ANNUAL

CONFERENCE & EXPO | 2016

Talk Title

Speaker Name

Speaker Title

Dynamic Patient Priority Assignment for Emergency Medical Services

Laura A. McLay and Soovin Yoon

Department of Industrial and Systems Engineering
University of Wisconsin-Madison

Table of Contents

Background

Efficient emergency medical service design requires rationing limited medical resources through patient triage. (Explain why it's important to consider resource limitation by simple example)

Background

But traditional triage systems are static and do not depend on the changes in available resources.

START (Simple Triage And Rapid Treatment)

- The most widely used triage protocol in the United States
- Classifies patient into four different classes:
  Minor, *immediate*, *delayed*, *expectant* based on medical conditions
- Ignores resource limitations

STM (Sacco Triage Method)

- Proposed by Sacco et al.(2005,2007)
- Solves a linear program right after the incident to determine the patient priority
- Considers resource limitations but still static

Background

There have been many MCI patient prioritization literatures, (list some).
But those commonly assumes that all of the patients arrive at time zero and there is no(or very little) additional demand after the prioritization decision is made.
That assumption is not true in routine emergencies as well as some type of MCIs.
Often there are demands continuously arising for a long time horizon.

Table of Contents

1. Introduction

2. Model
    - Setting
    - Methodology

3. Structural Properties

4. Computational Results
    - Performance Study by Scenarios
    - Evaluation by the number of Mismatches
    - Sensitivity Analysis on the Model Parameters

5. Conclusion

6. References

Setting

Resources

Ambulances are differentiated by the types of treatment they can provide.

- Advanced Life Support(ALS)
  - staffed by paramedics to serve urgent calls
  - service rate μ_A
- Basic Life Support(BLS)
  - staffed by EMTs to serve less serious calls
  - service rate μ_B

Patients

Calls arriving at rate $\lambda = \sum_i \lambda_{ip}$ have

- types $i \in \{1, \ldots, m\}$: Any possible information that is correlated with the urgency of a call can be used as a type information.
  - Incident type
  - Geographic information
- pre-assigned priorities $p \in \{1, 2, 3\}$

Setting

Information

The true urgency of an arriving call with priority 2 is known to the decision maker only probabilistically on its call type, with parameter $\alpha^i = P(\text{urgent}|\text{call type}=i,\text{priority}=2)$

We get different utility depending on the match between the server type and true priority of the call.

	Urgent	**Not urgent**
$a^i = 0$	U_{HA}	U_{LA}
$a^i = 1$	U_{HB}	U_{LB}

- We assume $U_{HA} > U_{LA}$ and $U_{HA} > U_{LB}$ so that we get more benefit by serving a high priority call with a ALS than serving a low priority call with any type of server
- We assume $U_{HA} > U_{HB}$ to penalize under-service of urgent calls

Setting

Reachability

Since ambulances are spatially located, only a subset can respond in a timely fashion. The probability that an arriving high(low) priority call is served successfully in-time is modelled as a non-increasing function $f^H(s)(f^L(s))$. The argument s is $s^A(s^B)$ when the dispatcher sends an ALS(BLS) to the call.

Therefore, when s^A ALSs and s^B BLSs are busy, we use

	High priority	**Low priority**
send ALS	$f^H(s^A)$	$f^L(s^A)$
send BLS	$f^H(s^B)$	$f^L(s^B)$

We implement faster deterioration rate of condition for urgent patients by assuming that $\nabla f^H(s) < \nabla f^L(s)$.

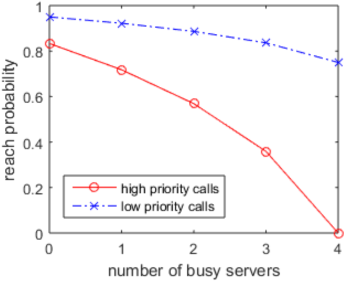

Figure: Example Reach Function

MDP Model

Time $t \in \{1, \cdots, T\}$
By uniformization, we get discrete time epochs.

State $s_t = (s^A, s^B)$
where $s^A(s^B)$ is the number of busy ALS(BLS) servers.

Transition $P_t(j|s_t, a_t)$
at most 1 event happens among:

- An urgent call arrives $(s^A, s^B) \to (s^A + 1, s^B)$
- A less urgent call arrives $(s^A, s^B) \to (s^A, s^B + 1)$
- An ALS server finishes service $(s^A, s^B) \to (s^A - 1, s^B)$
- A BLS server finishes service $(s^A, s^B) \to (s^A, s^B - 1)$

MDP Model

Action $a_t = (a_t^1, \cdots, a_t^i, \cdots, a_t^m)$
Based on the of type(i) and priority of an arriving call, assign either an ALS ($a^i = 0$) or BLS ($a^i = 1$), depending on the system congestion (s^A, s^B).

MDP Model

Reward $R_t(s_t, a_t)$
We only get rewards when there is a call arrival. The reward from the service is dependent on the action taken and state transition.
At time epoch t, if a call of type i arrives and our action choice for type i is to send a ALS ($a_t^i = 0$) then we get the conditional reward of

$$\begin{aligned} R_t(j, s_t, a_t) &= R_t((s^A + 1, s^B), (s^A, s^B), (a_t^1, \cdots, a_t^m)) \\ &= \alpha^i U_{HA} f^H(s^A) + (1 - \alpha^i) U_{LA} f^L(s^A) \end{aligned}$$

Analogously, if we send a BLS ($a_t^i = 1$) we get

$$\begin{aligned} R_t(j, s_t, a_t) &= R_t((s^A, s^B + 1), (s^A, s^B), (a_t^1, \cdots, a_t^m)) \\ &= \alpha^i U_{HB} f^H(s^B) + (1 - \alpha^i) U_{LB} f^L(s^B) \end{aligned}$$

