Submitted to
manuscript

Strategic Idling and Dynamic Scheduling in an Open-shop Service Network: Case Study and Analysis

Opher Baron, Oded Berman, Dmitry Krass, Jianfu Wang
Rotman School of Management, University of Toronto, Canada

This paper, motivated by a collaboration with a healthcare service provider, studies dynamic scheduling policies (DSPs) for stochastic open-shop service networks with two objectives: more traditional macro-level measures such as minimizing total system time or minimizing total tardiness and the atypical micro-level measure of reducing the incidents of excessively long waits at any workstation within the process. While work-conserving policies are optimal for macro-level measures, scheduling policies with strategic idleness (SI) might be helpful for the micro-level measure. Using the empirical data provided by the service provider, we give detailed statistical evidence that SI is used by its schedulers to effectively manage the micro-level measure. However, the company has no specific rules on implementing SI and the schedulers make decisions based on their own experience. Our primary goal is to develop a systematic framework for the joint usage of DSPs with SI. We provide an efficient way to intelligently combine the SI and DSP such that the resulting policies can simultaneously address both macro- and micro-level measures. We use a simulation model based on empirical data to demonstrate that an open-shop service network can be managed in a systematic fashion to deliver improved service level by the joint usage DSPs and SI.

Key words: strategic idleness, dynamic scheduling policy, open shop, healthcare service network
History:

1. Introduction

The primary goal of this paper is to develop effective dynamic scheduling policies for stochastic open-shop processes operating under multiple objectives. The system objectives may include a combination of the more traditional “macro-level” measures (such as minimizing total system time and minimizing total tardiness) with the “micro-level” objective seeking to limit the number of incidents where a customer experiences an “excessively long” wait at a workstation within the process. This combination of objectives is motivated by the understanding that customer’s perception of service quality is affected by both “macro” and “micro” level factors.

Open-shop service networks, where customers need to visit a set of stations without a specific service order (other than some possible precedence constraints), are quite common in modern service industry in both, the “brick and mortar” and “virtual service” operations. Examples of the former include retail stores, where customers may visit several departments before proceeding to the cashier, and hospitals, where patients often go through several diagnostic and treatment stations. Examples of virtual systems include contact centers, where calls may be served by a mixture of automated response units and human agents with different levels of expertise, and on-line websites, where customers typically visit a number of pages before proceeding to the checkout page.

This paper was motivated by the example of an open shop in the healthcare service industry operated by XYZ (the real name of the company is removed and relevant data has been disguised for confidentially reasons) – one of the leaders in preventive healthcare services in North America. Their flagship service is composed of 10-20 different medical tests, each test performed in a different station. This service provides customers with a comprehensive evaluation of their current state of health and allows them to actively manage their health care. The order in which customers take most of these tests is immaterial (there are only a few precedence constraints), so XYZ actually operates an open-shop service system. XYZ’s service is primarily targeted towards busy professionals who are willing, for a fee, to have a complete assessment of their health performed in just a few hours. It should be noted that under the Canadian medical system, most of the tests can be done for free, but would likely take days or even months to schedule and complete. Thus, convenience is the main selling feature, and delivering excellent customer service is of paramount importance for XYZ. Tight management of customer waiting time in the system is deemed to be essential. While, due to the inherent variability of the times it takes to perform the different tests and procedures, some waiting time is unavoidable, the goal is to minimize waiting times and maximize customer perception of service quality. To study the waiting times in XYZ we interviewed their key personal and collected two months of data, comprising 41 business days with just over 2000 customers and about 24,000 station visits.

While there are many determinants of service quality in service networks, the link between customer waiting times and the perceived service quality is well-recognized (Friedman and Friedman, 1997; Taylor, 1994). Waiting times, that are easily quantifiable, have long been the focus of much of the queueing literature. The most common measure of waiting time is the expected overall waiting time for service. A related measure is the probability that the total system time or the total waiting time exceeds a certain predetermined threshold. XYZ uses both of these service level measures (SLMs) with the following stated targets: mean system time of less than four hours, and the probability of system time longer than four hours of less than 50%.

The two SLMs described above take a "macro" view of the service network, essentially treating it as a one-stage system and looking at the overall system or waiting time. Such SLMs may be sufficient in the manufacturing context, where customers, after placing an order, remain "outside the system" and essentially view the system as a "black box". However, in service contexts the customer is not just an outside observer: they experience the internal performance of the system as well, i.e., waiting times in front of individual workstations. A poor experience at a given workstation may lead to a poor perceived service quality, even when the macro-level SLM (e.g., the overall waiting time) does not indicate a problem.

In fact, XYZ's senior management noticed that there have been two types of complaints about the service experience: the first one is with respect to the total time customers spend in the system and the second, more prevalent one, is with respect to a long wait for a particular station. Specifically, from customer satisfaction surveys XYZ found that customers whose wait for service at any station exceeded 20 minutes were substantially less satisfied with their service than others. This led XYZ to define a "micro-level" SLM: the waiting time in front of any station should not exceed 20 minutes. This measure is taken very seriously: when a customer wait time at a station reaches 15 minutes an anxious "yellow face" appears on process schedulers' screens, alerting them to a potential issue. Once the waiting time reaches 20 minutes, an angry "red face" appears, and a service breakdown is considered to have occurred. The total number of red faces is carefully tracked: the goal is to keep this number below 100 per month.

The importance of such micro-level measures has been observed in other settings. For example, Bouch, Kuchinsky and Bhatti (2000) show that for an on-line service, when a single web page takes longer than 8 seconds to load, customer service quality ratings fall off dramatically. The adverse impact of long waiting time at a particular station is further supported by the marketing literature, e.g., Soman and Shi (2003) show that, given the total system time and price, people prefer a situation in which they are constantly making progress towards their goal; and by the psychology of queueing literature, e.g., Larson (1987) shows that customers' perception of the queueing experience may vary nonlinearly with the delay.

