HARVARD Kennedy School
JOHN F. KENNEDY SCHOOL OF GOVERNMENT

Workload Management in Telemedical Physician Triage and Other Knowledge-Based Service Systems

Faculty Research Working Paper Series

Soroush Saghafian
Harvard Kennedy School

Wallace J. Hopp
University of Michigan, Ann Arbor

Seyed M.R. Iravani
Northwestern University

Yao Cheng
Northwestern University

Daniel Diermeier
University of Chicago

August 2017

RWP17-035

Workload Management in Telemedical Physician Triage and Other Knowledge-Based Service Systems

Soroush Saghafian[1], Wallace J. Hopp[2], Seyed M. R. Iravani[3], Yao Cheng[3], Daniel Diermeier[4]

[1] *Harvard Kennedy School, Harvard University, Cambridge, MA*
[2] *Ross School of Business, University of Michigan, Ann Arbor, MI*
[3] *Dept. of Indust. Eng. and Man. Sci., Northwestern University, Evanston, IL*
[4] *Office of the Provost, University of Chicago, Chicago, IL*

Abstract

Telemedical Physician Triage (TPT) is an example of a Hierarchical Knowledge-Based Service System (HKBSS), in which a second level of decision agent (telemedical physician) renders a decision on cases referred to him/her by the primary level agents (triage nurses). Managing the speed-versus-quality tradeoff in such systems presents a unique challenge because of the interplay between agent knowledge and flow of work between the two levels. We develop a novel model of agent knowledge, based on the beta distribution, and deploy it in a Partially Observable Markov Decision Process (POMDP) model to describe the optimal policy for deciding which cases (patients) to refer to the second level for further evaluation. We show that this policy has a monotone control-limit structure that reduces the fraction of decisions made at the upper level as workload increases. Because the optimal policy is complex, we use structural insights from it to design two practical heuristics. These heuristics enable an HKBSS to adapt efficiently to workload shifts by adjusting the criteria for referring decisions to the upper level based on partial real-time queue length information. Finally, we conduct analytic and numerical analyses to derive insights into the management of a TPT system. We find that (1) the telemedical physician should evaluate more patients as congestion in the emergency room waiting area increases, (2) training that improves accuracy of the physician and/or nurses can be effective even if it only does so for a single patient type, but training that improves consistency must do so for all patient types to be effective, and (3) patient classification in triage should consider environmental and operational conditions in addition to the patient's medical condition.

Keywords: Decision Flow Network; Knowledge-Based Decision Making; Telemedical Triage; POMDP.

1 Introduction

Triage is the process used by hospital Emergency Departments (EDs) to assess patient urgency, traditionally through a short interview by a triage nurse.[1] In addition to speed, accuracy is vital in triage because errors in classification lead to errors in prioritization, which can cause dangerous delays in treating urgent patients (see, e.g., Wiler et al. (2010), FitzGerald et al. (2010), Saghafian et al. (2012, 2014), Traub et al. (2015), and the references therein). Seeking to improve triage decisions, some hospitals have begun experimenting with *Telemedical Physician Triage* (TPT) (see, e.g., Traub et al. (2013)).

In TPT, a triage nurse has the option after examining a patient to refer that patient to a telemedicine booth through which a remote physician conducts a video conference and renders a triage decision. The Telemedical Physician (TP) typically services multiple hospitals, and hence patients referred to him/her may have to wait in a queue. Therefore, when considering referring a

[1] Most EDs in the U.S. use the five level Emergency Severity Index (ESI) system. ESI-1 patients are life threatening emergencies that are routed immediately to a resuscitation area, while ESI-4 and 5 patients are simple enough to be sent to a fast track. The patients that remain in the main ED are ESI-2 patients, who cannot wait without clinical risk and hence are urgent, and ESI-3 patients who can wait without risk and hence are non-urgent.

patient to the TP, a triage nurse must balance the queueing delay with the benefit from a review by the more knowledgeable physician.

We use the term *Hierarchical Knowledge-Based Service System* (HKBSS) to refer to a system like TPT in which hierarchically organized agents with different knowledge levels assess cases and must either issue a decision or refer the case to a higher level. In the TPT setting, there are only two levels of hierarchy (i.e., triage nurses and TP) and the decisions are binary (i.e., patients are either urgent or not). Other HKBSS examples with two levels and binary decisions include the U.S. Department of State Bureau of Consular affairs, in which agents must decide whether or not to grant visa applications and can refer cases to a supervisor, and the mortgage department of a bank, in which loan officers must decide whether or not to approve mortgages and can refer cases to a manager.

While speed versus quality tradeoffs are common in operations management, those of an HKBSS present a unique modeling challenge to incorporate a representation of agent knowledge and decisions into a queueing model. Such a model is needed to address the following questions: (1) What is the structure of decisions at the lower and upper levels that strikes an optimal balance between speed and quality? (2) How do environmental factors (e.g., costs of decision errors, case mix, fluctuations in workload, etc.) affect the optimal policy? (3) What levers (e.g., agent training, information sharing between levels, etc.) are effective for improving system performance? (4) Are there simple methods that can be used as practical policies for managing case flow in a real-world HKBSS?

By addressing these questions in the context of TPT we generate new insights into the effective management of telemedical physician triage, and also provide the analytic building blocks for evaluating any similar HKBSS.

2 Related Studies

Researchers have explored knowledge-based organizations in the context of social networks and their effects on organizational performance (see, e.g., Albrecht and Ropp (1984), Brass (1995), Burt (1992), and Huberman and Hogg (1995)). While promising, these knowledge management studies only address the *information flow* aspect of an HKBSS.

In contrast, the field of operations management has focused predominantly on the *work flow* aspect. There are only a few papers in the operations management literature that have combined task flow and information/decision making. Among them, Shumsky and Pinker (2003) studied a two-level system that processes tasks with different levels of complexity. Using a principal-agent framework, they focused on the impact of performance-based incentives at the lower level, and investigated incentives that would lead to system-optimal referral rates. However, unlike our study (i) they do not explicitly model knowledge-based decision making, and (ii) they do not address the ability of workload management policies to react to workload spikes.

Motivated by call centers that provide medical advice, Wang et al. (2010) considered the analysis of diagnosis service systems where one must strike a balance between accuracy of advice, waiting time, and capacity/staffing costs. In their study, a longer service entails higher accuracy and a higher congestion. They use a M/G/K model to balance these. But, because they consider only a single level of hierarchy, they do not consider the issue of balancing quality and speed by routing cases between levels, as we do in this paper.

In another study with a single level of hierarchy, Anand et al. (2011) examined the trade-off between service quality and speed by modeling quality as a linearly decreasing function of service rate. In our two-level framework, a better decision can be made by a more knowledgeable agent, but only if the case is transferred to the upper level. Because this involves a workload-dependent delay in the queue of the upper level agent, the relationship between quality and time is nonlinear.

Another study that considered a quality and speed trade-off is Hopp et al. (2007) which modeled the ability of employees to determine how much time to allocate to customers. In their work, an agent can decide when to terminate a "discretionary service," and the reward is an increasing concave function of service time. However, in their model customers are homogenous and service providers are assumed to be perfectly knowledgeable. In contrast, in our study, agents are neither homogenous nor perfectly knowledgeable: they differ (across tiers) in skill level, and the accuracy of their judgment depends on their skill levels. Another distinct feature of our study is the consideration of a two-level hierarchy instead of a traditional single-level service system.

Alizamir et al. (2013) also considered a single-level service system in which an agent can decide to perform more diagnostic tests to improve the quality of the customer classification but at the expense of more delays. Similar to our work, they studied the effect of congestion, but unlike ours, only one case can be processed at a time in their study, and there is no queue formed after the first test is performed. Finally, their representation of knowledge via the quality of sequential Bayesian tests is less descriptive than our beta distribution model, and hence, cannot depict decision consistency and other decision characteristics beyond accuracy. Thus, their model is not fully suited to studying agent training, assessment sharing, and other relevant policies to improving the performance of an HKBSS.

Rajan et al. (2015) studied the speed-quality tradeoff in telemedicine for treating chronically ill patients. We also consider the use of tele-medicine in this paper, but rather than focusing on how to increase the use of this technology, we focus on how to maximize its effectiveness in a triage

Reference	Hierarchical Decision Levels	Balking or Reneging	Separate False Positive and False Negative Error Costs	Assessment Sharing Scenarios	Knowledge-Based Decision Making	Workload Fluctuations
Shumsky and Pinker (2003)	✓					
de Véricourt and Sun (2009)					✓	
Wang et al. (2010)			✓		✓	
Anand et al. (2011)		✓				
Alizamir et al. (2013)			✓		✓	
Debo and Veeraraghavan (2014)		✓				

Others: Hopp et al. (2007);
Kostami and Rajagopalan (2013);
Tan and Netessine (2014);
Wang et al. (2015);						

Our setting	✓	✓	✓	✓	✓	✓

Table 1: Summary of related studies on speed-quality tradeoffs.

setting.

To consider the judgement accuracy/congestion tradeoff, de Véricourt and Sun (2009)[2] modeled the decision process of a service provider using binary probabilistic cues. In their framework, customers belong to one of two possible types, and accuracy is defined as the probability that the customer type is correctly identified assuming that only false negatives may occur. We broaden this representation of server knowledge by (1) using beta distributions to model the knowledge level of the decision makers, which enables consideration of consistency as well as accuracy of assessments, (2) considering both class-based false negative and false positive errors and their costs by allowing the assessment of the beta distributions to depend on the true class of the customer, and (3) considering more than a single level of hierarchy in the system to explicitly connect work flows and knowledge-based decisions.

Table 1 provides a comparative summary of the extent to which available studies on speed-quality tradeoffs address the key features of an HKBSS. Incorporating these features is an important part of our contribution.

Finally, we note that some of the elements of Table 1 have also appeared in studies that do not consider speed-quality tradeoffs. For instance, similar to our work, Bassamboo et al. (2006) studied workload spikes in queueing networks. They characterized asymptotically optimal policies for dynamic routing in such networks with varying arrival rates. However, unlike our study, they assumed that the customer classes/characteristics are perfectly known to decision makers. Hence, they did not consider decision making errors, which are a key element of almost any HKBSS.

3 Modeling Hierarchical Knowledge-Based Service Systems (HKBSSs)

We now turn to the development of a formal model to analyze the performance of HKBSSs. We begin by summarizing common characteristics of these systems:

[2] This paper seems to be an earlier version of Alizamir et al. (2013).

- **Hierarchical Structure.** HKBSSs have a hierarchical structure that allows a lower level agent to pass a case to a higher level for a decision. In this paper, we focus on two-level systems in which cases are first assessed at the lower level (e.g., triage nurses), and are then sent to the upper level agent (e.g., the Telemedical Physician (TP)) if needed.

- **Limited Capacity.** Since agents at both levels require time to process cases and make decisions, the system has a limited capacity at each level. Because of this, customers may balk if delays are too long. In the TPT system, the ED may go on a "diversion", which signals ambulances to take patients to another hospital, if congestion becomes severe. Patients may also leave without being seen if waits are too long (although typically this occurs after triage is done and the patient is in the waiting area of the ED). Whether a patient balks due to sever congestion at the lower or upper level, the result is costly to the hospital in both financial and reputational terms. However, to be thorough, we also consider HKBSSs in which balking is not an issue. We also discuss how our findings are relevant to HKBSSs in which reneging is more prevalent than balking.

- **Binary Primary Decisions.** In many HKBSSs, including TPT, the primary decisions are binary. As noted earlier, in visa processing and consumer loan evaluation, the decisions are restricted to "yes" or "no". In the TPT system, the key triage decision is whether the patient is urgent or not.

- **Knowledge-Based Decisions.** The quality of decisions depends on the decision maker's knowledge level. For instance, in the TPT system, triage nurses and the TP make judgments about whether the patient is urgent or not based on their skill and experience.

- **Speed-Quality Tradeoff.** Sending cases to an upper level agent (e.g., the TP) results in higher quality decisions (i.e., fewer decision errors) but also additional delay and congestion at the upper level agent. Because of queueing effects, this tradeoff is particularly pronounced in systems with highly utilized servers (e.g., overcrowded EDs).

These characteristics provide us with a foundation for modeling the performance of an HKBSS. Specifically, we consider the two-level hierarchy shown in Figure 1, in which there are K agents at the lower level (ℓ), labeled as agents $1, 2, \ldots, K$, and one agent at the upper level (u), designated as agent 0.

