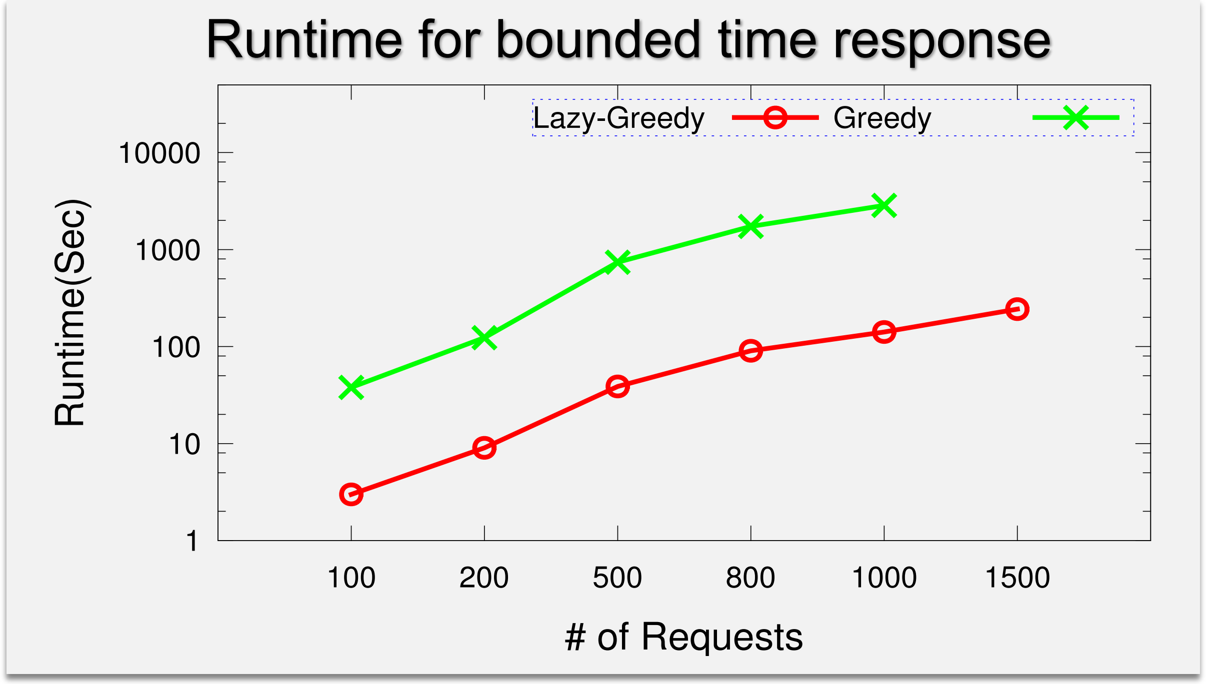
→ Pop event $e \in \xi$ iteratively until list is empty