Nevertheless, the systematic treatment of such micro-level SLMs in queuing literature is relatively new. The only prior paper we are aware of is Baron, et al. (2013), where it was demonstrated analytically for a two station tandem queue network, that a scheduling policy with *Strategic Idling* (SI) might be helpful in reducing the probability of long waits. The idea behind SI policies is that when a downstream station accumulates a long queue, continuing to operate upstream stations at the normal rate may lengthen the queues downstream and increase the probability of long waits (i.e., the expected number of red faces). A better idea may be to idle the upstream stations (or to temporarily reduce their service rate), allowing the downstream queues to dissipate. A wise use of such idling effectively re-distributes the waiting times more evenly among the stations and reduces the number of red faces. We note that the scheduling policies employing SI violate one of the more common assumptions in queuing analysis: the "work conserving" property, which states that a workstation should continue to operate as long as there are customers waiting to be served. However, the potential payoff of SI may be very attractive. Indeed the "classical" way of reducing probabilities of long waits is by adding capacity to the system (e.g., adding a doctor in the healthcare setting), which is often quite expensive. However, using a dynamic scheduling policy with SI can potentially achieve the same objective at a negligible (or even a negative) cost: by simply idling some resources in the system.

Given the results from Baron et al. (2013) discussed above, and the fact that the number of red faces is used as part of performance evaluation for process schedulers at XYZ, one may expect that the schedulers will use the SI strategy. While our interviews with XYZ's management team indicate that such use of SI is not company's policy, our analysis of the empirical data points to convincing statistical evidence that XYZ's schedulers in fact do employ SI quite effectively to manage the number of red faces - simulation results indicate that the current scheduling rules without the use of SI would likely result in more than twice the current number of red face incidents.

Our primary interest is to study the benefit of dynamic scheduling policies (DSPs) and SI for open-shop service networks, such as the one operated by XYZ. We start by developing a framework that allows us to represent and implement general dynamic scheduling policies as simple scoring rules. A variety of different DSPs based on intuition and known theoretical results for stylized systems are proposed to address the macro-level SLMs. We then show how a *threshold-based policy* approach can be used to intelligently inject SI into a given DSP, resulting in a policy that can potentially address both macro and micro-level objectives simultaneously.

To test our policies we develop a simulation model for the XYZ process. The need for the simulation-based approach is driven by both, the complexity of stochastic open-shop networks, making analytical results very hard to obtain, and by the transient nature of real-life systems, such as the one operated by XYZ: since the system starts each workday with an empty queue and ends it in the same state after seeing relatively few customers, the steady-state regime may never set in. We use the empirical data on arrivals and service times to first understand the service process at XYZ and then to calibrate a detailed simulation model. Using this model we investigate the performance of several DSPs with and without SI and compare them to the performance of the empirical scheduling policies used in XYZ. We show that the automated policies achieve very promising results: the best DSPs are able to significantly outperform the actual schedules with respect to the macro-level measures. While without the use of SI the DSPs tend to perform poorly on the micro-level measures (the same effect is demonstrated for the actual scheduling policies), after the SI modification, DSPs are able to perform very competitively on the micro-level SLM, while maintaining their advantage with respect to macro-level SLMs.

To summarize, the paper makes several key contributions: (1) we narrow the gap between theory and practice on scheduling in open-shop service networks; (2) we shed light on the need for effective and systematic implementation of DSPs and SI to improve the SLM in such networks; (3) we provide a framework for developing algorithms for the joint usage of DSPs and SI in open-shop service systems; (4) using such algorithms and the simulation model of XYZ we demonstrate that an open-shop service network can be managed in a *systematic fashion* to deliver improved service level by jointly using DSPs and SI; and (5) we establish the usage of SI in practice (using statistical tests).

The plan for the remainder of the paper is as follows. In Section 2, we provide a brief literature review, focusing on known results for open-shop systems. In Section 3, we introduce the frameworks for using DSPs and SI in a stochastic open-shop service network. Then, using these frameworks, we propose several stylized DSPs and SI policies that are likely to be successful in practice. In Section 4, we analyze the empirical data provided by XYZ and present evidence for the usage of SI in XYZ. In Section 5, we use a simulation model to demonstrate the effect of SI and evaluate some of the DSPs and SI policies suggested in Section 3. In Section 6, we present conclusions and suggestions for future research.

2. Literature Review

Open shops were studied extensively in manufacturing settings (see, e.g., Roemer 2006), such as airplane maintenance, fire engine assembly, just-in-time systems, supply chain assembly systems, paper processing, and part kitting. A few papers consider service systems, such as accounting services, but they still consider common manufacturing measures as the service objective.

Open-shop problems are typically NP-hard and only consider macro-level measures (i.e., see Pinedo 2012 and reference therein). For the deterministic open shop with preemptions, polynomial time algorithms are available for the makespan objective, and maximum lateness objective. Without preemptions, for the makespan objective, only open shop with

$m = 2$ stations has the polynomial-time optimal policy (the Longest Alternate Processing Time first policy), and open shop with $m \geq 3$ stations is known to be NP-hard. For the maximum lateness objective, open shop with $m = 2$ stations is already strongly NP-hard. Furthermore, very little can be said about the total completion time objective $\sum_{i=1}^{n} F_i$; open shops with this objective is NP-hard for all $m \geq 2$ cases, with or without preemptions.

For the stochastic open shops, theoretical results are limited to the $m = 2$ case. Pinedo and Ross (1982) proved that the Longest Expected Remaining Processing time first (LERP) policy minimizes the expected makespan of a stochastic two-station open shop. Pinedo (1984) showed that the preemptive Shortest Expected Remaining Processing time first (SERP) policy minimizes the total expected completion time in a two-station open shop within the class of preemptive dynamic policies.

Alcaide et al (2006) developed a predictive-reactive approach to minimize expected makespan in an open shop with $m \geq 3$ stations; the approach is based on dynamically modifying a heuristic schedule, based on Alcaide et al. 1997, whenever an unexpected event occurs.