We assume that cases arrive at each of the lower level agents at rate λ_ℓ, so that each agent has his/her own queue of cases. We consider the scenario in which each lower level agent serves a separate queue, because it better represents a TPT setting in which triage nurses are in different (e.g., pediatric and main) EDs within a hospital, or are in different hospitals. However, since our focus is mainly on systems that are subject to high utilization, there is little difference between systems with a pooled queue and a system with separate queues. Furthermore, because the time of the upper level agent (e.g., the TP) is particularly valuable, we assume the system uses a protocol of not sending cases to the upper level without first examining them at the lower level.

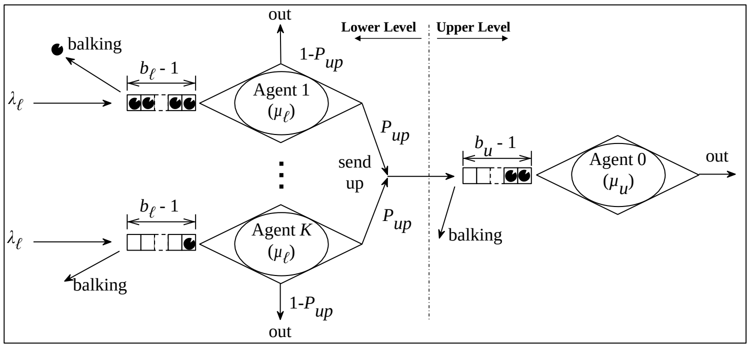

Figure 1: Modeling network flows in a two-level knowledge-based service system.

To model the possibility of cases not being handled due to congestion, we assume there is a limit on the agents' queue lengths beyond which arriving cases do not enter the system. Specifically, if a case arrives when the number of cases in a lower level agent's queue (including the case in service) is b_ℓ, then the case does not enter the system and leaves without a decision. When a case does enter the system, the lower level agent either makes a final decision and releases the case from the system, or passes it to the upper level. The upper level agent also has a queue of cases and, similar to the lower level, if a case arrives to the upper level agent when her queue is full (i.e., there are b_u cases at the upper level including one in service), then the case does not enter the upper level and leaves the system without a decision (e.g., a patient leaves without being seen). Otherwise, the upper level agent processes the case, makes a final decision and releases the case from the system.[3]

Our model is general enough to cover HKBSSs with no congestion related rejection by setting b_ℓ and b_u to large numbers. For instance, setting b_u very high represents a system like TPT in which customers (patients) rarely balk after being referred to the upper level (telemedical physician). Furthermore, since queue lengths are considered when making referral decisions, our main findings can be shown for an HKBSS with reneging instead of balking. We choose balking simply because it better represents our main motivating example.

We assume that the processing time of a case by an agent at level j ($j = \ell, u$) follows a known distribution with a finite mean $1/\mu_j$ and is independent of the agent's decision. In most of the paper, we assume the interarrival and service times are exponentially distributed, but we note that generalizations are possible. Since customer satisfaction is key in service systems, we characterize service quality via a holding cost as well as a congestion related balking cost, which might not be symmetric across the two hierarchical levels.

[3] The model takes queue lengths into account, and hence, optimizing it will determine whether it is better to refer cases to the upper level and incur congestion related rejection/balking cost there, or to make a less accurate decision at the lower level without any referral.

Another important aspect of HKBSSs is their ability to make correct decisions. Therefore, we also penalize wrong decisions by incorporating decision error costs at each level. In the TPT system, the challenge is to determine whether patients are urgent (yes) or not (no). Hence, in TPT or any other binary HKBSS, arrivals can be divided into cases for which the correct decision is $Y = 1$ (yes), and cases for which the correct decision is $Y = 0$ (no). We let p_ℓ^1 denote the fraction of cases arriving at the lower level for which $Y = 1$, and $(1 - p_\ell^1)$ denote the fraction of cases for which $Y = 0$. We further assume that p_ℓ^1 is known, and $0 < p_\ell^1 < 1$. However, we assume that the correct decision Y for an individual case is not observable by an agent when the decision is made. If the value of Y were known, the agent would always make a correct decision and the check provided by the upper level would be redundant. In such a scenario, the lower level would become a traditional flow network in which the only concerns would be the congestion related balking and holding costs. However, because the correct decisions are not known in our framework, agents must examine each case and, based on their knowledge and judgement, make a (possibly incorrect) decision. Moreover, since lower level agents may elect to route cases to the upper level, decisions can significantly affect the flow of cases.

After processing a case, the agent at level j ($j = \ell, u$) makes a decision a_j, where the lower level decision $a_\ell \in \{0, 1, UP\}$ and the upper level decision $a_u \in \{0, 1\}$. When an UP decision is made, the case is passed on to the upper level. Hence, the final decision for a case is always 0 or 1 (as long as it is not balked due to system congestion). To consider the cost of decision errors, we let c_1 (c_0) denote the error cost when a case's correct decision is $Y = 1$ ($Y = 0$) but the agent's decision is $a_j = 0$ ($a_j = 1$). In TPT, c_1 is the cost of misclassifying an urgent patient as non-urgent (which leads to a longer downstream wait and hence more risks), while c_0 is the cost of misclassifying a non-urgent patient as urgent (which may put him/her ahead of urgent patients who arrive later and increase their downstream wait and risks). We represent the vector of decision error costs by $\mathbf{c}_e \triangleq (c_0, c_1)$.[4] Similarly, we let $\mathbf{c}_b \triangleq (c_{b,\ell}, c_{b,u})$, and $\mathbf{h} \triangleq (h_\ell, h_u)$ denote the vector of balking and holding costs, respectively, where for generality we allow for asymmetric costs at the lower and upper levels. Since the total cost in our setting considers both the network flow component (holding and rejection/balking costs) and the decision making component (decision error costs), these two aspects are linked in our model; an optimal decision rule is one that strikes a speed-versus-quality balance between these two components.

To formalize the optimization model, we make use of the following notation. We represent by $\pi = \big(\pi_\ell, \pi_u\big)$ a control policy that jointly prescribes the agents' actions at the lower and upper levels at any time. With $\mathcal{K} \triangleq \{1, 2, \cdots, K\}$ denoting the set of all lower level agents, we represent

[4] We use a bold font to denote vectors and matrices.

the vector of queue lengths at the lower level at time t by $\mathbf{N}_\ell(t) \triangleq \big(N_k(t) : k \in \mathcal{K}\big)$. Similarly, we let $N_u^{\pi_\ell}(t)$ denote the queue length at the upper level at time t, which depends on the control policy adopted at the lower level, π_ℓ. We also let $\mathbf{E}_\ell^{\pi_\ell}(t) \triangleq \big(\mathscr{E}_{\ell,0}^{\pi_\ell}(t), \mathscr{E}_{\ell,1}^{\pi_\ell}(t)\big)$ and $\mathbf{E}_u^{\pi_\ell,\pi_u}(t) \triangleq \big(\mathscr{E}_{u,0}^{\pi_\ell,\pi_u}(t), \mathscr{E}_{u,1}^{\pi_\ell,\pi_u}(t)\big)$ denote the vector of cumulative number of decision errors up to time t at the lower level and upper level, respectively, where $\mathscr{E}_{j,y}^{\pi_j}(t)$ denotes the cumulative number of decision errors made up to time t under policy π_j for cases with correct decision $y \in \{0,1\}$. Also, let $B_l(t)$ and $B_u^{\pi_l}(t)$ be the cumulative number of customers who balked up to time t at the lower and upper level, respectively.

Using the notation above, the long-run average total cost of the system under a control policy $\big(\pi_\ell, \pi_u\big)$ is:

$$\begin{aligned} \varphi(\pi_\ell, \pi_u) \quad &\triangleq \quad \liminf_{t\to\infty} \frac{1}{t}\,\mathbb{E}\Big[\mathbf{c}_e\big(\mathbf{E}_\ell^{\pi_\ell}(t) + \mathbf{E}_u^{\pi_\ell,\pi_u}(t)\big)^T + \mathbf{c}_b\big(B_l(t), B_u^{\pi_l}(t)\big)^T \\ &\qquad + \int_0^t \mathbf{h}\big(\mathbf{N}_l(s)(\mathbf{1}_K)^T, N_u^{\pi_\ell}(s)\big)^T ds\Big], \end{aligned} \tag{1}$$

where "^{T}" represents the transpose operator, and $\mathbf{1}_K$ is a K-dimensional vector with all elements equal to 1. Finally, with Π_ℓ and Π_u denoting the set of all admissible (non-anticipative) policies at the lower and upper levels, respectively, we seek to find

$$(\pi_\ell^*, \pi_u^*) = \operatorname*{argmin}_{\pi_\ell \in \Pi_\ell,\ \pi_u \in \Pi_u} \varphi(\pi_\ell, \pi_u), \tag{2}$$

and refer to $\varphi^* \triangleq \varphi(\pi_\ell^*, \pi_u^*)$ as the optimal long-run average cost.

3.1 Modeling Knowledge

The decision-making in an HKBSS is based on agent "assessments" which in turn depend on their *knowledge/experience* level. An agent at level j makes an assessment $X_j = x$, where $x \in [0,1]$. The value of X_j may be explicit (e.g., a normalized credit score) or implicit (e.g., an estimate of agent's confidence that the correct decision is $Y = 1$). In either case, the value of x depends on the agent's knowledge. We use the beta distribution to model the knowledge level of an agent as follows. If the correct decision for a case is $Y = y$ ($y = 0, 1$), then the assessment by an agent at level j will be a random variable X_j^y with a beta probability density function $f_j^y(x)$ with parameters α_j^y and β_j^y:

$$f_j^y(x) = \frac{1}{\mathbb{B}(\alpha_j^y, \beta_j^y)} x^{\alpha_j^y - 1}(1-x)^{\beta_j^y - 1},$$

where $\mathbb{B}(\alpha_j^y, \beta_j^y) = \int_0^1 v^{\alpha_j^y - 1}(1-v)^{\beta_j^y - 1} dv$ is the beta function. When needed, we assume $\alpha_j^y \geq 1, \beta_j^y \geq 1$ to ensure that f is bounded, which also implies that assessments are continuous on the interval $[0,1]$. Examples of such beta distributions are shown in Figure 2. Notice that assessment

X_j is equal to X_j^1 (to X_j^0) with probability p_j^1 (with probability $1-p_j^1$). Hence, the density function of X_j is $f_j(x) = p_j^1 f_j^1(x) + (1-p_j^1)f_j^0(x)$, where p_j^1 denotes the fraction of cases entering level j for which the correct decision is $Y = 1$.[5] We denote by $F_j^y(x)$ and $F_j(x)$ the cumulative distribution functions corresponding to densities $f_j^y(x)$ and $f_j(x)$, respectively.[6]

The relationship between the parameters of the beta distribution and an agent's knowledge becomes clearer when we consider the mean and variance of the beta distribution. The mean of the above beta distribution, $\mathbb{E}[X_j^y]$, is an indication of how close, on average, a level j agent assessment is to the correct decision y. We refer to this mean as the *accuracy* of the agent. In contrast, $Var(X_j^y)$ is an indication of the *consistency* of agent k's assessments of cases for which the correct decision is y $(y = 0, 1)$. Accuracy and consistency together describe the *knowledge* of the agent. A more knowledgeable agent typically has a better accuracy (i.e., $\mathbb{E}[X_j^y]$ closer to y) and better consistency (i.e., lower $Var(X_j^y)$) regardless of whether $y = 0$ or $y = 1$, although we do not restrict our framework to such an assumption.

Using our notation, the mean and variance formulations of the beta distribution can be written as:

$$\mathbb{E}[X_j^y] = \frac{\alpha_j^y}{\alpha_j^y + \beta_j^y} \quad ; \quad Var(X_j^y) = \frac{\alpha_j^y \beta_j^y}{(\alpha_j^y + \beta_j^y)^2(\alpha_j^y + \beta_j^y + 1)}.$$

From these it is apparent that as α_j^0 decreases (or β_j^0 increases), the mean becomes closer to 0, so the agent has more accuracy when $Y = 0$. Similarly, as α_j^1 increases (or β_j^1 decreases), the agent has more accuracy when $Y = 1$. Moreover, when α_j^y and β_j^y increase proportionally, the variance is reduced, implying that the agent is more consistent in his/her assessments.