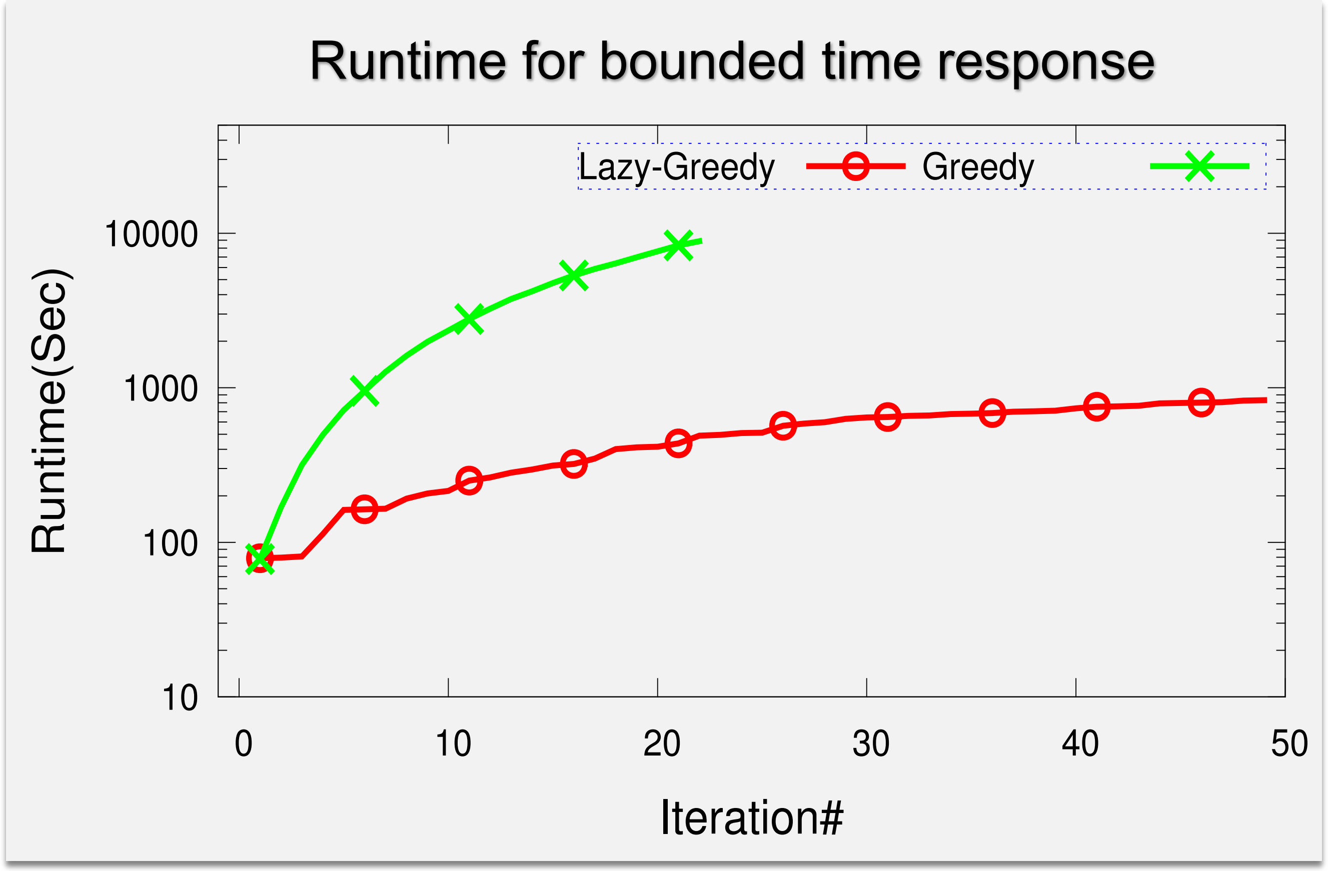
→ If *e* is request, then dispatch nearest ambulance and add a job-completion event to ξ