A vast literature is focused on the analysis, design, and control of queueing networks; see Stidham (2002) for a thorough survey of this research. Another important stream of queueing literature is focused on scheduling policies, which assign priorities to customers (or jobs) based on the current state of the system and the attributes of all customers. Scheduling policies can be categorized into two classes: static and dynamic. Static scheduling policies assign priority to each customer by a static rule which does not change while the customer is in the system. For example, to minimize customers' average waiting cost in a system with linear waiting cost rates, the $c\mu$ rule (see, e.g., Smith 1956) assigns static priority levels to customer k in an increasing order of $c_k \mu_k$ where c is the cost of waiting and μ^{-1} is the expected service time. Dynamic scheduling policies, in contrast, assign priorities that may be changed while customers are in the system. For example, to minimize customers' waiting cost in a system with convex waiting cost, the generalized $c\mu$ rule (see, e.g., Van Mieghem 1995) assigns dynamic priority levels to each customer k according to $c'_k(W_k(t))\,\mu_k(t)$, where $W_k(t)$ is customer k's waiting time at time t, and $c'_k(\cdot)$ is the first derivative of $c_k(\cdot)$.

Harrison (1996) used fluid models to provide asymptotically optimal scheduling heuristics under different objectives. This approach was extended by Maglaras (2000) who proposed discrete-review policies to translate the solution of the fluid optimal control into an implementable control policy in the stochastic network. For the job-shop scheduling problem with holding cost objective, Bertsimas et al. (2003) provided an efficient algorithm to round an optimal fluid solution such that the resulting schedule is asymptotically optimal. Dai and Lin (2005) proved that maximum pressure policies are throughput optimal in a class of stochastic processing networks. We do not use asymptotic approaches and directly consider scheduling in the stochastic network.

By far, the most popular objective in the literature employs is the total system time (see e.g., the survey by Gans, Koole, and Mandelbaum, 2003). A related measure is the probability that the total system time or the total waiting time exceeds a certain predetermined threshold. Baron, Berman, and Krass (2008), Baron and Milner (2009), de-Véricourt and Jennings (2011) and references therein also focused on the probability of long waiting time SLM.

Several other papers looked beyond the traditional measures. For example, de-Véricourt and Zhou (2005) analyzed a call-routing problem while considering both the call resolution probability and the average service time in the macro-level service level measure. Mehrotra et al. (2012) considered a similar problem with heterogeneous servers. Saghafian, Hopp, and Van Oyen (2012) analyzed the service policy in Emergency Departments while considering the weighted average of the expected length of stay and the expected time to first treatment.

The systematic study of the micro-level SLM focusing on the instances of excessive waits originates with Baron, et al. (2013), who demonstrate the advantage of policies with SI by applying a *Threshold Based Policy* (TBP). The idea behind the TBP is to compare the difference between queue lengths at different stations and to idle some upstream stations if this difference is larger than a predetermined threshold. They demonstrate the exact potential advantage of applying TBP in a tandem queue system with two servers.

There are two other settings where intentionally idling a capacitated resource has been previously considered. Strategic delays were first discussed in the literature in Afèche

(2013), who showed how such delays can allow a service provider to differentiate between customer types and thus improve the overall profit. The manufacturing process control literature also considers intentional idleness. The most prominent example is the Kanban manufacturing system where the total inventory between two stations is restricted to be lower than a threshold - for further details see Masin, Herrar, and Dar-el (2010) and references therein. In this case the motivation for idling is the need to control inventory and its cost without sacrificing too much capacity. In both cases, the motivation for intentional idling is significantly different from the current paper, which is motivated by improving the customer service experience. This difference leads to completely different analysis and implementation challenges.

3. Dynamic Scheduling Policies and Strategic Idleness Modification

We start by introducing stochastic open shop with precedence constraints in Section 3.1. We next develop a framework of completely reactive Dynamic Scheduling Policies (DSPs) that allow us to represent different DSPs in terms of simple scoring rules in Section 3.2. Using this framework, in Section 3.3, we propose several simple DSPs that have the potential to perform well in practice with regards to the macro-level Service Level Measures (SLMs). In Section 3.4, we show how any completely reactive DSP can be modified to "inject" Strategic Idleness (SI), allowing the policy to take into account our micro-level "red faces" SLM.

3.1. Stochastic Open Shop with Precedence Constraints

We consider a general stochastic open-shop problem with precedence constraints described as follows: a set of n customers $C = \{1, \dots, n\}$, who arrive at (possibly random) release dates $r_1^c, \dots, r_n^c$, wish to finish service within T^o time units after arrival (i.e., their due dates are $r_1^c + T^o, \dots, r_n^c + T^o$). The customers need to obtain service from a set of m stations $S = \{1, \dots, m\}$ that open at pre-scheduled release dates $r_1^s, \dots, r_n^s$ and close when all customers have finished service. For simplicity, we assume that $r_i^c < r_{i+1}^c$, for $i = 1, \dots, n-1$, and $r_j^s < r_{j+1}^s$, for $j = 1, \dots, m-1$, i.e., this implies no batch arrivals or stations openings at the same time. Customer i requires service from some subset

$S_i \subseteq S$ of stations, and she must visit every station in S_i exactly once. The order in which customer i receives services from stations in S_i is immaterial, as long as it satisfies precedence constraints $U_i = \{(h,k),\dots|h,k,\dots \in S_i\}$, where constraint (h,k) means that customer i must visit station h before becoming eligible for station k. For example, $U_i = \{(1,2),(1,3),...,(1,m)\,|1,2,...,m \in S_i\}$ means that customer i needs to visit station 1 before visiting any other stations. Note that, if $U_i = \emptyset$ for all i, the problem becomes a classic open-shop problem (see, e.g., Pinedo 2012). Customer i's service time at station j (for $j \in S_i$), X_{ij}, is a *continuous* random variable with distribution G_j. We assume that X_{ij} are independent and identically distributed for all j. The realization x_{ij} only becomes known upon service completion. We consider the problem *without* preemptions (i.e., customers are not allowed to leave the service at the current station before completion, nor is a station allowed to accept new customers before finishing the service of the current customer). The time customer i finishes service and exits the system is denoted by F_i, which is also a random variable.