This flexible framework enables us to model a wide range of agent knowledge scenarios including:

- **Perfect Knowledge:** When $\alpha_j^0 = 1$, $\beta_j^0 \to \infty$, $\beta_j^1 = 1$, and $\alpha_j^1 \to \infty$, we have $X_j^0 \xrightarrow{a.s.} 0$ and $X_j^1 \xrightarrow{a.s.} 1$, which implies that the agent at level j has perfect knowledge about both cases with correct assessment value $Y = 1$ and $Y = 0$, and therefore will never make a mistake.
- **No Knowledge:** When $\alpha_j^y = \beta_j^y = 1$ $(y = 0, 1)$, the beta distribution becomes a uniform distribution on $[0, 1]$, regardless of the correct decisions. This implies that a level j agent has no ability to diagnose cases beyond a random guess.
- **Biased Knowledge:** When α_j^0 is very close to α_j^1 and β_j^0 is very close to β_j^1, the assessments have essentially the same distribution regardless of the correct decision. Hence, in this case,

[5] Using our model in real-world settings requires estimation of the model parameters p_ℓ^1 and the beta distribution coefficients. These can be estimated using historical data or expert judgments (through test cases). For instance, once we have evaluated the correct decisions Y for a sample of cases, we can estimate p_ℓ^1, and estimate the density function $f_j^y(x)$. A more detailed explanation of this procedure is given in Online Appendix C.

[6] We assume these densities are tier-dependent but are the same among the agents within each tier (e.g., among lower level agents). This assumption is made for tractability, and also represents the fact that in practice knowledge levels vary significantly across tiers (e.g., between nurses and physicians) but only moderately within each tier (e.g., among nurses). However, extending our model and analyses to cases where these densities are different even within each tier is relatively straightforward.

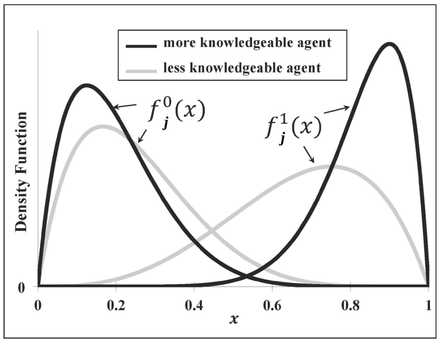

Figure 2: Examples of beta distributions for modeling agents' assessments.

the agent is providing stochastically the same assessment for both cases with $Y = 1$ and $Y = 0$ regardless of the signals he/she receives from cases. This enables us to model the cognitive biases of the decision makers (see, e.g., Hogarth (1980) pp. 166-170 for a complete list of such biases). For instance, the agent might consistently be biased towards approving (a positive bias) or rejecting (a negative bias) the cases (or in the triage example, towards assigning to a specific urgency/ESI level).

- **General Knowledge:** Adjusting parameters α_j^y and β_j^y alters the shape of the density function and hence the knowledge structure of an agent. For example, we can choose α_j^y and β_j^y for two agents such that they have the same consistency, but one agent has higher accuracy for cases with $Y = y$. To do this, we reduce α_j^0 and increase β_j^0 (increase α_j^1 and reduce β_j^1) so that $\mathbb{E}[X_j^0]$ decreases ($\mathbb{E}[X_j^1]$ increases) while $Var(X_j^y)$ remains constant. Alternatively, we can increase both α_j^y and β_j^y proportionally, so that $\mathbb{E}[X_j^y]$ remains unchanged but $Var(X_j^y)$ is decreased, to model a second agent that has the same accuracy, but better consistency.

3.2 Modeling Assessment Sharing

Our framework also allows us to represent settings in which lower level agents share their assessments with their upper level agent, and settings in which they do not. For instance, in TPT, the lower level agents (triage nurses) transfer patients to the upper level agent (TP) who examines them and makes a decision based solely on his/her own assessment. In some other systems (e.g., bank loan processing) the upper level agent uses the assessment of the lower level agent as part of his/her decision process. We study both scenarios in order to shed light on the *value of sharing assessments* in HKBSSs. To this end, we consider the following scenarios.

- **Independent Assessments (IA).** Under this scenario, the upper level agent makes an independent assessment of the case referred to him/her. However, based on our motivating examples, we assume the upper level agent knows lower agent's referral policy (but not the lower level's assessment), and note that in this case p_u^1 (and hence, $p_u^0 = 1 - p_u^1$) is a function of the lower level's referral policy. For instance, if the policy of the lower level agent is to refer the cases to the upper level only when his/her assessment is in set $\mathcal{A}$ (for some set $\mathcal{A} \subset [0, 1]$)

then using the Bayes' rule:

$$p_u^1(\mathcal{A}) = Pr\{Y=1|X_\ell \in \mathcal{A}\} = \frac{\big(\int_{\mathcal{A}} f_\ell^1(x_\ell)dx_\ell\big)p_\ell^1}{\big(\int_{\mathcal{A}} f_\ell^1(x_\ell)dx_\ell\big)p_\ell^1 + \big(\int_{\mathcal{A}} f_\ell^0(x_\ell)dx_\ell\big)(1-p_\ell^1)}. \tag{3}$$

- **Shared Assessment (SA):** Under this scenario, the assessment made at the upper level is a function of both the upper and the lower level agents' assessments. Specifically, using the Bayes' rule:

$$p_u^1(x_\ell) = Pr\{Y=1|X_\ell = x_\ell\} = \frac{f_\ell^1(x_\ell)p_\ell^1}{f_\ell^1(x_\ell)p_\ell^1 + f_\ell^0(x_\ell)(1-p_\ell^1)}. \tag{4}$$

Thus, under the SA (IA) structure, the density function of the upper level agent's assessment is not only a function of the upper level agent's assessments, but also the lower level's assessment (referral policy). For instance, under the SA structure:

$$f_u(x_u, x_\ell) = p_u^1(x_\ell)f_u^1(x_u) + (1-p_u^1(x_\ell))f_u^0(x_u).^7$$

Finally, we note that from an assessment sharing perspective, the SA scenario can be viewed as a special case of IA: in IA, the upper level agent only knows that $x_\ell \in \mathcal{A}$ (for cases referred to him/her), while in SA, he/she knows the exact value of x_ℓ. If $\mathcal{A} = (0,1)$ then no information is shared, and if $\mathcal{A} = \{x_l\}$ the exact assessment is shared. However, the upper level agent can consider a middle case A to take advantage of the filtering made at the lower level, when sharing the exact assessment is not possible. Ideally, such filtering should take into account the queue length information (for all queues) at the time the case was referred to the upper level. However, the information technology infrastructure to allow reporting of the queue length information for each case routed to the upper level would also allow sharing of the full lower assessments. Since the latter makes the former superfluous, we do not consider the scenario in which queue lengths are shared without sharing assessments. Hence, for the IA scenario, we assume A is either $(0,1)$ or a fixed strict subset of it that is calculated for a given queue length (e.g., the average queue length).

3.3 Modeling Rationality

In order for our model to represent reality, we must ensure that the agents' assessments are "rational" in the sense that their assessments, x_j, are positively (or at least not negatively) correlated with the correct type of the case, y. We believe with some minimum training, agent assessments should possess this property. We formalize rationality in the following definition.

Definition 1 (Rational Assessments) *Agents' assessments are said to be rational if, and only if, $Pr\{Y=1|X_j = x\}$ is (weakly) increasing in x (for $x \in (0,1)$, $j \in \{\ell, u\}$).*

[7] For notational simplicity, however, we may suppress the dependency of f_u to x_ℓ (or set $\mathcal{A}$), and use $f_u(x)$ whenever it is clear from the context.

The following result connects the rational assessments to the stochastic *likelihood ratio ordering* (denoted by "$\leq_{lr}$").[8] All proofs are provided in the appendix.

Lemma 1 (Rationality and Likelihood Ordering) *In both assessment sharing scenarios (IA and SA), the agents' assessments are rational if, and only if,* $X_j^0 \leq_{lr} X_j^1$ $(j \in \{\ell, u\})$.

We note that since $\leq_{lr}$ is stronger than the usual stochastic ordering (which yields ordering in expectation), the above lemma also implies an ordering in agents' accuracy: when assessments are rational, we have $\mathbb{E}[X_j^0] \leq \mathbb{E}[X_j^1]$.

We define rational assessments in the context of the beta distribution model of agent knowledge as follows.

Definition 2 (Rationality Condition (RC)) *The Rationality Condition (RC) is said to hold if either (a)* $\beta_j^0 \geq \beta_j^1$ *and* $\alpha_j^0 < \alpha_j^1$*, or (b)* $\beta_j^0 > \beta_j^1$ *and* $\alpha_j^0 \leq \alpha_j^1$ $(j \in \{\ell, u\})$.

We note that the RC is not restrictive, and holds for a wide range of beta distributions. It can be viewed as a minimum knowledge requirement for agents working in a real-world setting.

Lemma 2 (Rationality) *In both assessment sharing scenarios (IA and SA), the agents' assessments are rational if, and only if, the RC holds.*

3.4 Optimal Knowledge-Based Decisions: A POMDP Model

In order to address the first of the four fundamental questions raised in the Introduction and characterize the structure of the optimal decisions, we now model the dynamics of the system. Because the correct decision for each patient cannot be observed directly, we model the dynamics of the system as a Partially Observable Markov Decision Process (POMDP), where in addition to the number of cases at the lower and upper level queues, for each case in the system, we keep track of the latest belief about the correct decision being $Y = 1$. This latest belief serves as a *sufficient statistic* for each case in the system, and is updated in the following manner. Each case arrives at a lower level agent with a prior probability of p_l^1. Once assessed by a lower level agent, this probability is updated to $p_u^1 \triangleq T_1\left(p_l^1, x_l\right)$, where x_l is the lower level agent's assessment (a noisy observation/signal), and $T_1(\cdot)$ is a Bayesian updating operator. If the case is referred to the upper level agent and is assessed by him/her, the probability is further updated to $\tilde{p}_u^1 \triangleq T_2^{\nu}\left(p_u^1, x_u\right)$, where $\nu \in \{IA, SA\}$ represents the assessment sharing scenario. In particular, based on Section 3.2, when

[8] If f and g represent the densities or probability mass functions of two random variables X and Y, respectively, and $f(\xi)/g(\xi)$ is increasing in ξ over the union of the supports of X and Y, then X is said to be greater than Y in the *likelihood ratio ordering* ($Y \leq_{lr} X$).

$\nu = SA$, we assume the Bayesian operator T_2^{ν} utilizes the exact value of x_l, and when $\nu = SA$, we assume it only utilizes the fact that $x_l \in \mathcal{A}$ (even if it takes x_l as an input).

Using the well-known uniformization technique, we can transfer the underlying continuous-time Markov chain to a discrete-time equivalent one. We can also re-scale time (without loss of generality), and assume that the event rate is $\Lambda \triangleq K(\lambda_l + \mu_l) + \mu_u = 1$. In this transformed system, we define the system state to be $(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u)$, where $\mathbf{n}_\ell = (n_k : k \in \mathcal{K})$ is a K-dimensional vector denoting the number of cases in the lower level agents' queues, and n_u denotes the number of cases in the upper level's agent queue. Also, $\mathbf{p}_\ell$ denotes a $K \times b_l$ matrix with its i, j element being the prior probability of a case at the jth position[9] in the queue of the lower level agent i ($i \in \mathcal{K}, j \in \{1, 2, \cdots, b_l\}$). Thus, each element of $\mathbf{p}_\ell$ is equal to $p_l^1 \in (0, 1)$ unless the corresponding position in the queue is empty, in which case we assign a zero to that element (without loss of generality). Similarly, $\mathbf{p}_u$ is a vector with its jth element denoting the belief that a case in the jth position of the upper level agent's queue has a correct decision $Y = 1$.