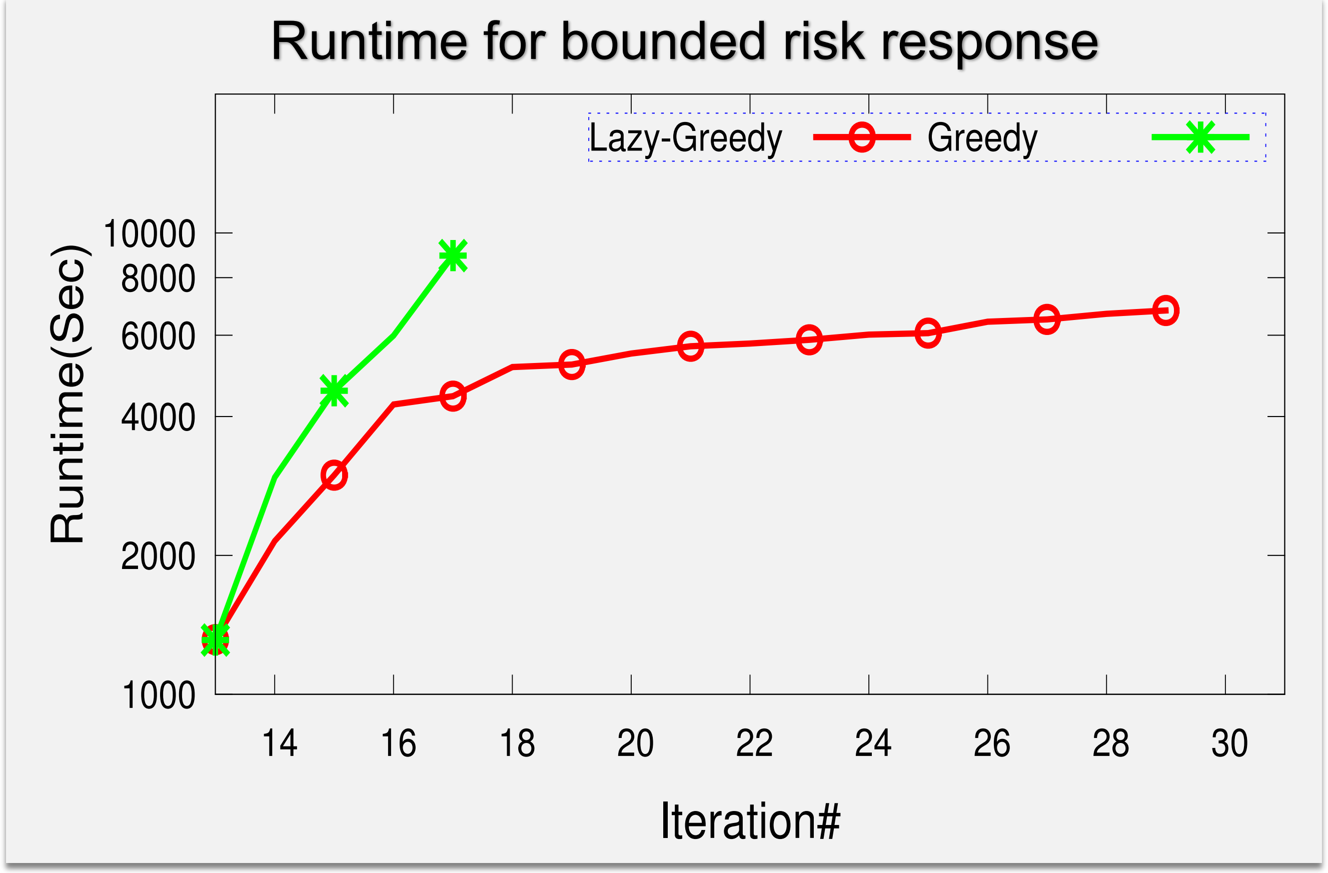
→ Otherwise, for job-completion event, add the ambulance to available set of fleet

Runtime Gain for Lazy-Greedy

- Lazy greedy
  - Scales gracefully with #requests for bounded time response.
  - Solves real problems within 10 minutes.
  - Efficient for bounded risk response also.