We treat the scheduling problem as a multi-objective problem. Specifically, we consider two macro-level SLMs as our objectives: minimize **the expected total lateness**,

$$E\left[\sum_{i=1}^{n}\left(F_i - \left(r_i^c + T^o\right)\right)\right], \tag{1}$$

and minimize **the expected number of tardy customers**,

$$E\left[\sum_{i=1}^{n} 1\left(F_i > \left(r_i^c + T^o\right)\right)\right]. \tag{2}$$

Note that since r_i^c and T^o are independent of the scheduling policy, the expected total lateness objective is equivalent to the more typical expected total system time objective, $E\left[\sum_{i=1}^{n}\left(F_i - r_i^c\right)\right]$, or the expected total completion time objective, $E\left[\sum_{i=1}^{n} F_i\right]$.

In addition to the macro-level SLMs above, we consider a micro-level SLM as our third objective: minimize **the expected number of "red faces"** (i.e., the number of instances of unacceptably long waits),

$$E\left[\sum_{i,j \in S_i} 1\left(W_{ij} > T^s\right)\right], \tag{3}$$

where T^s is the threshold used to identify "red faces" and W_{ij} is the random variable denoting the time customer i spent in the waiting room before entering station j.

In the manufacturing literature, DSPs that consider unexpected real-time events (e.g., station breakdown, defective material, job cancelation, etc.) have been classified into three categories (see, e.g., Ouelhadj and Petrovic, 2009): (1) a completely reactive scheduling policy generates no firm schedule in advance and makes decisions locally in real-time; (2) a predictive-reactive scheduling policy develops a schedule first and revises it in response to real-time events following some scheduling/rescheduling methods; and (3) a robust pro-active scheduling policy follows a pre-set schedule that satisfies performance requirements predictively in a dynamic environment. Note that this taxonomy can also be applied in stochastic scheduling models with uncertainties caused by stochastic service times, as well as other of unexpected events.

Since, as we detailed in the literature review above, polynomial time algorithms for deriving the optimal scheduling policy in open shops are not available, it is hard to generate any firm schedule in advance. Therefore the advantage of predictive-reactive or robust pro-active DSPs is difficult to see for the open-shop service network we are interested in. Thus, we focus our investigation on completely reactive DSPs. We will initially consider only work-conserving DSP, i.e., a customer cannot be waiting for a station which is currently idle.

3.2. Work-conserving Dynamic Scheduling Policies in Open Shops

The completely reactive work-conserving DSPs in an open-shop environment take actions at three types of events: service completions, customer arrivals, and station openings. At these points, either a customer, or a station, or both, free up and must be "matched up" with the available customers/stations for the next stage of processing. In fact, by introducing dummy stations and customers, it suffices to only consider service completion events. We can imagine that the system starts with n dummy stations serving n customers with service completion times equal to customer arrival times $r^c_1, \dots, r^c_n$, and m stations serving m dummy customers with service completion times equal to station opening times $r^s_1, \dots, r^s_n$. Henceforward, we only consider decisions at service completions events.

Following a common simplifying assumption in queueing and stochastic scheduling literature, we assume that no two service completions happen at the same time, i.e., there exists an $\epsilon > 0$, such that the time interval between any two service completions is at least ϵ. This assumption fits XYZ and simplifies the discussion and rules below.

Consider a service completion involving customer i and station j occurring at some time t. We assume that at this time customer i enters the "waiting room" (either physical or virtual) and station j becomes idle. Let $\Omega_j \subseteq C$ be the set of *eligible* customers (i.e., satisfy all precedence constraints) that still require service from station j and who are in the waiting room at time t, and $\Psi_i \subseteq S_i$ be the set of idle stations at time t whose services are still required by customer i. Since any customer i only visits stations in S_i once, we have $j \notin \Psi_i$ and $i \notin \Omega_j$ at time t. Also, for any waiting customer h other than customer i, Ψ_h is either $\emptyset$ or equal to $\{j\}$, so we have $\Psi_i \cap \Psi_h = \emptyset$ (note that if $k \in \Psi_h$ for some $k \neq j$ then customer h was waiting while station k was idle, violating the work-conserving assumption). Similarly, for any idle station $k \neq j$, Ω_k is either $\emptyset$ or $\{i\}$, so we have $\Omega_j \cap \Omega_k = \emptyset$. This indicates that, at each service completion, the DSP needs to perform at most two assignments: assign a customer $h \in \Omega_j$ to station j (assuming $\Omega_j \neq \emptyset$) and assign a station $k \in \Psi_i$ to customer i (assuming $\Psi_i \neq \emptyset$); no other assignments are possible.

These observations allow us to represent a DSP with two scoring rules, where higher is better. We assign a score, $PT^c_h \geq 0$, to customers $h \in \Omega_j$, and a score, $PT^s_k \geq 0$, to stations $k \in \Psi_i$. To make PT^c_i and PT^s_j general enough, we define them as arbitrary non-negative functions of the history of the system up to time t.

Definition 1. Completely reactive and work-conserving $\mathcal{DSP}$ in stochastic open-shop networks is defined by scoring rules PT^c_i and PT^s_j as follows:
Suppose customer i completes service on station j at time t:
1) For the station assignment, if $\Omega_j \neq \emptyset$ we assign customer $h^* = \arg\max_{h \in \Omega_j} PT^c_h$ to be the next customer of station j; let station j stay idle if $\Omega_j = \emptyset$.
2) For the customer assignment, if $\Psi_i \neq \emptyset$ we assign station $k^* = \arg\max_{k \in \Psi_i} PT^s_k$ to be the next station of customer i; let customer i join the waiting room if $\Psi_i = \emptyset$.
Once these two assignments are made, the service process continuous until the next service completion event; then similar assignments are taken.

Note that the assumption that no two service completions occur simultaneously ensures that the definition above is complete. If, instead, several service completions occur at once, we can no longer assume that customer i is the only eligible waiting customer for any station in Ψ_i, nor that station j is the only idle station needed by any customer in Ω_j; there may be several customers "competing" for the same station and several stations "competing" for the same customer. Therefore, in addition to scoring rules, one must provide tie-breaking rules in order to specify the DSP in this case.

From Definition 1, we see that any DSP is completely determined by the selections of PT_i^c and PT_j^s. We next discuss different choices of these scoring rules that result in different DSPs.