To define the optimality equation, we let $\mathbb{Z}_{+,b} \triangleq \{0, 1, \cdots, b\}$, denote the state space by $\mathcal{S} \triangleq \mathbb{Z}_{+,b_\ell}^K \times \mathbb{Z}_{+,b_u} \times [0,1]^{Kb_\ell + b_u}$, and represent by $\mathcal{F}$ the set of all real-valued functions defined on $\mathcal{S}$. We then consider the functional operators $T_{A,k}$, $T_{S,k}$, and $T_{S,u}$ (all from $\mathcal{F}$ to $\mathcal{F}$) corresponding to an arrival event at lower level agent $k \in \mathcal{K}$, service completion at that agent, and service completion at the upper level, respectively. For any function $J \in \mathcal{F}$, we then define the functional operator $\tilde{T}_* : \mathcal{F} \to \mathcal{F}$ as

$$\tilde{T}_* J \triangleq \sum_{k \in \mathcal{K}} \big(\lambda_l T_{A,k} J + \mu_l T_{S,k} J\big) + \mu_u T_{S,u} J. \tag{5}$$

With these, the optimal long-run average cost optimality equation for any state $(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u) \in \mathcal{S}$ can be written (in the functional form) as

$$\varphi^* + J(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u) = T_* J(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u), \tag{6}$$

where $T_* J(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u) \triangleq \mathbf{h}\big(\mathbf{n}_l (\mathbf{1}_K)^T, n_u\big)^T + \tilde{T}_* J(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u)$, and φ^* is the optimal long-run average cost. In (5), the operators $T_{A,k}$, $T_{S,k}$, and $T_{S,u}$ are defined as follows.

$$T_{A,k} J(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u) \quad \triangleq \quad \mathbb{1}_{\{n_k = b_\ell\}} c_{b,\ell} + \mathbb{1}_{\{n_k \neq b_\ell\}} J(\mathbf{n}_\ell + \mathbf{e}_k, n_u, \psi_{k, n_k+1}(\mathbf{p}_\ell, p_\ell^1), \mathbf{p}_u), \tag{7}$$

where $\mathbb{1}$ is the indicator function, $\mathbf{e}_k$ is K-dimensional vector with a one as the kth element and zeros elsewhere, and $\psi_{i,j}(\mathbf{A}, \xi)$ is a matrix operator that changes the i, j element of matrix $\mathbf{A}$ to ξ. In addition,

$$\begin{aligned} T_{S,k} J(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u) \triangleq\ & \mathbb{1}_{\{n_k = 0\}} J(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u) \\ & + \mathbb{1}_{\{n_k \neq 0\}} \int_0^1 f_\ell(x_\ell) \min_{i \in \{0,1,2\}} \Big\{ \phi_i(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u, x_\ell) \Big\} dx_\ell, \end{aligned} \tag{8}$$

[9] First position is the patient under service, second is the the first one in line, etc.

where

$$\begin{aligned} \phi_0(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u, x_\ell) &\triangleq J(\mathbf{n}_\ell - \mathbf{e}_k, n_u, \psi_{k,n_k}(\mathbf{p}_\ell, 0), \mathbf{p}_u) + c_1 T_1(p_\ell^1, x_\ell), \quad (9) \\ \phi_1(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u, x_\ell) &\triangleq J(\mathbf{n}_\ell - \mathbf{e}_k, n_u, \psi_{k,n_k}(\mathbf{p}_\ell, 0), \mathbf{p}_u) + c_0(1 - T_1(p_\ell^1, x_\ell)), \quad (10) \end{aligned}$$

and

$$\begin{aligned} \phi_2(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u, x_\ell) &\triangleq \mathbb{1}_{\{n_u = b_u\}} \big(c_{b,u} + J(\mathbf{n}_\ell - \mathbf{e}_k, n_u, \psi_{k,n_k}(\mathbf{p}_\ell, 0), \mathbf{p}_u) \big) \\ &+ \mathbb{1}_{\{n_u \neq b_u\}} J(\mathbf{n}_\ell - \mathbf{e}_k, n_u + 1, \psi_{k,n_k}(\mathbf{p}_\ell, 0), \psi_{1,n_u+1}(\mathbf{p}_u, T_1(p_\ell^1, x_\ell))) \quad (11) \end{aligned}$$

represent the cost-to-go under decisions $a = 0$, $a = 1$, and $a = UP$ by the lower level agent $k \in \mathcal{K}$ (after making assessment x_l), respectively. Finally,

$$\begin{aligned} T_{S,u} J(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u) \triangleq\ & \mathbb{1}_{\{n_u = 0\}} J(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u) \\ &+ \mathbb{1}_{\{n_u \neq 0\}} \int_0^1 f_u(x_u) \min_{i \in \{0,1\}} \Big\{ \tilde{\phi}_i(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u, x_u) \Big\} dx_u, \quad (12) \end{aligned}$$

where $\tilde{\phi}_0(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u, x_u) \triangleq J(\mathbf{n}_\ell, n_u - 1, \mathbf{p}_\ell, \psi^-(\mathbf{p}_u)) + c_1 T_2^\nu(\mathbf{p}_u \tilde{\mathbf{e}}_\mathbf{1}^T, x_u)$ and $\tilde{\phi}_1(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u, x_u) \triangleq J(\mathbf{n}_\ell, n_u - 1, \mathbf{p}_\ell, \psi^-(\mathbf{p}_u)) + c_0(1 - T_2^\nu(\mathbf{p}_u \tilde{\mathbf{e}}_\mathbf{1}^T, x_u))$ denote the cost-to-go under decisions $a = 0$ and $a = 1$ by the upper level agent under the assessment sharing scenario $\nu \in \{IA, SA\}$, respectively. In these definitions, $\psi^-(\cdot)$ is a vector operator that takes a vector, deletes its first element, and shifts all other elements one position to the left. Furthermore, $\tilde{\mathbf{e}}_\mathbf{1}$ is a b_u-dimensional vector with a one as its first element and zeros elsewhere.

3.5 Structure of Optimal Decisions

We start by analyzing the optimal decisions at the upper level. We define the critical fractile value

$$x_u^*(\mathbf{p}_u \tilde{\mathbf{e}}_\mathbf{1}^T) \triangleq (T_2^\nu)^{-1}(\mathbf{p}_u \tilde{\mathbf{e}}_\mathbf{1}^T, \frac{c_0}{c_0 + c_1}), \quad (13)$$

where for all $y \in [0, 1]$

$$(T_2^\nu)^{-1}(\mathbf{p}_u \tilde{\mathbf{e}}_\mathbf{1}^T, y) \triangleq \inf \Big\{ x \in [0, 1] : T_2^\nu(\mathbf{p}_u \tilde{\mathbf{e}}_\mathbf{1}^T, x) \geq y \Big\}. \quad (14)$$

We note that this critical fractile is similar to that of a *newsvendor* problem with underage cost c_0, overage cost c_1, and a demand distribution T_2^ν that depends on $\mathbf{p}_u \tilde{\mathbf{e}}_\mathbf{1}^T$. In the following result, we show that the minimization in (12) is fully characterized by the critical fractile $x_u^*(\mathbf{p}_u \tilde{\mathbf{e}}_\mathbf{1}^T)$. This, proves that the upper level optimal policy is a *control-limit* policy defined by $x_u^*(\mathbf{p}_u \tilde{\mathbf{e}}_\mathbf{1}^T)$ for both $\nu = IA, SA$.[10]

[10] When $\nu = IA$ the critical fractile $x_u^*(\mathbf{p}_u \tilde{\mathbf{e}}_\mathbf{1}^T)$ is constant in $\mathbf{p}_u \tilde{\mathbf{e}}_\mathbf{1}^T$ since the Bayesian operator T_2^ν does not use the value of $\mathbf{p}_u \tilde{\mathbf{e}}_\mathbf{1}^T$. Nevertheless, for generality, we write it as $x_u^*(\mathbf{p}_u \tilde{\mathbf{e}}_\mathbf{1}^T)$.

Proposition 1 (Control-Limit Policy: Upper Level) *Under the RC,*

$$\min_{i\in\{0,1\}}\left\{\tilde{\phi}_i(\cdot)\right\} = \mathbb{1}_{\{x_u \le x_u^*(\mathbf{p}_u\tilde{\mathbf{e}}_\mathbf{1}^T)\}}\tilde{\phi}_0(\cdot) + \mathbb{1}_{\{x_u > x_u^*(\mathbf{p}_u\tilde{\mathbf{e}}_\mathbf{1}^T)\}}\tilde{\phi}_1(\cdot). \tag{15}$$

Hence, for both $\nu = IA, SA$, the optimal decision at the upper level is a control-limit policy defined by the following convex policy regions: $a_u^*(x_u) = 0$ for all $x_u \in \left[0, x_u^*(\mathbf{p}_u\tilde{\mathbf{e}}_\mathbf{1}^T)\right]$ and $a_u^*(x_u) = 1$ for all $x_u \in \left(x_u^*(\mathbf{p}_u\tilde{\mathbf{e}}_\mathbf{1}^T), 1\right]$. Furthermore, when the monotonicity of $T_2^\nu(\mathbf{p}_u\tilde{\mathbf{e}}_\mathbf{1}^T, x_u)$ in x_u (established in Lemma 6 in Online Appendix A) is strict, $x_u^*(\cdot)$ is the unique solution to

$$\frac{f_u^0(x_u^*)}{f_u^1(x_u^*)} = \frac{p_u^1 c_1}{(1-p_u^1)c_0}, \tag{16}$$

for both $\nu = IA, SA$, where $p_u^1 = \mathbf{p}_u\tilde{\mathbf{e}}_\mathbf{1}^T$.

We now turn our attention to the optimal policy at the lower level. We start by establishing Proposition 2, which states that the cost of adding a case with updated belief $p_u^1 \triangleq T_1\left(p_l^1, x_l\right)$ to the upper level agent's queue is concave in p_u^1. The proof of this result is established via Lemmas 3 and 4, which demonstrate that (i) the functional operator T_* *preserves* this concavity property, and (ii) the function $J(\cdot)$ can be viewed as the limit (as the discount rate goes to zero) of a *relative cost difference*, where costs are calculated in an infinite-horizon discounted cost setting. Moreover, the proof of point (i) itself is based on the fact that the integration and minimization operators in (8) and (12) preserve concavity, which we show in Lemma 5.

Proposition 2 (Concavity in Updated Belief) *$J(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \psi_{1,n_u+1}(\mathbf{p}_u, p_u^1))$ is concave in $p_u^1 \in [0,1]$ for all states with $n_u < b_u$.*

Lemma 3 (Concavity Preservation) *Let $\mathcal{F}_c \subseteq \mathcal{F}$ be the set of all real-valued functions defined on $\mathcal{S}$ that satisfy the concavity property of Proposition 2, and denote by $J_n \in \mathcal{F}$ the optimal discounted cost function over n periods. If $J_n \in \mathcal{F}_c$ then $J_{n+1} = T_* J_n \in \mathcal{F}_c$.*

Lemma 4 (Limiting Behavior) *There exists a sequence of discount rates $\zeta_{i\in\mathbb{N}} \to 0$ such that*

$$J(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \psi_{1,n_u+1}(\mathbf{p}_u, p_u^1)) = \lim_{i\to\infty}\left[J_\infty^{\zeta_i}(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \psi_{1,n_u+1}(\mathbf{p}_u, p_u^1)) - J_\infty^{\zeta_i}(\mathbf{0}, 0, \mathbf{0}, \mathbf{0})\right], \tag{17}$$

where $J_\infty^{\zeta_i}(\cdot)$ is the infinite-horizon discounted cost when the discount rate is ζ_i.

Lemma 5 (Concavity Preservation Operation) *Let $g_1(y,x), g_2(y,x), \cdots, g_m(y,x)$ be m real-valued concave functions in y. Then the function*

$$g(y) \triangleq \mathbb{E}_{X\sim f(x)}\left[\min_{i\in\{1,2,\cdots,m\}} g_i(y, X)\right] = \int_x \left[\min_{i\in\{1,2,\cdots,m\}} g_i(y,x)\right] f(x)dx \tag{18}$$

is also concave in y.

The concavity property established by Proposition 2 significantly simplifies characterizing the optimal policy. We first note that, because of this result and the fact that functions $\phi_0(\cdot)$ and $\phi_1(\cdot)$ are affine in $p_u^1 \triangleq T_1\left(p_\ell^1, x_\ell\right)$, the minimization in (8) is pointwise minimum of three concave functions (which is itself concave). To characterize this minimization, similar to (13), we define the critical fractile value

$$x_\ell^*(p_\ell^1) \triangleq (T_1)^{-1}(p_\ell^1, \frac{c_0}{c_0 + c_1}), \tag{19}$$

where for all $y \in [0,1]$

$$(T_1)^{-1}(p_\ell^1, y) \triangleq \inf\Big\{x \in [0,1] : T_1(p_\ell^1, x) \geq y\Big\}. \tag{20}$$

The critical fractile $x_\ell^*(p_\ell^1)$ defined above is the lower level agent's assessment for which he/she would be indifferent between decisions $a = 0$ and $a = 1$, had he/she not have the option of referring the case to the upper level. With these, we can now establish the optimal policy of the lower level as a *double control-limit* policy, where only cases with assessments that fall between the two control limits[11] enter the queue of the upper level. The main intuition behind the proof of this result is shown in Figure 3, which depicts the behavior of functions $\phi_i(\cdot)$ for $i \in \{0,1,2\}$ in terms of the lower level's updated belief $p_u^1 \triangleq T_1\left(p_\ell^1, x_\ell\right)$. Due to this specific structure, it is first shown that all the policy regions must be *convex sets*, which in turn results in the double control-limit structure.

