Aalto University
School of Science

Solving the dynamic ambulance relocation and dispatching problem

Einari Tuukkanen
Presentation *26*
04.12.2020

MS-E2191 Graduate Seminar on Operations Research
Fall 2020

Ambulance relocation and dispatching problem

Objective

Minimize response time

Relocation

Where the ambulance should wait after service?

Dispatching

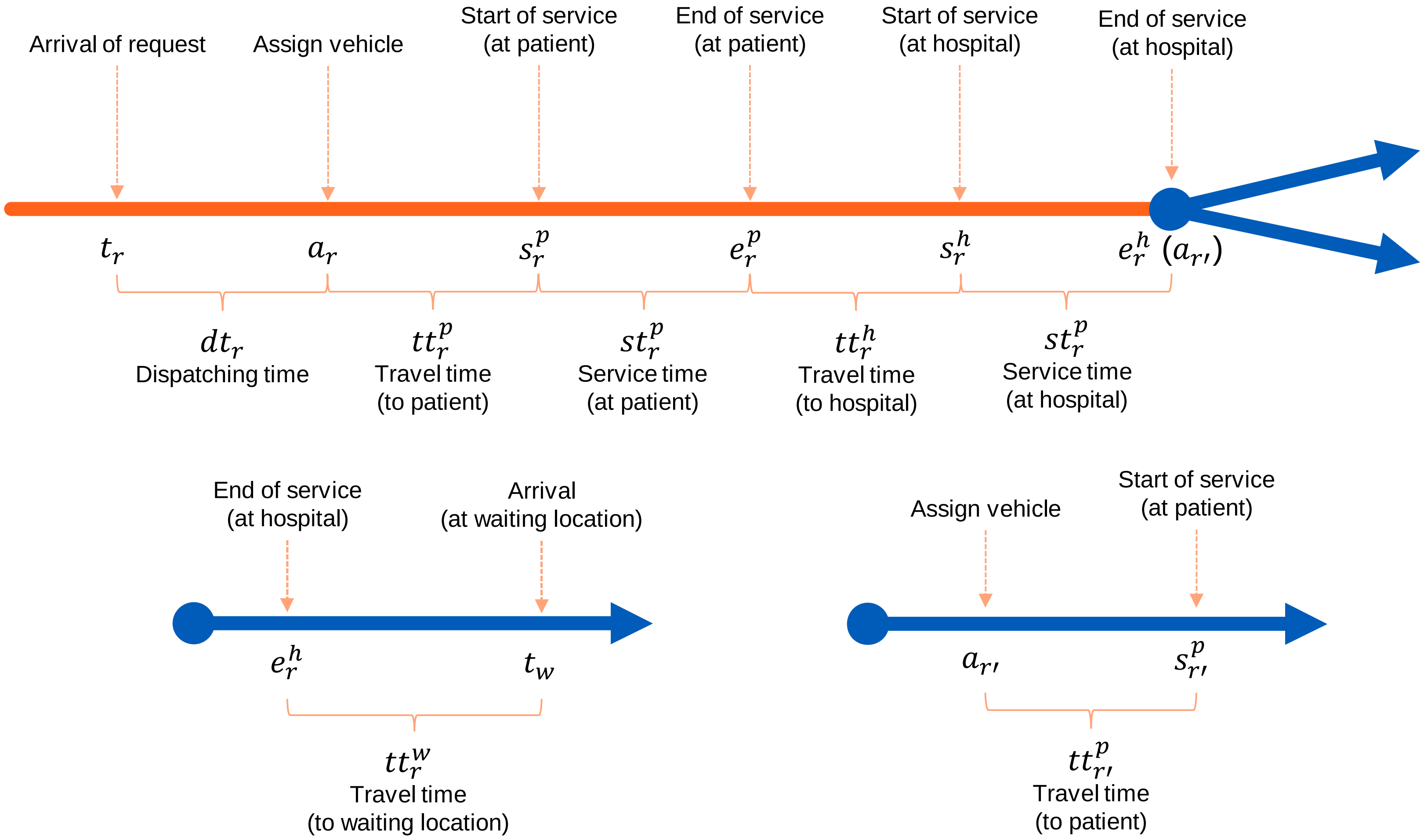
Which ambulance to dispatch?