3.3. Dynamic Scheduling Policies

To describe a DSP (or, more precisely, the scoring rules defining the policy) formally, we introduce the following notation (we omit t for convenience):

S_i^F: the set of stations customer i has visited by time t;

S_i^U: the set of stations customer i still requires service from at time t;

(Note that $S_i^F(t) \cup S_i^U(t) = S_i$, $\forall t$.)

u_j: the number of customers who still need service from station j at time t, i.e., $u_j = \sum_{i=1}^{n} 1\left(j \in S_i^U\right)$;

w_i^{TS}: customer i's total system time until time t, i.e., $w_i^{TS} = t - r_i^c$;

w_i^{TW}: customer i's total waiting time since she entered the system until time t;

w_i: customer i's current waiting time (i.e., the time since the last service completion of customer i and until time t);

$\bar{s}_j$: the average service time of station j;

n_j: the number of servers at station j.

Using the definitions above, for any $k \in \Psi_i$, the quantity $\frac{u_k \bar{s}_k}{n_k}$ represents the *remaining average workload* of station k. Since stations with the higher remaining average workload at time t can be thought of as "bottlenecks" over the remainder of the process, it seems reasonable to perform customer assignment so as not to keep more valuable resources idle. Thus, all DSPs we consider employ the same station scoring rule: $PT_k^S = \frac{u_k \bar{s}_k}{n_k}$, assigning customer i to the station with the highest remaining workload.

Our DSPs do differ with respect to station assignment rules (i.e., deciding which customer should be assigned next to the freed-up station j).

1. *Longest System time first (LS) policy* assigns to station j the customer in Ω_j who has the longest system time among all waiting customers who still require service from this station, i.e., $PT_i^c = w_i^{TS}$.

This rule is motivated by the idea that the customer who has already accumulated a long system time (because of waits or long processing times) is more likely to be tardy, and thus should be prioritized.

2. *Longest Mean Overage Processing time first (LMOP) policy* assigns to station j the waiting customer in Ω_j who has the longest mean overage service time, i.e., $PT_i^c = \frac{1}{|S_i^F|}\sum_{k\in S_i^F}(x_{ik} - \bar{s}_k)$.

Similar to the LS policy, LMOP policy prioritizes the customer who experienced longer than usual service times (represented by a longer mean overage service time). This customer thus has a higher risk of being tardy.

3. *Longest Accumulated Waiting time first (LAW) policy* assigns the waiting customer in Ω_j who has accumulated the longest waiting time, i.e., $PT_i^c = w_i^{TW}$.

This policy is also motivated by prioritizing customers who have a higher risk of being tardy. By counting only waiting time we are giving preference to customers who have already been “victimized” by long waits.

4. *Longest Current Waiting time first (LCW) policy* assigns the waiting customer in Ω_j who has the longest current waiting time, i.e., $PT_i^c = w_i$.

This policy follows the spirit of first-come-first-serve policy and prioritizes customers who enter the centralized waiting room earlier.

5. *Shortest Expected Remaining Processing time first (SERP) policy* assigns the waiting customer in Ω_j who has the shortest total expected remaining processing time, i.e., $PT_i^c = \left(\sum_{k\in S_i^U}\bar{s}_k\right)^{-1}$.

This policy is motivated by the optimality of preemptive SERP policy in a two-station open shop with the total expected completion time objective (see, e.g., Pinedo 1984).

6. *Longest Expected Remaining Processing time first (LERP) policy* assigns the waiting customer in Ω_j who has the longest total expected remaining processing time, i.e., $PT_i^c = \sum_{k\in S_i^U}\bar{s}_k$.

This policy is motivated by the optimality of LERP policy in a two-station open shop with the expected makespan objective (see, e.g., Pinedo and Ross 1982).

3.4. Strategic Idleness Modification - Generalized TBP

Recall that in addition to the two macro-level objectives (SLMs), the total lateness and the total number of tardy customers, we are also interested in the micro-level objective: the total number of "red faces" (incidents of long waits). Since such incidents often occur at bottleneck stations, one strategy to reduce their number is to intentionally delay service at stations that are upstream from the bottlenecks when there is already a long queue in front of the bottleneck station. We call such intentional delays "Strategic Idleness" (SI). The overall idea is to modify a given work-conserving DSP so that when the DSP assigns a free customer to a free station, instead of starting the service immediately, they both stay idle for a certain time period (which is determined by the SI policy and could be zero) prior to the service start. Note that with this modification the station remains assigned to a customer during SI period; thus customer and station assignments can be implemented using the same rules that are used to specify the original work-conserving DSPs. Note that non-idle policy can be thought of as a special SI modification where the idling time is always zero.

There are many possible policy classes that may involve SI. Baron et al. (2013) introduced a specific family of *Threshold Based Policies (TBP)*. As mentioned earlier, the idea behind the TBP is to compare the difference between queue lengths at different stations and to idle some upstream stations if this difference is larger than a predetermined threshold.

While defining a TBP in the two station tandem queue setting is straightforward, as there is only one upstream and one downstream station, it is already more difficult for a n-station tandem queue. The difficulty grows further in an open-shop service network, where not every customer may need to go through every station, and each customer may take a unique path through the network. Thus the "upstream" and "downstream" stations may not be clearly defined. To extend the definition of the TBP to this setting, we proceed as follows. Recall that at each time period, u_k gives the number of customers still requiring service from station k. For station j and customer i we define δ_{ij} to be a function of $u_1, \ldots, u_m$ which is decreasing in u_j, non-increasing in u_k for all $k \in \{1, \ldots, m\}$, $k \neq j$, and $\delta_{ij}(0, \ldots, 0) = 0$. To make δ_{ij} customer-specific, we allow it to depend on the set S_i^U. For every station j, we also define a non-negative threshold TH_j.

Definition 2. Modified policy DSP+TBP
When customer i is ready to enter station j under DSP, delay the service starting time as long as $\delta_{ij}(u_1, \ldots, u_m) \geq TH_j$.

We say that customer i is *stopped* (at station j) if this customer is assigned to station j while this station is *idled*. Intuitively, the TBP modification will idle station j if the number of customers who still require this station is low compared to other stations in the system. Several examples are presented below.