Proposition 3 (Double Control-Limit Policy: Lower Level) *Under the RC, there exist* $\underline{x}_\ell^*(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u)$ *and* $\overline{x}_\ell^*(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u)$, *where* $0 \leq \underline{x}_\ell^*(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u) \leq x_\ell^*(p_\ell^1) \leq \overline{x}_\ell^*(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u) \leq 1$, *such that*

$$\min_{i \in \{0,1,2\}} \Big\{\phi_i(\cdot)\Big\} = \mathbb{1}_{\{x_\ell \leq \underline{x}_\ell^*(\cdot)\}}\phi_0(\cdot) + \mathbb{1}_{\{x_\ell \geq \overline{x}_\ell^*(\cdot)\}}\phi_1(\cdot) + \mathbb{1}_{\{\underline{x}_\ell^*(\cdot) < x_\ell < \overline{x}_\ell^*(\cdot)\}}\phi_2(\cdot). \tag{21}$$

Hence, for both assessment sharing scenarios $\nu = IA, SA$, *the optimal decision at the lower level is a double control-limit policy defined by the following convex policy regions:* $a_u^*(x_u) = 0$ *for all* $x_\ell \in \big[0, \underline{x}_\ell^*(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u)\big]$, $a_\ell^*(x_u) = 1$ *for all* $x_\ell \in \big[\overline{x}_\ell^*(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u), 1\big]$, *and* $a_\ell^*(x_u) = UP$ *otherwise.*

3.6 Sensitivity of Optimal Policy

In this section, we delve into the second question posed in the Introduction: How do environmental factors affect the optimal policy? This is a vital question from a managerial standpoint because it raises the issue of what characteristics must be considered when designing an effective system.

[11] These cases can be thought of as more complex ones for which the agent has a less clear judgement.

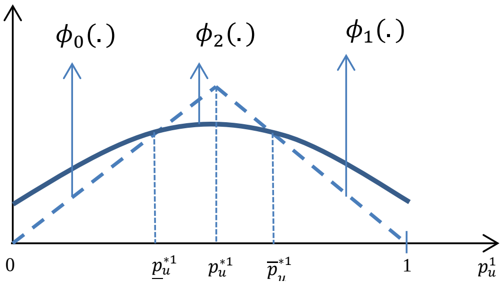

Figure 3: The structure of decision-making at the lower level as a function of the updated belief $p_u^1 \triangleq T_1\left(p_\ell^1, x_\ell\right)$ $[p_u^{*1} \triangleq \frac{c_0}{c_0+c_1}]$.

By understanding how various factors affect the optimal policy, we can also gain insights into how different HKBSSs should be managed differently. Here we explore the impact of: (i) decision error costs (c_0, c_1), (ii) the population mix entering the system (p_ℓ^1), and (iii) system workloads ($\mathbf{n}_\ell$, n_u, λ_l).

3.6.1 The Effect of Decision Error Costs

The source of decision error costs c_0 and c_1 differ among HKBSSs. In the TPT system, these costs are influenced by downstream conditions—notably, the number and type of patients waiting in the waiting area of the ED to be treated following the triage stage. If the waiting area is empty, then c_1 is close to zero, because the patient will move immediately into treatment regardless of whether or not he/she is classified correctly as urgent. But if the waiting area is full of non-urgent patients, then c_1 is high because misclassifying an urgent patient will result in a long, and potentially dangerous, wait. Similarly, c_0 is close to zero if the waiting area is empty, because no one will be forced to wait longer due to misclassifying a non-urgent patient as urgent. However, if the waiting area has many non-urgent patients, then each of them will incur a delay cost by jumping the misclassified patient ahead of them. Worse, an urgent patient who arrives after the misclassified non-urgent patient has entered treatment may have to wait. So c_0 increases in both the number of non-urgent patients in the waiting area and the arrival rate of urgent patients. These observations imply that c_0 and c_1 will vary in both magnitude and ratio over the course of a day for exogenous reasons.

In contrast, in a visa processing system, c_0 and c_1 are typically independent of the downstream conditions, and are relatively stable over long periods of time. The cost of granting someone a visa who should not receive one, c_0, might change as attitudes toward terrorism or international students change. But such changes would be on a time scale of months or years rather than hours.

The following result describes the effect of decision error costs on the control-limit structure established in Section 3.5.

Proposition 4 (The Effect of Decision Error Costs) $\underline{x}_\ell^*(\cdot)$ *is non-increasing in* c_1, $\overline{x}_\ell^*(\cdot)$ *is* *non-decreasing in c_0, and $x_u^*(\cdot)$ is non-decreasing (non-increasing) in c_0 (c_1) for both assessment sharing scenarios $\nu = IA, SA$. Furthermore, (i) $\overline{x}_\ell^*(\cdot) - \underline{x}_\ell^*(\cdot) \to 0^+$ as $c_0, c_1 \to 0^+$, and (ii) $\overline{x}_\ell^*(\cdot) - \underline{x}_\ell^*(\cdot) \to 1^-$ as $c_0, c_1 \to \infty$.*

The implication of Proposition 4 for the TPT system is as follows. When the waiting area of the ED is relatively empty, decision error costs c_0 and c_1 are both negligible, so Proposition 4 suggests that cases should not be routed to the TP. Thus, utilizing a TP during such hours may not be economical. In contrast, when the waiting area is crowded and the ED expects a high near-term arrival rate of urgent patients, c_0 and c_1 are both high, and Proposition 4 suggests that most cases should be routed to the TP. This may require the ED to hire additional TPs or implement other capacity management mechanisms to avoid overloading the TP. Of note, when the waiting area is crowded but the ED does not expect a high near-term arrival rate of urgent patients, c_1 might be high while c_0 might be in a middle range, because c_0 is less sensitive (compared to c_1) to the current situation of the waiting area, and is more sensitive to the delay of future arrivals who are truly urgent. Proposition 4 suggests that in such circumstances $\underline{x}_\ell^*(\cdot)$ will be close to zero but $\overline{x}_\ell^*(\cdot)$ may be in a middle range, implying that not all cases should be routed to the TP. This can limit the need for hiring additional TPs or implementing other capacity management mechanisms.

3.6.2 The Effect of Population Mix

The mix of cases served by an HKBSS may depend on many factors including geographical location, type of service provided, and various economic factors. In the TPT system, for instance, the ratio of urgent to non-urgent patients varies among EDs, with Level 1 trauma centers on one side of the spectrum and small community level EDs on the other. The following result provides some insights into the impact of case mix on the optimal control policy.

Proposition 5 (The Effect of Population Mix) *The control-limits $\underline{x}_\ell^*(\cdot)$, $\overline{x}_\ell^*(\cdot)$, and $x_u^*(\cdot)$ are all non-increasing in p_ℓ^1 for both assessment sharing scenarios $\nu = IA, SA$.*

Proposition 5 establishes that, regardless of the assessment sharing scenario, an HKBSS that sees a higher percentage of Y=1 type cases should impose lower control-limits for making a 0 or 1 decision at both the lower and upper levels. This means that an increase in the percentage of Y=1 type cases translates to a decrease (increase) in the size of the region for which $a_j^*(x) = 0$ ($a_j^*(x) = 1$). Interestingly, however, the effect of an increase in percentage of Y=1 type cases on the size of the region for which the cases are routed to the upper level (i.e., the region for which $a_\ell^*(x) = UP$) is not necessarily monotone. For the TPT system, it means that a level 1 trauma center may or may not need to route more cases to the TP compared to a community hospital ED.

Moreover, Proposition 5 implies differences in how patients could be classified in the two EDs. For example, it suggests that a TP that serves EDs in both a level 1 trauma center and a community hospital should assess some patients in the level 1 trauma center as urgent but assess identical patients as non-urgent in the community hospital. This is in sharp contrast with the prevailing belief that triage classification should depend only on the medical conditions of a patient. Because triage classifications are used for prioritization, they need to be viewed as relative rather than absolute ratings, which implies that environmental conditions also matter.[12]

3.6.3 The Effect of Workload

Workload obviously impacts the performance of an HKBSS. Understanding this impact and how it can be managed by using information about workload can help managers respond efficiently to fluctuations (e.g., spikes) in workload. Clearly, if the workload increase is large enough, more capacity (e.g., triage nurses) will be needed. But, since capacity is expensive, it might be essential to look for more cost-effective alternatives for helping the system to cope with workload spikes. In this section, we investigate how the optimal control-limit policy reacts to changes in the workload.[13] In Section 4, we will use the resulting insights to design effective heuristic workload unbalancing policies that alter the criteria for directing/referring cases from the lower level to the upper level in response to shifts in workload levels.

We start by presenting the following result.

Proposition 6 (Inverted V-Shape: Queue Lengths) *The control-limits $\underline{x}^*_\ell(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u)$ and $\overline{x}^*_\ell(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u)$ are non-increasing and non-decreasing in n_k ($\forall k \in \mathcal{K}:\ n_k < b_l$) for both assessment sharing scenarios $\nu = IA, SA$, respectively. Therefore, the optimal policy has the following property at the lower level in both SA and IA scenarios:*

$$a^*_\ell(x) = 1\ (0)\ \textit{at queue length}\ \mathbf{n}_\ell \implies a^*_\ell(x) = 1\ (0)\ \textit{at queue length}\ \mathbf{n}_\ell + \mathbf{e}_k\ (\forall k \in \mathcal{K}:\ n_k < b_l).$$

Proposition 6 implies that the optimal decision rule for any lower level agent follows an inverted V-shape structure in his/her queue length, a typical pattern of which is shown in Figure 4 (right). A similar but rather intuitive fact—that the "UP" region shrinks as the upper level queue length

[12] Other environmental factors that might vary across hospitals are the decision error costs, which may in turn be affected by the case mix. For instance, as mentioned in Section 3.6.1, if the waiting area is full of non-urgent patients (which is more likely to occur in a community hospital ED than in a level 1 trauma center, assuming that they operate at an equal level of congestion), then c_1 is high. This in turn will affect the optimal threshold levels (in the direction implied by Proportion 4). Thus, differences in decision error costs caused by the difference in case mix will further differentiate optimal decisions in level 1 trauma centres from those in community hospital EDs.

[13] As noted in Section 3.6.1, the decision error costs may also vary based on the system's downstream congestion. To provide clear insights, we focus here on the role of workload by keeping all else (e.g., the decision error costs) equal.

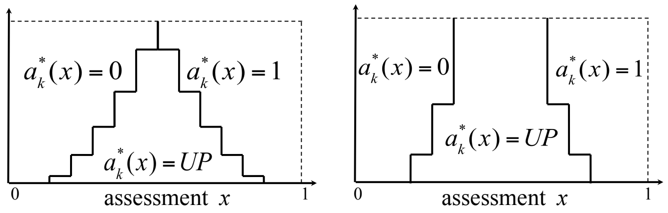

Figure 4: Typical inverted V-shape structure of a *lower level agent* optimal decision, where x represents the lower level agent's assessment. *Left:* in (x, n_u) space; *Right:* in (x, n_k) space $(k = 1, 2, \ldots, K)$.

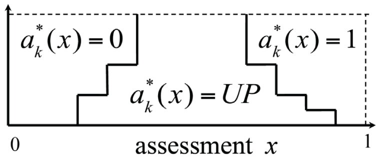

Figure 5: Typical Inverted V-shape structure of a *lower level agent* optimal decision in (x, λ_ℓ) space, where x represents the lower level agent's assessment.

increases—is shown in Figure 4 (left). Similarly, corresponding to an increase in the arrival rate, we have the following property at the lower level:

Proposition 7 (Inverted V-Shape: Arrival Rate) *The control-limits $\underline{x}_\ell^*(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u)$ and $\overline{x}_\ell^*(\mathbf{n}_\ell, n_u, \mathbf{p}_\ell, \mathbf{p}_u)$ are non-increasing and non-decreasing in λ_l for both assessment sharing scenarios $\nu = IA, SA$, respectively. Therefore, the optimal policy has the following property at the lower level in both SA and IA scenarios:*

$$a_\ell^*(x) = 1\ (0)\ \textit{at arrival rate}\ \lambda_\ell \quad \Longrightarrow \quad a_\ell^*(x) = 1\ (0)\ \textit{at arrival rate}\ \lambda_\ell + \varepsilon,$$

for any $\varepsilon > 0$.

Proposition 7 implies that the optimal decision rule for any lower level agent also follows an inverted V-shape structure in the arrival rate, a typical pattern of which is shown in Figure 5.

Returning to the workload rebalancing policies, Propositions 6 and 7 suggest that, to be effective, any decision under an optimal workload rebalancing policy that uses queue length but not arrival rate information (arrival information but not queue lengths) must mimic to the extent possible the property described by Proposition 6 (7). Moreover, the decisions under an optimal policy (which uses both queue length and arrival rate information) are described by both Propositions 6 and 7, a typical pattern of which is shown in Figure 6.