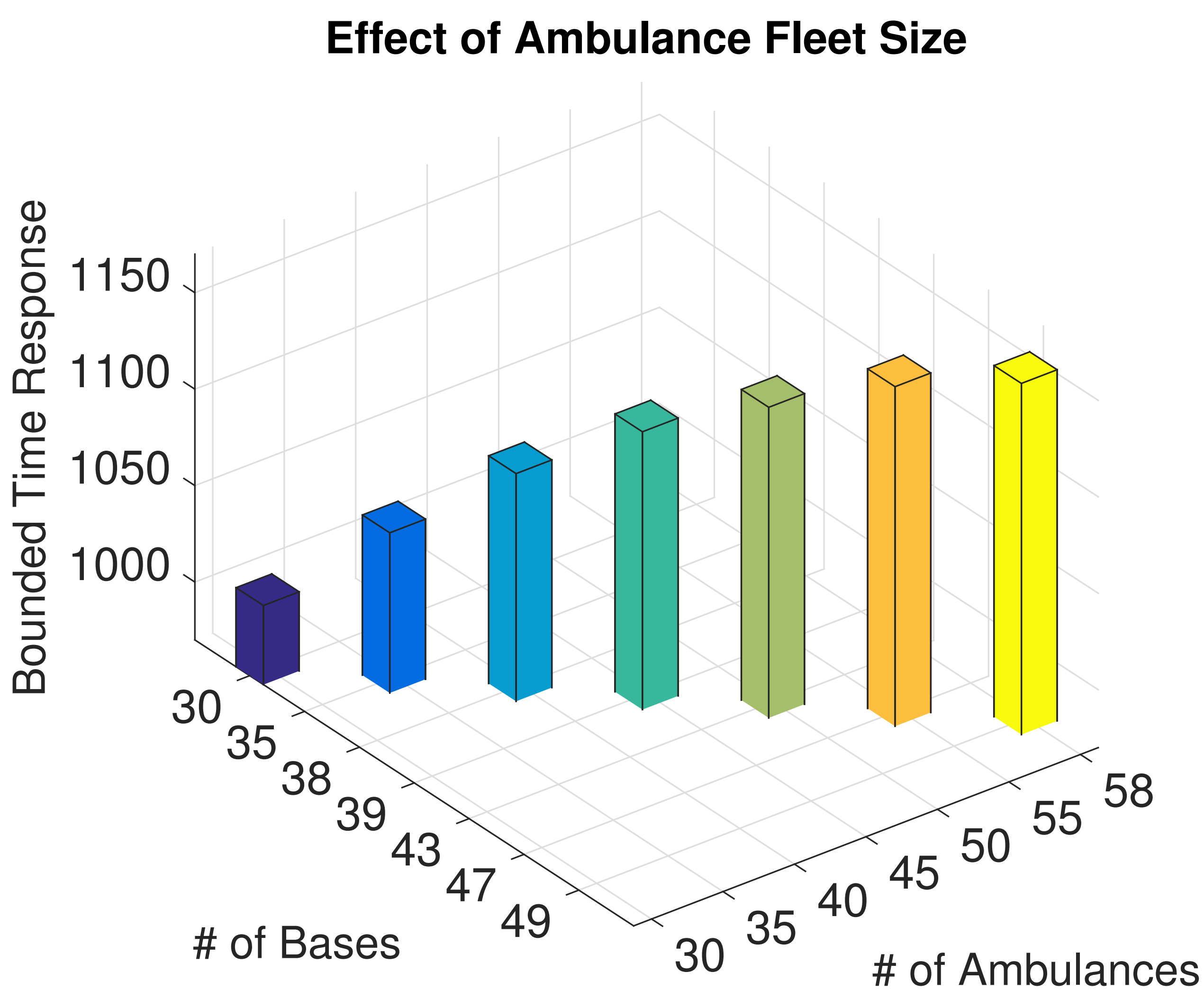


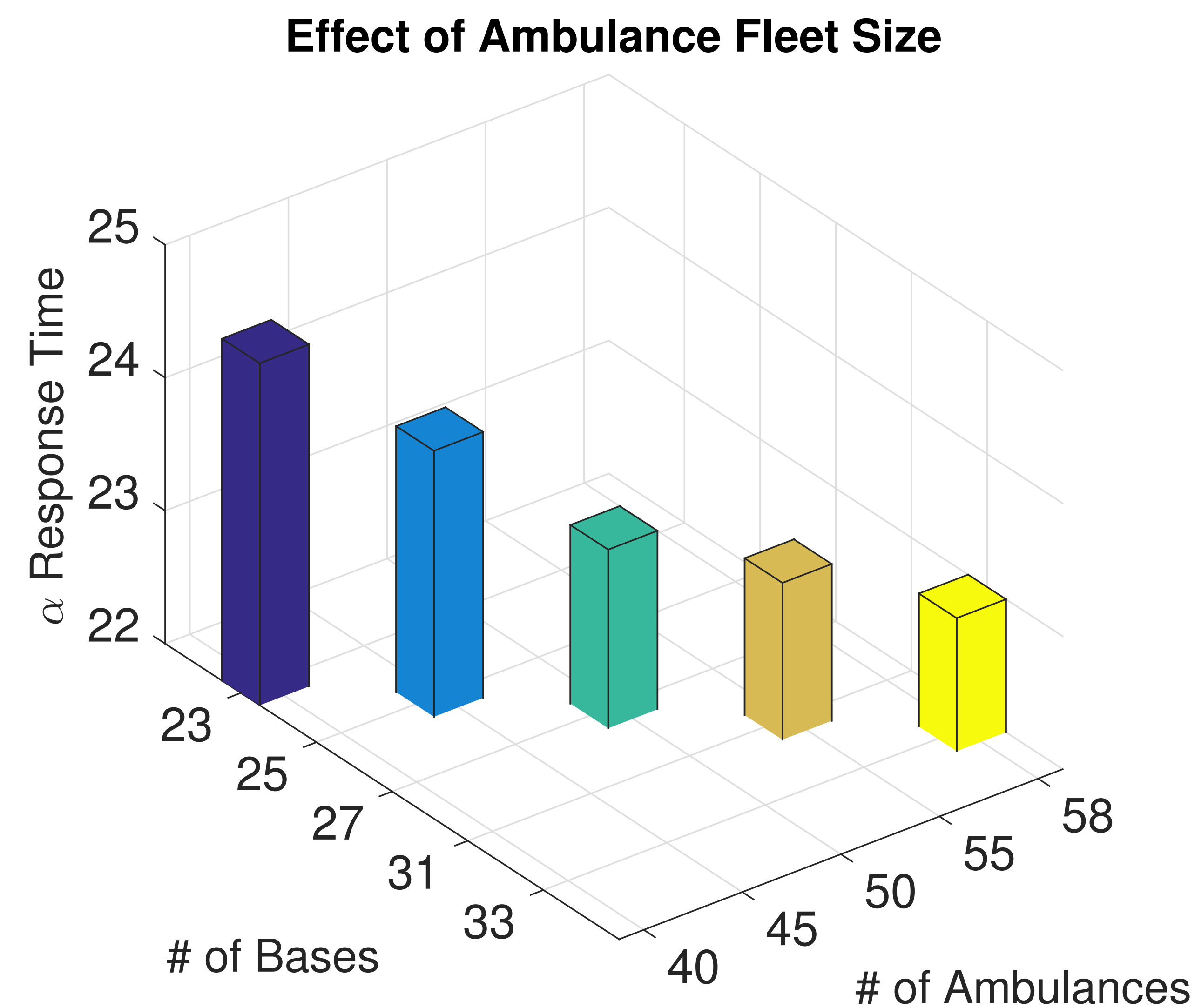


School of Information Systems

Effect of Ambulance Fleet Size

+ Increasing ambulance fleet size:
  + Bounded time response increases monotonically
  + Bounded risk response decreases monotonically
  + Number of required bases increases to accommodate extra ambulances