When customer i is stopped at station j under the TBP, there are, in principle, two options: (1) we can serve the next customer waiting for station j, allowing this customer to overtake customer i (provided this customer is not stopped under the TBP) - we call this "TBP with overtaking"[1]; (2) we can simply idle station j until the block is released under the TBP - leading to "Overtake-free TBP" (note that the latter option is more in line with the goal of minimizing excessive waits for the current customer).

The definition of TBP above is very flexible. The following specification of TBP will be used in the numerical experiment in Section 5.3:

Maximum workload TBP: the difference between the number of customers who need service from station j and the number of customers who need service from the busiest station still required by customer i, i.e., $\delta\left(u_1, ..., u_m, i, j\right) = \max_{l \in S_i^U} u_l - u_j$.

Of course, there are many other possible specifications of TBP. For example:

Maximum workload Kanban: the number of customers who need service from the busiest station still required by customer i, i.e., $\delta\left(u_1, ..., u_m, i, j\right) = \max_{l \in S_i^U} u_l$.

For stations with more than one server, we can also consider the number of servers in the specification:

[1] We note that overtaking does not arise for serial queues, since in such systems all customers have exactly the same service set S_i^U, and thus all customers waiting for station j would either be stopped or not stopped under the TBP.

Normalized Maximum workload TBP: the difference between the number of customers who need service from station j and the number of customers who need service from the busiest station still required by customer i normalized by the number of servers at the respective stations, i.e., $\delta\left(u_1,...,u_m,i,j\right)=\max_{l\in S_i^U}\frac{u_l}{n_l}-\frac{u_j}{n_j}$.

It is also possible to normalize either of the above rules by the expected service time required at each station.

4. Case Study - Data Collection and Analysis

In the previous section, we defined a number of different dynamic scheduling policies (DSPs) for a stochastic open shop and showed how they can be modified to incorporate the threshold-based strategic idleness delays. We next apply these ideas to the case of the medical clinic operated by XYZ Inc. The background of this case study and the data used in our analysis is described in the current section.

4.1. Company Background and Description of Data

The clinic operated by XYZ consists of up to 21 different medical tests. These include a series of routine diagnostic tests, including blood and urine lab tests, chest X-ray, Abdominal Ultrasound, Fitness Test, Treadmill Test, Physician Exam, Audio Visual Test, Nutrition and Review with doctors. While the stations above are required by almost every customer, there are add-on assessments that can be requested, such as Optometry, Echocardiogram, Genetic Risk Assessment, etc. Each test is conducted at a specific station with possibly multiple (up to 8) servers.

On average, each customer visits 10 stations, including all nine routine diagnostic tests and one add-on. The incoming customers are directed through the process by specially trained schedulers. Every time a customer finishes a test she is led to the waiting room from which she will be picked up for the next test.

As discussed above, XYZ considers three SLMs as their objectives: the expected average total system time, the expected number of tardy customers, and the number of red faces, given by (1-3), respectively. The company's goal is to complete the service in four hours (i.e., in (2) $T^o=4$ hours) and a “red face” is defined as a wait exceeding 20 minutes at any given station (i.e., in (3) $T^s=20$ minutes).

We first focus on obtaining a detailed picture of XYZ's actual service and waiting time performance. We obtained two months of data from XYZ. For each customer visit, the data contains the basic information of the appointment, such as the customer's appointment time, arrival time, and departure time. For each specific test, the data also contains the customer number, station number, starting time, ending time, and service time. During these two months, there were 41 business days, in which just over 2000 customers visited the clinic, and about 24,000 tests were performed. The number of customers who visited the clinic each business day ranged from 25 to 61 (with a mean of 49 and a standard deviation of 7.2).

4.2. The IT System at XYZ

XYZ's main selling point is convenience: instead of having to wait for weeks or months for all tests to be scheduled and performed, the customer can have the full assessment completed in a matter of a few hours, and have the results reviewed by the doctor who has a comprehensive view of the customer's current state of health. XYZ's clients are mostly busy executives and professionals who are willing to pay several thousand dollars for this convenience.

Not surprisingly, XYZ is extremely customer-focused, promising to complete the assessment in four hours. "Red face" incidents are regarded as significant service failures and are carefully tracked by the IT system. When a customer's wait reaches 15 minutes at any station, an anxious "yellow face" appears on the schedulers' screen, alerting them to bring this customer into service. When a customer's wait reaches 20 minutes, an angry "red face" flashes on the scheduler's screen, which typically triggers an immediate response - the customer is offered an apology, and the customer's service is expedited as much as possible. The number of red faces that occur during each day is tracked and used as part of performance reviews for process schedulers and their supervisors.

4.3. Operational Procedures at XYZ

XYZ schedules the arrivals of its customers at different times throughout the morning of each day. The clinic opens at 7:00am, and closes when all customers leave, typically around 16:00pm. Figure 1 illustrates the histograms of average daily scheduled arrivals alongside

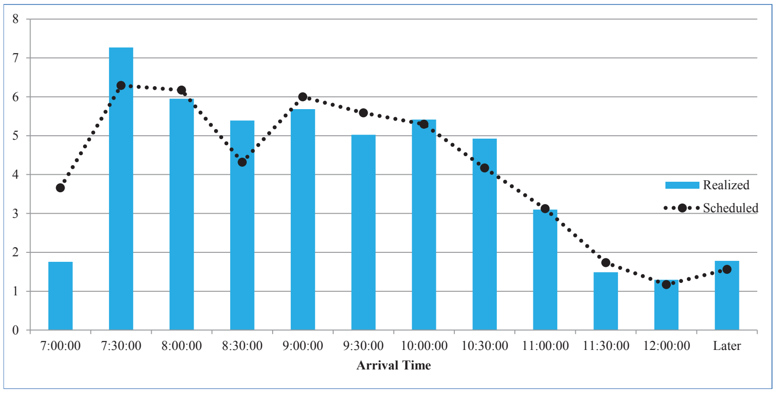

Figure 1 The Histogram of Average Daily Arrivals.

the histogram of average daily realized arrivals. Note that, the resulting arrival pattern is not stationary (contrary to the assumptions of traditional queueing models). On average, XYZ schedules 5 customers every half hour starting from 7:00am and typically until 10:30am. Then, from 10:30am to 12:00pm, the number of scheduled arrivals is gradually reduced.