3.7 Levers for Improving Performance

To address the third question raised in the Introduction, about which levers are effective for improving system performance, we examine two options that address the knowledge-based decision

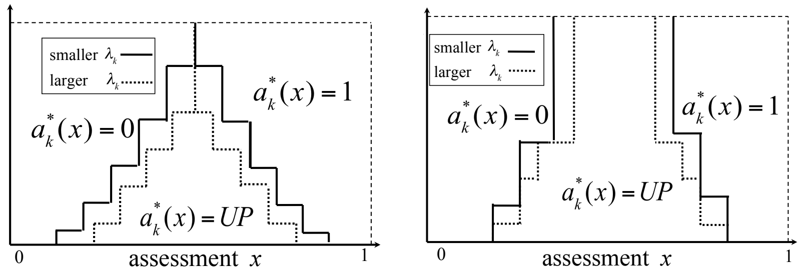

Figure 6: Typical inverted V-shape structure of a lower level agent optimal decision, where x represents the lower level agent's assessment. *Left:* in (x, n_u, λ_ℓ) space; *Right:* in (x, n_k, λ_ℓ) space.

making that is central to any HKBSS. These are (i) agent training, and (ii) information sharing between levels.

3.7.1 Agent Training

Agent training can improve performance by increasing agent knowledge (accuracy and/or consistency) and/or reducing cognitive biases. We find that, regardless of the assessment sharing structure, the impact of a one-sided training program (which helps agents to make better assessments about either "yes" or "no" cases but not both) relative to a two-sided program (which helps agents to make better assessments about both type of cases) depends on whether the implemented training program targets improving an agent's consistency or accuracy. Figure 7 illustrates this for an upper level agent by depicting his/her optimal control-limits and percentage reduction in error cost under two training strategies: Strategy 1 (highly effective one-sided training) and Strategy 2 (moderately effective two-sided training)[14]. In Figure 7 (left column) the training is assumed to improve the agent's consistency (variance) while keeping the accuracy (mean) constant. It does so by assuming $\alpha_u^0 = f(\xi_0)$, $\beta_u^0 = 2f(\xi_0)$, $\alpha_u^1 = 3g(\xi_1)$, and $\beta_u^1 = g(\xi_1)$, where $f(\xi_0) = (2-9\xi_0)/27\xi_0$ and $g(\xi_1) = (3-16\xi_1)/64\xi_1$. In this setting, $Var(X_u^0) = \xi_0$ and $Var(X_u^1) = \xi_1$ while the accuracy levels are constant: $\mathbb{E}[X_u^0] = 1/3$ and $\mathbb{E}[X_u^1] = 3/4$. We further assume $\xi_0, \xi_1 \in (0, 3/80]$ so that variances are positive and various suitable conditions (RC, boundedness, etc.) hold. To capture the effect of Strategy 1, we fix $\xi_0 = 3/80$ and let $\xi_1 = (3-\gamma)/80$, where $\gamma \in \{0, 1, 2\}$ represents the agent's *training level.* Under Strategy 2, we let $\xi_0 = \xi_1 = (3-\gamma/2)/80$. Thus, in Figure 7 (left column), Strategy 1 represents a highly effective one-sided consistency improvement, and Strategy 2 represents a moderately effective two-sided consistency improvement.

In Figure 7 (right), the effect of training is captured by improving the agent's accuracy while keeping the consistency constant. We do this by finding the appropriate values of α and β pa-

[14] Note that keeping the effectiveness (i.e., the magnitudes of improvement in assessments' accuracy and/or consistency) the same, a two-sided training program is clearly better than a one-sided one. Thus, it is sufficient to consider these two strategies.

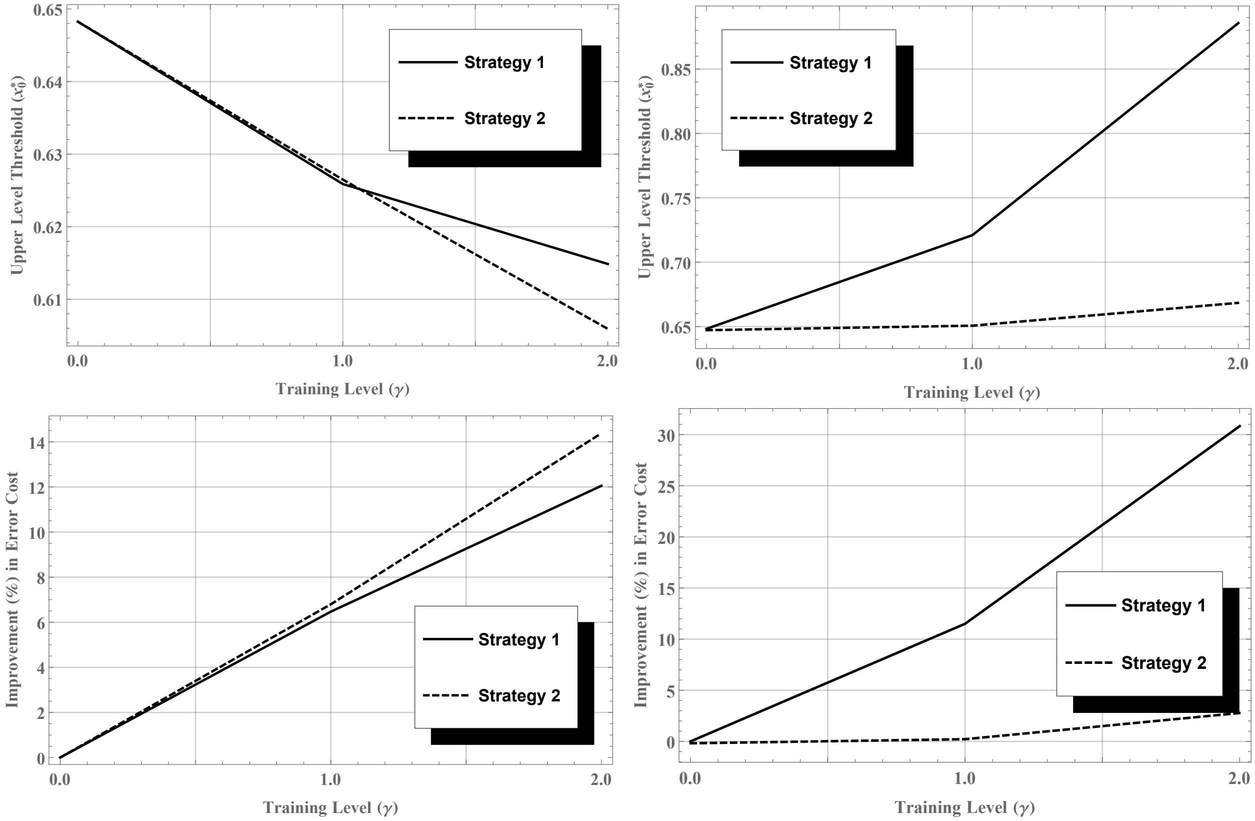

Figure 7: Effect of training programs on optimal decision thresholds (top row) and cost improvements (bottom row). *Left column:* consistency training; *Right column:* accuracy training. [Strategy 1: highly effective one-sided training; Strategy 2: moderately effective two-sided training] [Main parameters: $c_0 = 2$, $c_1 = 1$, $p_u^1 = 0.5$].

rameters such that the variances are fixed at the base level of 3/80 while means are improved through training. In particular, under Strategy 1, we set $\alpha_u^0 = f(\xi_0)$ and $\beta_u^0 = 2f(\xi_0)$ for a fixed $\xi_0 = 3/80$, and find α_u^1 and β_u^1 such that $Var(X_u^1)$ is fixed at 3/80 but the mean (for cases with y=1) varies according to $\mathbb{E}[X_u^1] = 3/4 + 0.1\gamma$, where $\gamma \in \{0, 1, 2\}$ represents the agent's training level. Under Strategy 2, we find α and β parameters such that $Var(X_u^0) = Var(X_u^1) = 3/80$ while $\mathbb{E}[X_u^0] = 1/3 - 0.1\gamma/2$ and $\mathbb{E}[X_u^1] = 3/4 + 0.1\gamma/2$ ($\gamma \in \{0, 1, 2\}$). Thus, in Figure 7 (right column), Strategy 1 represents a substantial one-sided accuracy improvement, and Strategy 2 represents a moderate two-sided accuracy improvement. We can summarize our insights about agent training (from this and many similar experiments we performed) as follows:

Observation 1 (Agent Training) *The impact of training programs depends on whether the training program targets improving an agent's consistency or accuracy. When training targets consistency, a moderately effective two-sided strategy typically has a stronger impact than a highly effective one-sided strategy, but for training programs that target an agent's accuracy, the result is reversed.*

In practice, training programs may affect both accuracy and consistency of assessments. Nevertheless, Observation 1 can guide decision makers in deciding whether they should focus more on (a) accuracy or consistency, and (b) two-sided or one-sided training strategies. For instance, in the TPT setting, triage nurses can be trained via presenting them with a balanced mix of urgent and non-urgent patients, or with a mix that emphasizes one of these two types. Furthermore, training techniques may put their main emphasis on nurses' accuracy or on their consistency. Improving consistency typically requires presenting nurses with a wide range of cases that cover the whole spectrum of patients who truly fall in one category (urgent or non-urgent), while improving accuracy requires considering the most likely assessments of a nurse and shift it towards 0 (1) for patients who are truly non-urgent (urgent). Observation 1 suggests that training nurses to be more consistent in their assessments requires improving consistency for both urgent and non-urgent patients. However, training nurses to be more accurate in their assessments can achieve a substantial improvement in performance even if the accuracy applies only to urgent or to non-urgent patients.

3.7.2 Assessment Sharing

Passing the lower level assessment to the upper level can improve the performance of an HKBSS by improving upper level decisions. For example, triage assessments made by nurses can be communicated to the telemedical physician via electronic forms. While it is not surprising that additional information improves decision-making, the magnitude of the benefit and the conditions under which sharing is most valuable are not obvious. For example, consider two scenarios: one in which the knowledge gap between the upper level and lower level agents is primarily caused by higher *accuracy* of the upper level agent (labeled HA gap), and one in which this gap is caused primarily by higher *consistency* of the upper level agent (labeled HC gap). Knowing which of these scenarios benefit more from assessment sharing is of potential value in managing and designing an HKBSS, since implementing assessment sharing will typically involve a cost.

To provide insights, we fix the knowledge of the lower level agent and vary the gap between consistency and/or accuracy of the upper and lower level. We do so by assuming $\alpha_\ell^0 = 1$, $\beta_\ell^0 = 2$, $\alpha_\ell^1 = 3$, and $\beta_\ell^0 = 1$ so that $\mathbb{E}[X_\ell^0] = 1/3$, $Var[X_\ell^0] = 1/18$, $\mathbb{E}[X_\ell^1] = 3/4$, and $Var[X_\ell^1] = 3/80$. We then consider a low gap $\eta^L = 5\%$ and a high gap $\eta^H = 20\%$, and find the upper level agent's beta distributions parameters separately for each scenarios so that (a) under the HA gap scenario: (i) $\mathbb{E}[X_u^0] = (1 - \eta^H)\,\mathbb{E}[X_\ell^0]$, (ii) $Var[X_u^0] = (1 - \eta^L)\,Var[X_\ell^0]$, (iii) $\mathbb{E}[X_u^1] = (1 + \eta^H)\,\mathbb{E}[X_\ell^1]$, and (iv) $Var[X_u^1] = (1 - \eta^L)\,Var[X_\ell^1]$, and (b) under the HC gap scenario: (i) $\mathbb{E}[X_u^0] = (1 - \eta^L)\,\mathbb{E}[X_\ell^0]$, (ii) $Var[X_u^0] = (1 - \eta^H)\,Var[X_\ell^0]$, (iii) $\mathbb{E}[X_u^1] = (1 + \eta^L)\,\mathbb{E}[X_\ell^1]$, and (iv) $Var[X_u^1] = (1 - \eta^H)\,Var[X_\ell^1]$. Next, we consider a case entering the system for which the lower agent makes an assessment $x_\ell \in \mathcal{A}$, and routes the case to the upper level. We assume $P_\ell = 0.5$ (in the base case), $\mathcal{A} = [0.4, 0.6]$, and assessment sharing follows the IA and SA settings introduced in Section 3.2.