MDP Model

To get the expected reward, we weight each conditional reward by transition probability.

$$R_t(s_t, a_t) = \sum_j P_t(j|s_t, a_t) R_t(j, s_t, a_t)$$

Therefore given $s_t = (s^A, s^B)$ and $a_t = (a^1, \cdots, a^i, \cdots, a^m)$, we compute the expected reward as

$$\begin{aligned} R(s^A, s^B, a^1, \ldots, a^m) = & \sum_i \lambda_t^{i1} (\alpha^{i1} U_{HA} + (1 - \alpha^{i1}) U_{LA}) f^A(s^A)/\Lambda \\ & + \sum_i \lambda_t^{i2} (1 - a^i)(\alpha^{i2} U_{HA} + (1 - \alpha^{i2}) U_{LA}) f^A(s^A)/\Lambda \\ & + \sum_i \lambda_t^{i2} a^i (\alpha^{i2} U_{HB} + (1 - \alpha^{i2}) U_{LB}) f^B(s^B)/\Lambda \\ & + \sum_i \lambda_t^{i3} (\alpha^{i3} U_{HB} + (1 - \alpha^{i3}) U_{LB}) f^B(s^B)/\Lambda \end{aligned}$$

Solution Methodology

Finite-time discrete MDP is solved by backward induction to maximize total expected reward.

$$U_t(s_t) = \sup_a \{R_t(s_t, a) + \sum_j P_t(j|s_t, a) U_{t+1}(j)\}$$

The size of possible action set to be evaluated at each time epoch t grows exponentially with the number of type m, but we don't really have to evaluate the whole set to find the optimal solution, due to the structural property of the problem that follows.

Table of Contents

1. Introduction
2. Model
   - Setting
   - Methodology
3. Structural Properties
4. Computational Results
   - Performance Study by Scenarios
   - Evaluation by the number of Mismatches
   - Sensitivity Analysis on the Model Parameters
5. Conclusion
6. References

Type Independence of Optimal Action

Proposition 1

For any time epoch t and state s, the optimal action for the call type i is to send an ALS server if and only if the following equality is true:

$$\alpha^{i2}U_{LA}f^{H}(s^{A})+(1-\alpha^{i2})U_{LA}f^{L}(s^{A})>\alpha^{i2}U_{HB}f^{H}(s^{B})+(1-\alpha^{i2})U_{LB}f^{L}(s^{B}))$$

which does not depend on a^{k} for all $k\in\{1,\cdots,m\},k\neq i$.

Proposition 1 implies that the optimal decision of sending an ALS server or a BLS server to type i call can be made regardless of decision for other call types. Therefore, the number of action we have to evaluate at each time epoch t to solve by the backward induction can be reduced from exponential 2^{m} to linear m.

Optimality of Threshold-Type Policy

Proposition 2

For any time epoch t and state s, a threshold value $\bar{\alpha}_t(s)$ can be specified such that it is optimal to send ALS server to type i call if and only if $\alpha^i > \bar{\alpha}_t(s)$, if

$$U_{HA}f^H(s^A) - U_{HB}f^H(s^B) - U_{LA}f^L(s^A) + U_{LB}f^L(s^B) > 0.$$

Optimality of Monotone Policy

Proposition 3

For each call type i, the optimal action a_t^i is

1. *non-decreasing in s^A if the value function $V_t(s^A, s^B)$ is concave in s^A*
2. *non-increasing in s^B if the value function $V_t(s^A, s^B)$ is concave in s^B,*

and the value function $V_t(s^A, s^B)$ is supermodular in (s^A, s^B).

Corollary 1

Value function $V_t(s^A, s^B)$ is

1. *Monotone non-increasing in s^A and s^B.*
2. *Convex(Concave) in s^A and s^B, if $f^H(s)$ and $f^L(s)$ is convex(concave) in s.*
3. *Supermodular.*

Table of Contents

1 Introduction

2 Model
- Setting
- Methodology

3 Structural Properties

4 Computational Results
- Performance Study by Scenarios
- Evaluation by the number of Mismatches
- Sensitivity Analysis on the Model Parameters

5 Conclusion

6 References

Parameter Setup by Scenarios

Evaluation by the number of Mismatches

The decision maker may be interested not only in the total expected QoL but also in the nominal number of patients that are mismatched to the type of service.

Sensitivity Analysis on the Arrival Rates

Show that the dynamic policy is even more effective when there happens unexpected demand increase.

Sensitivity Analysis on the Reachability Function

Show that the dynamic policy is robust on inaccurate model parameters(reachability function and alpha - that the dynamic policy is better than case 1, case 2, and any arbitrary static policy). For reachability, check the literature to decide the model function(possibly log/log-logistic/..) and give a -20 +20 percent perturbation on its function parameter.(check Mills et al.)

Sensitivity Analysis on the Type Information

Show that the dynamic policy is robust on inaccurate model parameters(reachability function and alpha - that the dynamic policy is better than case 1, case 2, and any arbitrary static policy). For alpha, treat it like a Bernoulli variable, and perturb the value by simulation.

Table of Contents

1. Introduction
2. Model
   - Setting
   - Methodology
3. Structural Properties
4. Computational Results
   - Performance Study by Scenarios
   - Evaluation by the number of Mismatches
   - Sensitivity Analysis on the Model Parameters
5. Conclusion
6. References

Table of Contents

1. Introduction
2. Model
   - Setting
   - Methodology
3. Structural Properties
4. Computational Results
   - Performance Study by Scenarios
   - Evaluation by the number of Mismatches
   - Sensitivity Analysis on the Model Parameters
5. Conclusion
6. References