We next investigate the order of service. While the order in which services are performed differs by customer, some dominant flows can be ascertained. To this end, we define $p_{i,j}$ as the probability that a customer visits station i before station j, given that station i and j are both visited by this customer:

$$p_{i,j} = \frac{\text{Number of times station } i \text{ is visited before station } j}{\text{Total number of visits containing both station } i \text{ and } j}.$$

For example, suppose that three customers, A, B and C, visit the clinic. Customers A and B visit station i before station j, while customer C visits station j before station i. In this example, we have $p_{i,j} = 67\%$, $p_{j,i} = 33\%$.

Table 1 presents p_{ij} for the nine routine stations. These values allow us to identify some dominant work flows For example, the 98% in the first row (Lab Work) and second column (Abdominal Ultrasound) means that 98% of the time Lab Work is completed before the Abdominal Ultrasound.

Note that $p_{i,j} + p_{j,i} = 100\%$ may not always hold, because of the occasional need to re-do a test (this affects less than 1% of all customers). The same reason causes the non-zero

	Labwork	Ab Ultra	PhyExam	Treadmill	Nutrition	FitnessTest	Xray	AuViTest	ReviewDr.
Lab Work	1%	98%	99%	99%	100%	100%	95%	100%	100%
Ab Ultra	3%	1%	73%	84%	100%	98%	82%	96%	100%
PhyExam	2%	27%	1%	69%	64%	68%	59%	69%	100%
Treadmill	1%	16%	29%	1%	59%	67%	55%	73%	99%
Nutrition	1%	0%	34%	37%	0%	56%	49%	58%	82%
FitnessTest	1%	3%	31%	32%	40%	1%	44%	52%	85%
Xray	5%	14%	31%	35%	38%	44%	9%	46%	63%
AuViTest	1%	5%	30%	26%	37%	46%	41%	1%	93%
ReviewDr.	0%	0%	0%	0%	13%	14%	19%	7%	1%

Table 1 The Service Order Matrix

items on the diagonal. In addition, some customers may choose not to perform all tests, resulting in $p_{i,j} + p_{j,i} < 100\%$ for some stations i and j.

There are three main observations from Table 1:

1. The two tests Lab Work and Abdominal Ultrasound are visited before any other stations in almost all cases. Since these two tests need to be performed on an empty stomach, the schedulers attempt to put them at the beginning of each customer's visit, so that customers can have a snack as soon as possible. Note that Lab Work, which is faster, is done before Abdominal Ultrasound 98% of the times.

2. The procedure "Review with a Doctor" is typically done after the customer finishes most tests. In this station, the doctor receives reports from all other stations and thus have a comprehensive view of customer's health situation. Although customers are given the option to skip this step and receive the test results via email, most of XYZ's customer choose to attend this station.

3. The order of customers' visits to all other stations is quite random - substantiating the view of this system as a open shop with only a few precedence constraints.

In view of these three observations, the network operated by XYZ can be loosely separated into three parts: starting with Lab Work, Abdominal Ultrasound and breakfast; continuing with the other required or optional tests in some random order; and concluding with reviewing results with the doctor. The network is illustrated on Figure 2; all three triangles marked with "W" represent the same centralized waiting room.

4.4. Waiting Time Distribution

Figure 3 illustrates the histogram of waiting times with bin interval of a minute. The label above each column represents the relative frequency of waiting times in the bin $[t-1,t)$, for

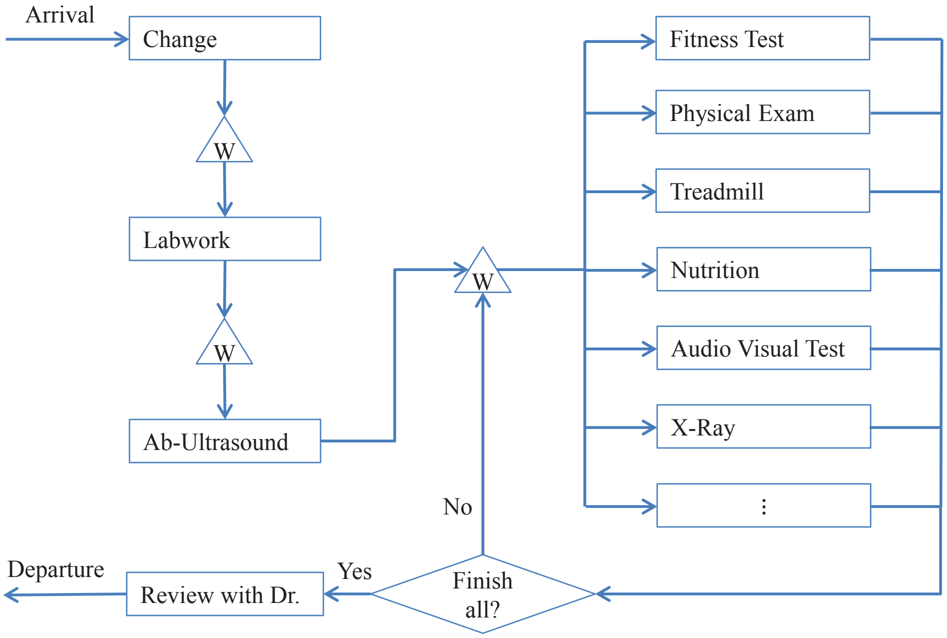

Figure 2 Typical Service Order at XYZ.