Contribution of the paper

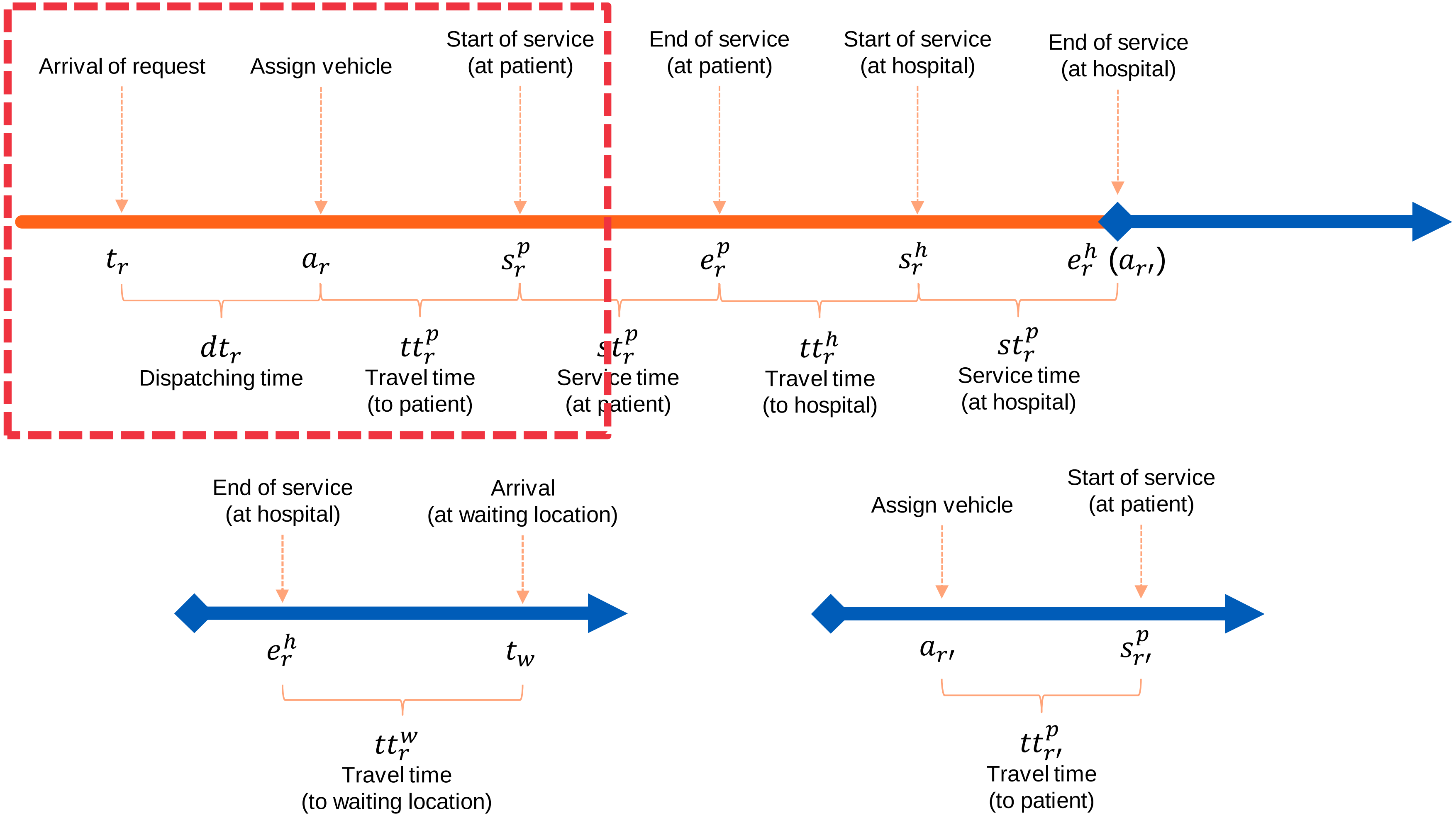
1. Construct stochastic dynamic model and solve with ADP
2. Take time-dependant information and variations into account
3. Improve dispatching and relocation strategies compared to the current ones