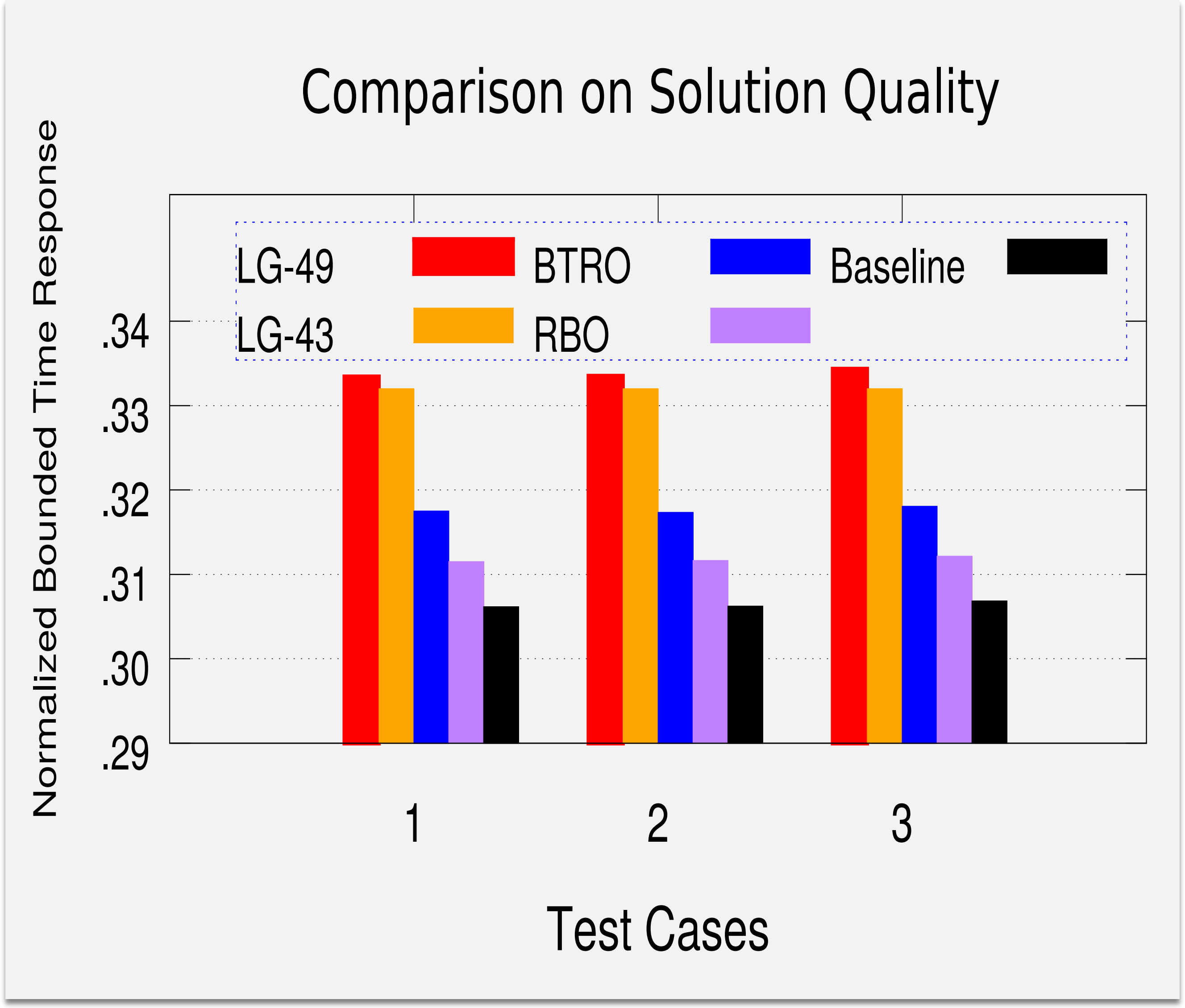


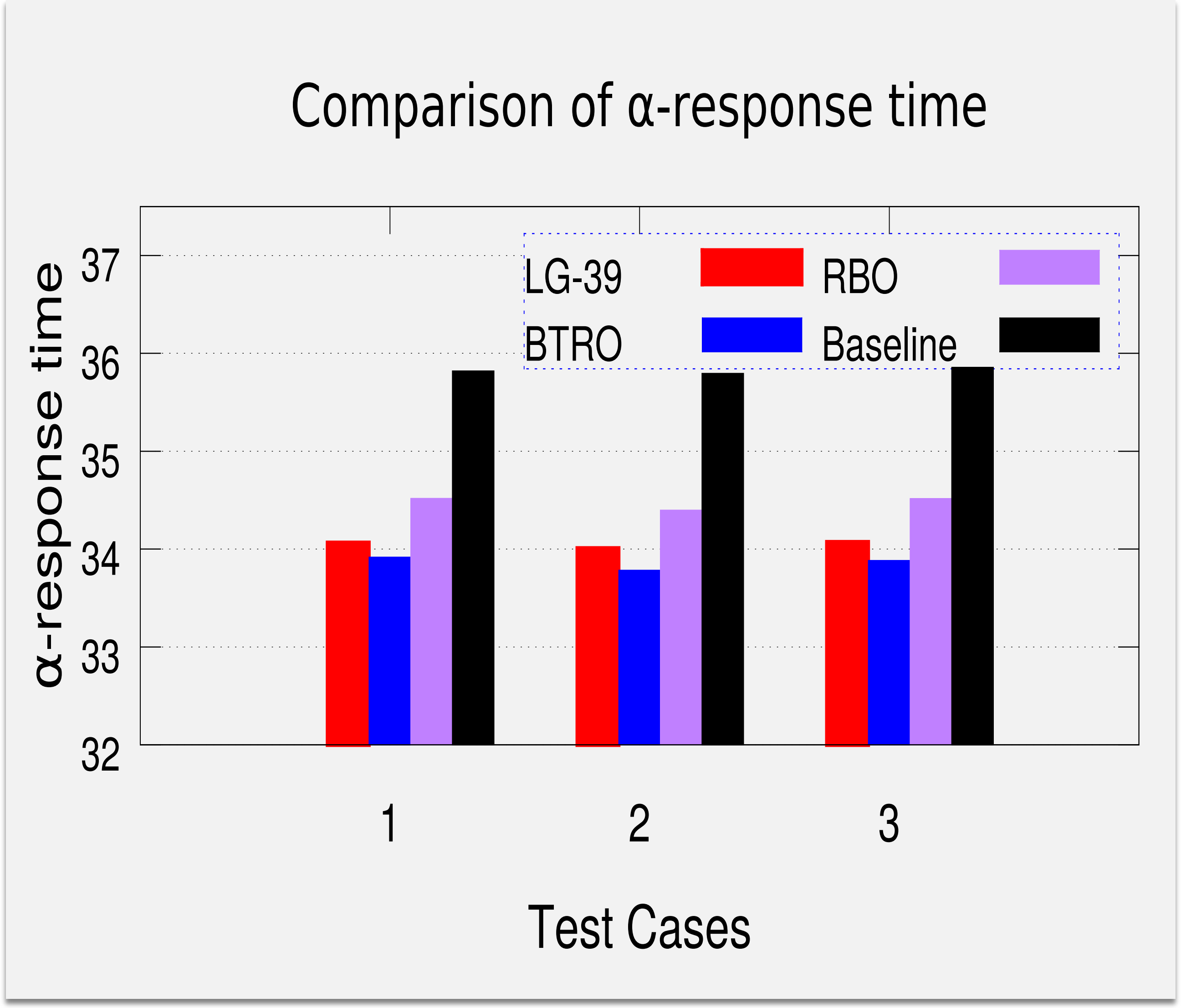
School of Information Systems

SMU Singapore Management University

Experimental Validation on Test Data Sets

- Our approach serves at least 3% extra requests within 15 minutes.
- Highly competitive with other approaches for bounded risk response by utilising less than 70% of the bases.





Conclusion

+ Strategic planning for EMS
  + Important large-scale problem for public health-care
  + Computationally challenging
  + We employ lazy greedy approach to add bases incrementally until marginal gain is significant
  + Our approach significantly improves the service level of EMS over existing benchmarks, on real-world data sets

Q & A

supriyog.2013@phdis.smu.edu.sg

MILP for Optimizing Bounded Risk Response

Variables:
δ^r : Response time for request r
z^r : Set to 1 if request r is served within δ

$$\max_{a,x} M - \delta$$

Minimise α-response time

s.t.

$$\frac{\delta^r - \delta}{M} \leq z^r, \quad \forall r \in \mathcal{R}$$

$$\frac{\sum_{r\in\mathcal{R}} z^r}{|\mathcal{R}|} \leq \alpha$$

Ensure that α% of requests are served within δ

$$\sum_{l\in\{\mathcal{B}_r \cup \perp\}} x_{rl} = 1, \quad \forall r \in \mathcal{R}$$

$$x_{rl} + \sum_{j\in P_r^l} x_{jl} \leq a_l, \quad \forall r \in \mathcal{R}, l \in \mathcal{B}_r$$

$$\sum_{l\in\mathcal{B}} a_l = |\mathcal{A}|$$

All the ambulances are allocated and each request is served by only one available ambulance

$$\delta^r \geq \sum_{l\in\mathcal{B}_r} x_{rl} \cdot T_{l,r.s} + x_{r\perp} \cdot \hat{M}, \quad \forall r \in \mathcal{R}$$