To capture the impact of assessment sharing, we consider the cost improvement due to assessment sharing (i.e., cost difference in the IA and SA settings) under scenario $i \in \{HA, HC\}$, and denote it by $\Delta_i(SA||IA)$. Figure 8 compares the impact of assessment sharing under HA and HC gap scenarios. From this figure, we observe that the impact depends on the ratio of error costs c_0/c_1. In particular, when c_0/c_1 is low, assessment sharing is more impactful when it yields better decisions for cases that are of true type $Y = 1$. Hence, when upper level knowledge for such cases is not much better than that of the lower level, the upper level can significantly benefit from the lower level assessment. Interestingly, we observe that this occurs more under the HA scenario than under the HC scenario. That is, when c_0/c_1 is low, assessment sharing has a higher impact under the HA scenario than under the HC scenario. However, the result is flipped when c_0/c_1 is high. Based on Figure 8 and similar experiments we performed, we make the following:

Observation 2 (Assessment Sharing) *In HKBSSs in which c_0/c_1 is low (high), assessment sharing is most valuable when the knowledge gap between the upper level and lower level is mainly due to higher accuracy (consistency). Furthermore, the relative advantage of assessment sharing under HA and HC gap scenarios, $\Delta_{HA}(SA||IA)/\Delta_{HC}(SA||IA)$, increases as the lower level assessment increases.*

The implication of our results for managing an HKBSS is that implementing assessment sharing is particulary beneficial in systems in which (a) upper level knowledge is characterized by higher accuracy compared to the lower level, and (b) c_0/c_1 is low. For instance, as mentioned earlier, in the TPT system, triage nurses make judgements about whether the patient is urgent ($Y = 1$) or not ($Y = 0$). In such a setting, classifying an urgent patient as non-urgent is typically more costly than classifying a non-urgent patient as urgent, and hence, c_0/c_1 is low. Thus, our results suggest that assessment sharing is particulary valuable in TPT systems in which the knowledge gap between the telemedical physician and the nurses is due to better accuracy. The gain from assessment sharing in such systems is also higher for patients that have a higher nurse assessment (i.e., are considered more likely to be urgent).

4 Heuristic Workload Rebalancing Policies

Finally, we turn to the fourth question raised in the Introduction, which is whether simple methods can be used as practical policies for managing workload fluctuations in an HKBSS. The optimal policy characterized in Section 3.5 provides important insights, but it (i) requires full information

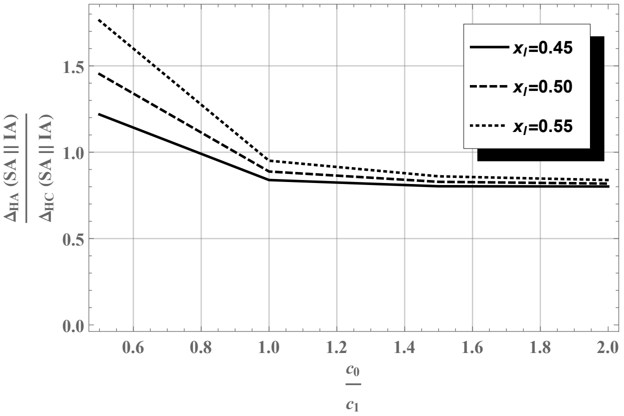

Figure 8: Effect of assessment sharing under HA and HC knowledge gap scenarios for various lower level assessments ($x_\ell \in \{0.45, 0.50, 0.55\}$).

about arrival rates, which are not always available in practice, and (ii) prescribes a different set of decision control-limits for each queue length, which may be impractical to implement. Furthermore, as can be seen from the results in Sections 3.5 and 3.6, the optimal policy depends on many parameters that may change over time. In this section, we appeal to the properties of the optimal policy to guide development of more practical alternatives.

4.1 Heuristics

We begin by noting that the queue length at the upper level has a stronger impact on the choice of the optimal control-limit (at the lower level), than do queue lengths at the lower level. That is, the optimal control-limits change more dramatically in n_u than in n_k ($k = 1, \dots, K$), as illustrated in Figure 4. This suggests that simple policies that effectively restrict the flow of cases to the upper level as n_u increases may perform well. Based on this intuition, we propose the following two simple and implementable heuristic policies for managing workload fluctuations in an HKBSS.

Green/Red Light ($\mathbb{GR}$). Under this policy, when the upper level queue length is less than some number, $\mathcal{N}_{GR}$, the system uses fixed (i.e., queue length independent) thresholds that are chosen so as to optimize performance under the normal arrival rate, $(\hat{\underline{x}}_\ell^*, \hat{\overline{x}}_\ell^*)$. When the upper level queue length is greater than or equal to $\mathcal{N}_{GR}$, lower level agents are prohibited from sending cases to the upper level (i.e., the "light" changes from green to red) and must make 0/1 decisions based on their own judgements (i.e., the thresholds collapse to $\hat{\underline{x}}_\ell^* = \hat{\overline{x}}_\ell^*$, so there is no "UP" region).

Switching ($\mathbb{SW}$). Under this policy, when the upper level queue length is less than some number, $\mathcal{N}_{SW}$, the system uses the optimal thresholds for the normal arrival rate, $(\hat{\underline{x}}_\ell^*, \hat{\overline{x}}_\ell^*)$. When the upper level queue length is greater than or equal to $\mathcal{N}_{SW}$, the system uses a second set of thresholds that are optimal for the maximum arrival rate.[15]

[15] Such a maximum arrival rate is typically determined by the system's manager. Due to the balking behavior

Note that the $\mathbb{GR}$ policy is similar to the $\mathbb{SW}$ policy except that $\mathbb{SW}$ uses spiked workload thresholds that satisfy $\hat{\underline{x}}^*_\ell \leq \hat{\overline{x}}^*_\ell$, but $\mathbb{GR}$ uses thresholds that satisfy $\hat{\underline{x}}^*_\ell = \hat{\overline{x}}^*_\ell$. This implies that there are fewer control variables in $\mathbb{GR}$ than in $\mathbb{SW}$, so it requires less search to find the best threshold. The reasons we consider both policies are that (i) the $\mathbb{GR}$ policy is intuitive and well-suited for practice, and (ii) it is useful to determine cases in which $\mathbb{GR}$ performs as well as $\mathbb{SW}$ despite having fewer control parameters.

In addition to computing referral thresholds, in order to implement the $\mathbb{GR}$ and $\mathbb{SW}$ heuristic policies, we must also compute the optimal queue length thresholds, $\mathcal{N}^*_{GR}$ and $\mathcal{N}^*_{SW}$. In practice, these would be set by the upper level agent or an external controller so as to minimize total expected cost over the possible range of arrival rates. We will do this numerically in our performance evaluations.

4.2 Benchmark Policies

We introduce some benchmark policies to evaluate the performance of our proposed heuristics. Detailed calculations of thresholds and costs under these policies are given in Online Appendix D.

Normal Operation ($\mathbb{NO}$): continue using the same decision thresholds as in normal situations (e.g., average arrival rate and queue lengths) without using information about either the new arrival rate or current queue length.

Reoptimization ($\mathbb{RO}$): adjust the decision thresholds so that they are equal to their optimal levels for the new arrival rate (but do not make use of real-time queue length information to alter thresholds).

4.3 Performance Analysis

To evaluate the performance of the two heuristics, we designed an extensive test suite that covers various parameter settings consisting of 85,750 cases. A detailed description of these settings is presented in Online Appendix B.

To consider performance across a range of utilizations, we let $\lambda_{norm} = 0.5$ and $\lambda_{\max} = 0.95$ denote the normal and max arrival rates (an average utilization that varies between 0.5 and 0.95 at the lower level), respectively. We consider 10 discrete arrival rates with 0.05 increments. Letting C^i_π denote the expected cost at arrival rate $\lambda_i = 0.5 + 0.05i$ $(i = 0, 1, \ldots, 9)$ under policy π, with C^0_π representing the expected performance under normal conditions, we consider the total expected

(maximum allowed queue lengths), the network remains stable even for large arrival rates. For instance, in some states, ED's are allowed to go on diversion. However, for an HKBSS in which balking does not occur, this arrival rate can also be found via stability analyses: the arrival rate at the lower agents should satisfy two conditions: (a) it should be less than the lower level service rate, and (b) be such that the total arrival rate of referred cases to the upper level under an optimal policy is less than the upper level service rate.

cost among all arrival rates: $TC_\pi = \sum_{i=0}^{9} C_\pi^i$. Here, we are giving equal weight to all C_π^i values because we believe a suitable policy is one that is potentially prepared for all arrival rates. Hence, we regard a policy π as a strong candidate if both TC_π and C_π^0 are close to that of the lower benchmark policy.

We start by comparing the performance of $\mathbb{RO}$ with the optimal policy under the IA scenario over 720 symmetric cases (i.e., $p_\ell^1 = 0.5$, $c_1 = c_0 = 1$, $\alpha_j^1 = \beta_j^0$ and $\beta_j^1 = \alpha_j^0$ $(j = \ell, u)$). These symmetric cases cover different values of system parameters including α_j^1, β_j^1, h, K, c_r and μ_u (see Online Appendix B for details). The results are summarized in Table 2. As the table shows, the optimality gap between $\mathbb{RO}$ and the optimal policy is on average 0.02 and 0.01 with respect to TC_π and C_π^0, respectively. Furthermore, in more than 95% (98%) of cases the gap in TC_π (C_π^0) is less than 5%. These indicate that the performance of $\mathbb{RO}$ is indeed very close to that of the optimal policy. This gives us confidence that we can use $\mathbb{RO}$ as a lower benchmark policy throughout the remainder of our numerical study.

To compare the performance of the proposed policies, we consider the following performance measures.

Opportunity (OP). This metric, which compares the gap between the cost of the upper bound ($\mathbb{NO}$) and lower bound ($\mathbb{RO}$) policies is defined as $OP = \frac{TC_{\mathbb{NO}} - TC_{\mathbb{RO}}}{TC_{\mathbb{RO}}}$. The value of OP (which is always positive) indicates the potential for improvement by using workload rebalancing to avoid cost explosions under spiked workload conditions. The larger this value, the more the potential benefit from workload rebalancing.

Efficiency Loss (EF). This metric, which measures loss in overall performance when using a heuristic policy ($\mathbb{GR}$ or $\mathbb{SW}$) instead of the lower benchmark policy ($\mathbb{RO}$), is defined as $EF_\pi = \frac{TC_\pi - TC_{\mathbb{RO}}}{TC_{\mathbb{RO}}}$ $(\pi \in \{\mathbb{GR}, \mathbb{SW}\})$. Obviously, we expect $EF_\pi \le OP$ $(\forall \pi \in \{\mathbb{GR}, \mathbb{SW}\})$. The lower the efficiency loss EF_π, the better the heuristic policy π. When EF is close to 0, it implies that the heuristic policy performs similarly to the lower benchmark policy $\mathbb{RO}$, and hence, is an efficient workload rebalancing policy. Note that since $\mathbb{RO}$ does not yield a true lower bound, it is possible for EF_π to be slightly lower than 0.

Adaptivity Loss (AD). This metric, which compares performance under normal operating conditions of the heuristic control policies ($\mathbb{GR}$ or $\mathbb{SW}$) relative to the lower benchmark policy ($\mathbb{RO}$), is defined as $AD_\pi = \frac{C_\pi^0 - C_{\mathbb{RO}}^0}{C_{\mathbb{RO}}^0}$ $(\pi \in \{\mathbb{GR}, \mathbb{SW}\})$. If a policy achieves low values of EF_π and AD_π, then it can be regarded as a "one-size-fits-all" type of strategy, since it can both resolve workload spikes effectively and perform well when there is no workload spike. We call such a policy "adaptive."

Table 3 presents the main results using the above metrics (for all 85,750 test cases) under IA and SA scenarios. In addition to computing the performance metrics, we performed paired T-tests

Metric	$(TC_{\mathbb{RO}} - TC_*)/TC_*$	$(C^0_{\mathbb{RO}} - C^0_*)/C^0_*$
Mean	0.02	0.01
Max	0.17	0.16
Fraction of cases with value $< 5\%$	95%	98%

Table 2: Performance comparison between $\mathbb{RO}$ and the optimal policy (denoted by "*").

on various system parameters (e.g., lower level and upper level knowledge, lower level and upper level error costs, upper level holding cost and upper level congestion related balking cost) to find the statistical relationship between these parameter values and the performance metrics. The results of the paired T-tests, as well as more details about the underlying numerical analysis, can be found in Online Appendix B.

Column 1 of Table 3 shows that, regardless of the assessment sharing structure, the $\mathbb{NO}$ policy does not work well compared to $\mathbb{RO}$. As the table shows, using workload rebalancing policies under the SA (IA) scenario can reduce the cost by an average of 22% (43%). This result together with the paired T-test results lead to the following observation regarding the opportunity metric.