Aalto University
School of Science

Problem breakdown



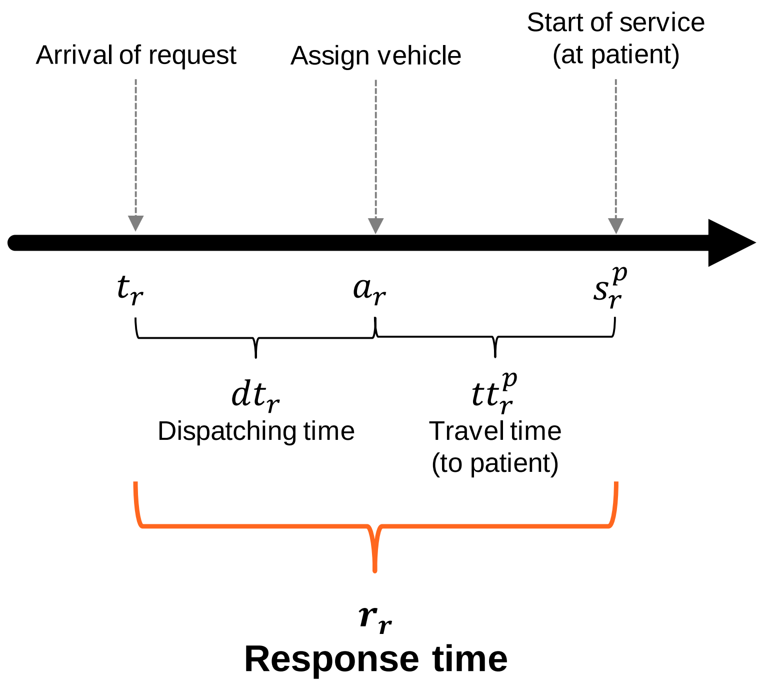
Problem breakdown



Objective function

Minimize response times

$$r_r = s_r^p - t_r$$



Aalto University
School of Science

Decisions are made at two events

When request is received

- Dispatcher decides which vehicle to send to a patient
- Served by first-come-first-served rule (queue allowed)
- All requests have equal priorities

When vehicle becomes idle

- After service, one must decide where the vehicle should be sent

System variables

- Emergency requests
  - Random
  - Short notice
  - → Distribution from real data

- Service times st_r^p and st_r^h
  - Random
  - → Distribution from real data

- Travel times tt_r^p, tt_r^h and tt_r^w
  - Time dependant

- Hospitals chosen according to historical data

Mathematical formulation

- State S_t
  - Status and location of all ambulances
  - Request queue
- Decision x_t
- Contribution $C(S_t, x_t)$

Myopic approach

$V_t(S_t) = \min_{x_t} C(S_t, x_t)$

Doesn't take future into account

→ **not realistic**

Mathematical formulation

- State S_t
  - Status and location of all ambulances
  - Request queue
- Decision x_t
- Contribution $C(S_t, x_t)$

- Solved by backward stepping
  - Calculate $V_t(S_t)$ from $V_{t+1}(S_{t+1})$
  - Requires evaluating $V_t(S_t)$ for all $S_t \in \boldsymbol{S}$

Bellmann's equation in expectation form

$V_t(S_t) = \min_{x_t}(C(S_t, x_t) + \mathbf{E}\{V_{t+1}(S_{t+1}(S_t, x_t, W_{t+1}))\})$,

where W_{t+1} denotes the random information

Dynamic evolution

$S_{t+1} = S^M(S_t, x_t, W_{t+1})$

Decisions

$\min_{\pi \in \Pi} \mathbf{E}[\sum_{t=0}^{T} \gamma^t C_t\left(S_t, X^\pi(S_t)\right)]$,

where decisions are made using a policy $X^\pi(S_t)$, Π denotes the family of all decision policies and γ is the discount factor.