$$a_l \geq 0, x_{rl} \in \{0,1\}, z^r \in \{0,1\}, \delta, \delta^r \geq 0$$

Compute response time, large penalty is not assisted

Lazy Greedy Algorithm

Initialize: $E \leftarrow \{\perp\}, it \leftarrow 0$;

$\mu_s, \boldsymbol{A} \leftarrow \text{FindAllocation}(\mathcal{R}, E \cup \{s\}, \mathcal{A}), \quad \forall s \in \mathcal{B}$;

$g^0 \leftarrow \max_{s \in \mathcal{B}} \mu_s$;

$s^* \leftarrow \operatorname{argmax}_{s \in \mathcal{B}} \mu_s$;

$E \leftarrow E \cup \{s^*\}$;

$\mathcal{B} \leftarrow \mathcal{B} - \{s^*\}$;

repeat

  $it \leftarrow it + 1$;

  repeat

    $s^* \leftarrow \operatorname{argmax}_{s \in \mathcal{B}} \mu_s$;

    $g^{it}, \boldsymbol{A} \leftarrow \text{FindAllocation}(\mathcal{R}, E \cup \{s^*\}, \mathcal{A})$;

    $\mu_{s^*} \leftarrow g^{it} - g^{it-1}$;

    if $\{\mu_{s^*} \geq \mu_s, \forall s \in \mathcal{B}\}$ **then**

      $E \leftarrow E \cup \{s^*\}$;

      $\mathcal{B} \leftarrow \mathcal{B} - \{s^*\}$;

      Break;

  until ***True***;

until $(\max_{s \in \mathcal{B}} \mu_s \leq \epsilon)$;

return $E, \boldsymbol{A}$

Initialize empty base set

In 1st iteration, compute utility for all bases similar to greedy and add the one with highest marginal gain

Compute the marginal gain for the best known unassigned base and update its upper bound on gain.

If gain for best known base for current iteration is better than all the upper bound in gain for other bases, then the best base is already found

Continue until the marginal gain is higher than a threshold