Observation 3 (Opportunity) *Workload rebalancing in both SA and IA scenarios is an effective mechanism for mitigating the negative effect of workload spikes. Furthermore, regardless of the assessment sharing structure, OP typically increases as the holding cost decreases, rejection/balking cost increases, lower level knowledge decreases and upper level knowledge increases.*[16]

The reasoning behind these results is as follows. Under the $\mathbb{NO}$ policy, when the arrival rate is large, the upper level is likely to be overloaded, resulting in much higher rejection/balking and holding costs than under the lower benchmark policy. This is particularly the case in scenarios in which the "UP" region (see, e.g., Figure 4) is large under normal conditions, and hence, the upper level is more likely to be a bottleneck. Obviously, when the holding cost decreases (resulting in less incentive to allow cases to leave the system), lower level knowledge decreases, or upper level knowledge increases, there is more incentive to send a case to the upper level, and hence the "UP" region becomes larger. This causes OP to increase. The rejection/balking cost also has a positive effect on OP. This is because extra rejection/balking costs only occur when the upper level is overloaded, which becomes less likely when workloads are rebalanced. Hence, when the rejection/balking cost increases, $TC_{\mathbb{NO}}$ increases, but $TC_{\mathbb{RO}}$ is not affected as strongly, which causes OP to increase. The managerial implication of these results is that if the holding cost is not too large, the rejection/balking cost is not too small, and/or the upper level agent is significantly more experienced than the lower level agents, then the system is vulnerable to a workload spike.

[16] Knowledge level is measured in terms of accuracy and/or consistency of assessments.

Column #	1	2	3	4	5
Metric	OP	$EF_{\mathbb{GR}}$	$EF_{\mathbb{SW}}$	$AD_{\mathbb{GR}}$	$AD_{\mathbb{SW}}$
Average (IA)	0.22	0.01	-0.01	0.00	0.00
Variance (IA)	0.24	0.003	0.0001	0.0001	7E-05
Average (SA)	0.43	0.03	-0.01	0.00	0.00
Variance (SA)	1.3	0.010	0.0001	0.0001	9E-05

Table 3: Performance of heuristic policies under the IA and SA scenarios as measured by average and variance of various metrics.

Columns 2 and 3 of Table 3, along with the paired T-test results (Online Appendix B), lead to the following observation about the efficiency of the proposed heuristic policies ($\mathbb{GR}$ and $\mathbb{SW}$).

Observation 4 (Efficiency) *Under both the SA and IA scenarios, the heuristic policies* $\mathbb{GR}$ *and* $\mathbb{SW}$ *perform, on average, close to* $\mathbb{RO}$*. Furthermore, both policies are typically more efficient when lower level knowledge is close to upper level knowledge. However,* $\mathbb{GR}$ *becomes typically more efficient as the holding cost increases, but* $\mathbb{SW}$ *becomes typically more efficient as the holding cost decreases.*

To understand the intuition behind the above result, we note that both $\mathbb{GR}$ and $\mathbb{SW}$ reduce the fraction of cases that are sent to the upper level, albeit at a higher error cost. When the gap between the knowledge level of the lower level agent and higher level agent is small, referring fewer cases to the upper level does not result in a risk of a high error cost, and hence, the potential loss due to errors is low. Moreover, both $\mathbb{GR}$ and $\mathbb{SW}$ lose efficiency compared to $\mathbb{RO}$ (i.e., $EF_{\mathbb{GR}}$ and $EF_{\mathbb{SW}}$ increase) because the system can maintain a lower holding cost if more cases leave. The efficiency of $\mathbb{GR}$ is also low when the holding cost is low, since unlike $\mathbb{SW}$ it stops sending "difficult" jobs to the upper level whenever the upper level queue becomes higher than a threshold. In such a situation, $\mathbb{GR}$ does not selectively use the upper level capacity to reduce error costs. But when holding cost is not a dominant concern, paying attention to error costs becomes more important.

From Table 3, we can also compare the performance of the two myopic heuristic policies with each other by considering columns 2 and 3. From this comparison and the paired T-test results (Online Appendix B), we can make the following observation:

Observation 5 ($EF_{\mathbb{GR}}$ vs. $EF_{\mathbb{SW}}$) *On average, the overall performance of* $\mathbb{SW}$ *is better than that of* $\mathbb{GR}$*. However, the* $\mathbb{GR}$ *policy can be used instead of the slightly more complex* $\mathbb{SW}$ *policy without a significant performance loss unless the holding cost is low, upper level knowledge is high, or the lower level knowledge is low.*

This is due to the fact that, unlike $\mathbb{GR}$, $\mathbb{SW}$ mimics the threshold structure from the optimal policy. The managerial implication of the above results is that when holding cost is low, upper level knowledge is high, and lower level knowledge is low, the system should keep sending the most difficult cases to the upper level even when the upper level queue length is large. Therefore, the manager should choose the slightly more complex $\mathbb{SW}$ policy over the $\mathbb{GR}$ policy under such conditions. Under other conditions, the simpler $\mathbb{GR}$ policy is generally good enough.

We next consider the AD_π metric for policy $\pi \in \{\mathbb{GR}, \mathbb{SW}\}$ (i.e., columns 4 and 5 of Table 3), and present our main findings in Online Appendix F. As indicated there, we observe that for realistic ranges of parameters the heuristics exhibit reasonable performance under both normal and spiked workload situations, and hence, they should be considered as effective "one-size-fits-all" workload rebalancing policies.[17]

4.4 Robustness of Heuristic Policies

Finally, we study the robustness of our proposed heuristic policies, $\mathbb{GR}$ and $\mathbb{SW}$, to their optimal queue length thresholds $\mathcal{N}^*_{GR}$ and $\mathcal{N}^*_{SW}$. This is of practical significance, because as noted earlier many of the parameters that affect calculation of these thresholds may change over time. Furthermore, for many other reasons, it might not always possible to find the exact values of $\mathcal{N}^*_{GR}$ and $\mathcal{N}^*_{SW}$ in real-world settings. Our results presented in Online Appendix E indicate that the proposed heuristic control policies can be safely implemented, even if their control parameter ($\mathcal{N}^*_\pi$) is not fully optimized. This gives us further confidence that the proposed heuristics are robust to a variety of parameters that may change over time or parameters that are subject to misestimations for other reasons.

5 Conclusions

Hierarchical Knowledge-Based Service Systems (HKBSS), in which the interplay between workflow/queueing dynamics and knowledge-based decision making governs system performance, are prevalent in modern organizations. In this paper, we focused on the emerging practice of Telemedical Physician Triage (TPT), but also noted other settings in which similar structures occur. We constructed a POMDP model based on a novel model of agent knowledge, and used this framework to provide an analytic description of the optimal policy for processing and referring cases in a two-level system with binary decisions. This showed that the optimal decision thresholds are described by control-limits with an inverted V-shape structure. These imply that lower level agents should make decisions on a higher proportion of cases as the workload at the upper level grows. We used this structural insight to design two practical heuristic policies, which we term the Green/Red Light and Switching policies. Via numerical tests, we demonstrated that these are both effective

[17] In Online Appendix G, we also shed light on the ability of these policies to improve performance via assessment sharing.

in adjusting system performance to fluctuations in workload and robust to errors in the heuristic control parameters.

In addition to providing a practical framework for managing the time versus quality tradeoff in an HKBSS, our work provides several important managerial insights not obtainable from previously available models. By describing the sensitivity of the optimal policy to various environmental parameters, our results shed light on the factors that managers of HKBSSs need to consider. For example, by examining the sensitivity of the optimal policy to the decision error costs and interpreting those costs in the context of the TPT system, we showed that a hospital ED should increase its use of a remote telemedical physician to make patient triage decisions as congestion in the ED waiting area increases. This is an insight that clinicians experimenting with TPT did not have prior to our work. Furthermore, by examining the case mix which varies among EDs (e.g., a level 1 trauma center versus a community hospital), we demonstrated that different EDs need to utilize TPT in different ways. Specifically, if a patient with same medical condition is evaluated by the same telemedical physician for a level 1 trauma center and for a community hospital, the optimal policy may recommend classifying the patient as urgent in one but as non-urgent in the other. This insight is in sharp contrast with the prevailing belief that triage classification should depend only on the medical conditions of a patient. Our model highlights the reality that since triage classifications are used for prioritization, they should take into account the environmental and operational conditions in which they are made.

By leveraging our model of agent knowledge, we also described how agent training and information sharing between agents can be used to improve system performance. We showed that training can be useful by improving either consistency or accuracy in agent decision making. However, to be effective, training that targets consistency must focus on improving consistency for both types of patients in the binary space, but training that targets accuracy can be effective even if it improves accuracy for only one type of patient. These findings can help managers design the most suitable training strategies for their system.

Finally, we showed that sharing the assessment of the lower level agent with the upper level agent can improve decision making at the upper level. While the benefit of information is intuitive, our model further enabled us to show that assessment sharing is most useful in systems in which the upper level agent is more accurate than the lower level and the ratio of the error costs is low. A TPT system, where the error cost of misclassifying a non-urgent patient is typically smaller than that of misclassifying an urgent patient, has this property and so would benefit significantly from having the triage nurse share his/her assessment with the telemedical physician (e.g., through electronic forms). Again, this is a new and potentially useful insight for designing TPT systems.

References

Albrecht, T.L, V.A. Ropp. 1984. Communicating about innovation in networks of three U.S. organizations. *J. of Communication* **34** 78–91.

Alizamir, S., F. de Véricourt, P. Sun. 2013. Diagnostic accuracy under congestion. *Management Sci.* **59**(1) 157–171.

Anand, K., M.F. Pac, S. Veeraraghavan. 2011. Quality-speed conundrum: Trade-offs in customer-intensive services. *Management Sci.* **57**(1) 40–56.

Bassamboo, A., J.M. Harrison, A. Zeevi. 2006. Design and control of a large call center: Asymptotic analysis of an LP-based method. *Oper. Res.* **54**(3) 419–435.

Brass, D.J. 1995. A social network perspective on human resources management. *Personnel and Human Resources Manag.* **13** 39–79.

Burt, R.S. 1992. *Structural Holes: The Social Structure of Competition*. Harvard University Press, Cambridge, MA.

de Véricourt, F., P. Sun. 2009. Judgment accuracy under congestion in service systems. Working Paper, Fuqua School of Business, Duke University, Durham, NC.

Debo, L., S. Veeraraghavan. 2014. Equilibrium in queues under unknown service times and service value. *Oper. Res.* **62**(1) 38–57.

FitzGerald, G., G.A. Jelinek, D. Scott, M.F. Gerdtz. 2010. Emergency department triage revisited. *Emergency Med. J.* **27** 86–92.

Hogarth, R.M. 1980. *Judgment and Choice: The Phsycology of Decision*. Chichester: Wiley.

Hopp, W.J., S.M.R. Iravani, G. Yuen. 2007. Operations systems with discretionary task completion. *Management Sci.* **53**(1) 61–77.

Huberman, B.A., T. Hogg. 1995. Communities of practice: Performance and evolution. *Comuputational and Math. Organization Theo.* **1** 3–92.

Kostami, V., S. Rajagopalan. 2013. Speed-quality trade-offs in a dynamic model. *Manufacturing Service Oper. Management* **16**(1) 104–118.

Rajan, B., T. Tezcan, A. Seidmann. 2015. The process implications of using telemedicine for chronically ill patients: Analyzing key consequences for patients and medical specialists.

Saghafian, S., W.J. Hopp, M.P. Van Oyen, J.S. Desmond, S.L. Kronick. 2012. Patient streaming as a mechanism for improving responsiveness in emergency departments. *Oper. Res.* **60**(5) 1080–1097.

Saghafian, S., W.J. Hopp, M.P. Van Oyen, J.S. Desmond, S.L. Kronick. 2014. Complexity-augmented triage: A tool for improving patient safety and operational efficiency. *Manufacturing Service Oper. Management* **16**(3) 329–345.

Shumsky, R., E. Pinker. 2003. Gatekeepers and referrals in services. *Management Sci.* **49**(7) 839–856.

Tan, T.F., S. Netessine. 2014. When does the devil make work? an empirical study of the impact of workload on worker productivity. *Management Sci.* **60**(6) 1574–1593.

Traub, S.J., R. Butler, Y.H. Chang, C. Lipinski. 2013. Emergency Department physician telemedical triage. *Telemed J. E. Health.* **19**(11) 841–845.

Traub, S.J., C. Stewart, R. Didehban, D. Nestler, Y.H. Chang, S. Saghafian, C.A. Lipinski. 2015. Emergency Department rapid medical assessment: Overall effect and mechanistic consideration. *The J. of Emrgency Medicine* **48**(5) 620–627.

Wang, X., L. G. Debo, A. Scheller-Wolf. 2015. Managing nurse lines practical challenges and the developing theory. *International J. of Prod. Res.* (forthcoming).

Wang, X., L.G Debo, A. Scheller-Wolf, S.F. Smith. 2010. Design and analysis of diagnostic service centers. *Management Sci.* **56**(11) 839–856.

Wiler, J.L, C. Gentle, J.M. Halfpenny, A. Heins, A. Mehrotra, M.G. Mikhail, D. Fite. 2010. Emergency Department rapid medical assessment: Overall effect and mechanistic consideration. *Annals of Emrgency Medicine* **55**(2) 142–160.