Computationally difficult to solve

Aalto University
School of Science

Approximate dynamic programming (ADP)

- Capable of handling high-dimensional state spaces
  - Three curses of dimensionality
    - *State space with space vector* S_t
    - *Outcome space for the random variable* W_t
    - *Decision vector* x_t

- Using ADP we make decisions by **stepping forward**
  - Approximate value function

Optimize the sample estimate

$$\hat{v}_t^n = \min_{x_t}\left(C(S_t^n, x_t) + \gamma \bar{V}_t^{n-1}\left(S_t^{M,x}(S_t^n, x_t)\right)\right)$$

No need to calculate EV here!

Update the estimate of value function around the post-decision state variable

$$\bar{V}_{t-1}^n(S_{t-1}^{x,n}) = (1 - \alpha_{n-1})\bar{V}_{t-1}^{n-1}(S_{t-1}^{x,n}) + \alpha_{n-1}\hat{v}_t^n$$

Utilizes *spatial* and *temporal aggregation*, *post-decision state* and *step size* α.

Test scenario: Vienna

Vienna, Austria

- 1.7 million inhabitants (20% of Austria's population)
- Area 414.6 km^2
- Approximately twice the size of Helsinki

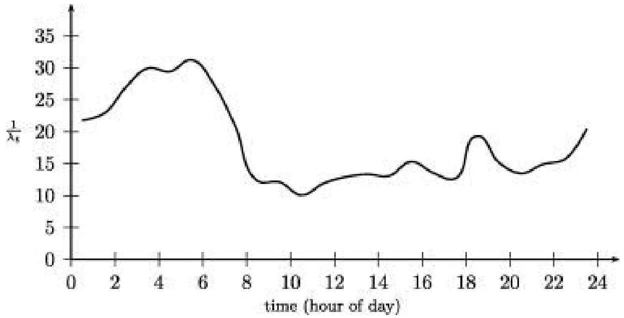
Regulations

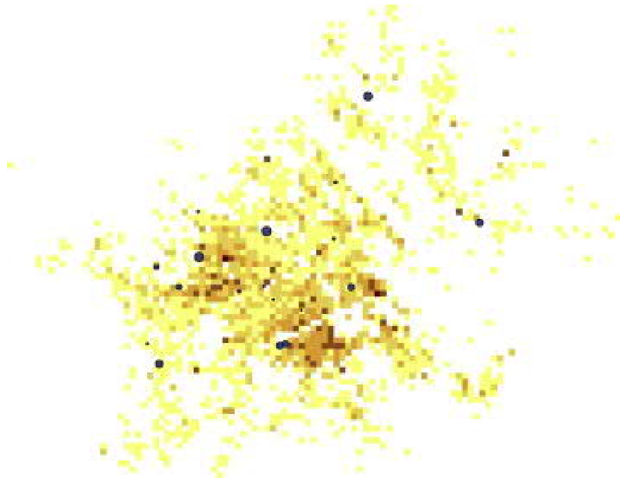
- Must always dispatch the closest ambulance
- Idle ambulances cannot be relocated
- Only idle vehicles can be dispatched

Aalto University
School of Science

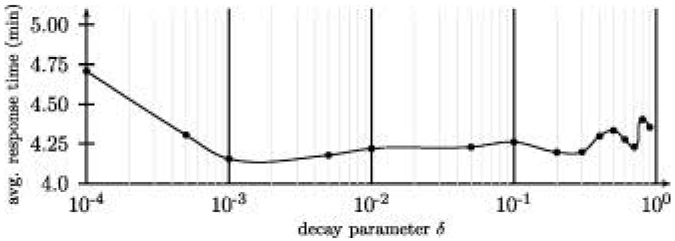
Test scenario: Vienna

- Data from a single service provider (WRK)
- Request data from 42 working days (11/2009)
  - ~90 calls/day
  - Average time between calls ~16 min
- 14 ambulances
- 16 waiting locations (max 2 vehicles per location)
- Real-world road network (with traffic)
- Real-world hospital and patient location distribution





Training parameters

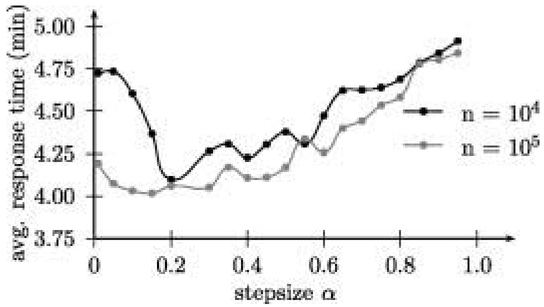


At iteration n, with probability $e^{-\delta n}$ use the myopic way

$$V_t(S_t) = \min_{x_t} C(S_t, x_t)$$

and with increasing probability $1 - e^{-\delta n}$ use the ADP method

$$\hat{v}_t^n = \min_{x_t} \left(C(S_t^n, x_t) + \gamma \bar{V}_t^{n-1} \left(S_t^{M,x}(S_t^n, x_t) \right) \right)$$

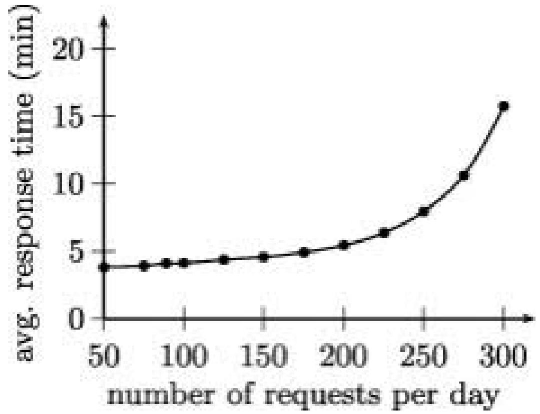


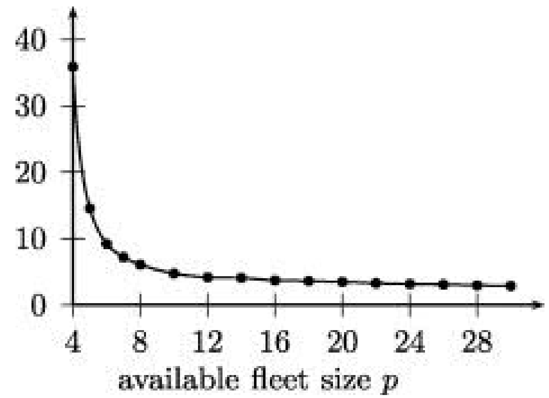
Smoothing parameter α

$$\bar{V}_{t-1}^n(S_{t-1}^{x,n}) = (1 - \alpha_{n-1})\bar{V}_{t-1}^{n-1}(S_{t-1}^{x,n}) + \alpha_{n-1}\hat{v}_t^n$$

Results

Waiting location	Average response time $\mu(r_r)$ (min)
Home	4.60
Closest	4.61
Random	5.12
ADP optimized	4.05
ADP optimized + allow dispatching any idle ambulance	4.01





Conclusion

- Solved dynamic ambulance dispatching and relocation problem using ADP
  - With extensive testing and real-world data, ADP provided a **12.89% better solution** compared to the current strategy
  - It can be shown that always sending the closest vehicle is not globally optimal
- Multiple possibilities for alternative applications and extensions

References

Schmid, V. (2012). Solving the dynamic ambulance relocation and dispatching problem using approximate dynamic programming. European journal of operational research, 219(3), 611-621

Also available here:
https://www.sciencedirect.com/science/article/pii/S0377221711009830

Homework

1. Besides ambulance dispatching & relocating, what other applications could the model (or slightly altered version of it) be used for?

2. How could the model be extended, for example, so that it better models real-world ambulance dispatching & relocating (or any alternative application of your choosing)?

Briefly justify your answers.

Some ideas can be found from the section *6. Conclusion and outlook* of the article.

DL: 9 am 11.12.2020 – return to einari.tuukkanen@aalto.fi