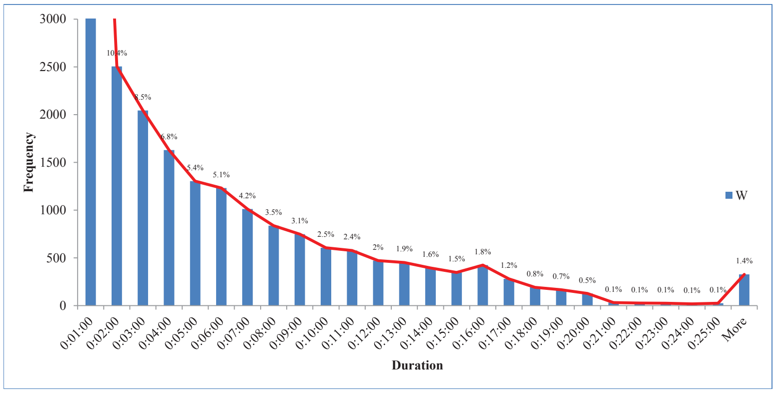

Figure 3 The Histogram of Waiting Times.

$t = 1, 2, ..., 30$. The mean and standard deviation of the waiting times are 5 and 7 minutes respectively.

Observe that the frequency of W decreases with t smoothly, except at two points: $t = 16$ and 21. At $t = 16$, the histogram of W breaks the decreasing pattern and showing an unusual peak. At $t = 21$, the histogram decreases steeply to 0.1% (from 0.5% at $t = 20$), and it stays at a fixed level 0.1%. These two sudden changes in Figure 3 appear to be related to the appearance of yellow faces (at 15 minutes) and red faces (at 20 minutes) on the schedulers' screen.

The schedulers try their best to keep the number of red faces low. Our hypothesis is that these sudden changes in the histogram of W are reflections of these two alarm signals. No special action is taken before the first alarm. Once a yellow face appears, schedulers attempt to expedite, causing density concentration at 15 minutes. From this point, the customer is watched very carefully, and, for the most part, not allowed to get beyond 20 minutes. This causes the dip at 20 minutes. Discussions with the personal at XYZ further supports this hypothesis.

These observations demonstrate that the schedulers manage the waiting time to control not only the macro-level SLMs, like the total system time, but also the micro-level SLM, i.e., the number of red faces.

4.5. Performance Analysis

In this section, we analyze the performance of XYZ's open-shop network focusing primarily on the nine routine stations. To calculate the utilization of each station, we first focus on each of its servers, and calculate the average starting time (when the first customer arrives) and the average closing time (when the last customer leaves). Then, for each station, based on the statistics of its servers, we derive the average time span (the time between starting and closing) and the average busy time in it. At last, each station's utilization is obtained as the ratio of its average busy time and its average time span. We attribute the waiting time to the station that immediately followed the wait.

Table 2, sorted by utilization, summarizes the performance of these nine routine stations over the two months. The table also provides the three main SLMs and average total waiting time.

Based on the empirical data, the service levels were already quite good. The current average total system time is just over four hours. The average total waiting time is about

Stations\Ave.	# Servers/day	Waiting Time	Service Time	Utilization	# Red Faces
FitnessTest	7.4	5:46	20:28	78%	45
PhyExam	7.7	4:50	32:45	73%	59
Doc Review	7.7	7:06	14:36	73%	173
Treadmill	4	4:48	21:50	72%	20
Ab Ultra	4	5:23	16:16	71%	10
Nutrition	3.9	4:42	19:49	69%	18
AuViTest	3.9	6:57	16:51	63%	55
Xray	1	5:23	6:14	50%	6
Labwork	1	4:19	3:23	48%	2

Ave.SystemTime	4:04:26
SystemTime≥4hrs	52.5%
Ave.TotalWaitTime	1:01:08
# Red Faces	456

Table 2 Summary of CHA Stations in the Empirical Data.

an hour, i.e., less than a quarter of the total time a customer spends in the system. The incidence of red faces (waits more than 20 minutes) is 456: with 10 stations per visit on average (9 routine stations and 1 “add-on” station), this corresponds to less than 2.5% of all tests and about 25% of customers experiencing a “red face”.

From Table 2, we see that the overall utilization, between 48% and 78%, and the waiting time, averaging between 4-7 minutes per station, are not high. In a service network with moderate variability (observed coefficient of variations were between 0.24 and 0.6 and close to 0.5 on average), we expect relatively low utilizations to be required to maintain short waiting times. This confirms the customer-centric emphasis of XYZ.

We also observe that Review with Doctor, Fitness Test, and Audio Visual Test account for most of red faces incidents; these stations also have the longest average waiting times. The Review with Doctor is the one contributing the largest number of red faces (38% of the red faces). Although it seems like a good idea for XYZ to hire more doctors to improve the SLMs, the cost of this action may be prohibitive.

The Fitness Test and Audio Visual Test together generate 100 red faces (21%) in total. The main problem at these two stations appear to be late starting times. On average, the Fitness Test and the Audio Visual Test start 2.5 hours and 1.5 hours, respectively, after the clinic opens. Moreover, they are often not at full capacity as some operators arrive even later.

We note that while the Fitness Test and Review with Doctor may be considered bottleneck stations of the network, since they both have high capacity, utilization and large associated waiting times. The Audio Visual Test, with a relatively low utilization, does not meet the standard definition of a bottleneck station. Therefore, for lack of a better name, we call Fitness Test, Review with Doctor, and Audio Visual Test *problematic* stations, and other stations "non-problematic".

4.6. Evidence of Strategic Idleness in Practice

As discussed earlier, Baron, et al. (2013) demonstrated analytically, in a two-station tandem queue network, that a scheduling policy incorporating Strategic Idleness (SI) might be helpful in reducing the number of red faces. The basic idea is that when a long queue accumulates at the downstream station, it may be better to idle the upstream stations to allow the downstream queues to dissipate. However, in service operations, we are not aware of any empirical evidence of the use of SI in practice. As discussed below, the data obtained from XYZ strongly suggests that this technique is, in fact, practiced by XYZ's process schedulers (largely without the knowledge of management).

From the empirical data provided by XYZ, we discovered a number of instances where a customer was waiting for a station that was free and waiting for this customer. In other words, certain part of customers' waiting time is spent waiting for an idle station that appears to be waiting for this customer; we call such period an Overlapped Waiting (OW) time.

Initially, we thought that the existence of OW is a data error, but OWs are quite abundant and accompany 78.7% of services. Figure 4 depicts the histogram of the OW with bin intervals of one minute. The average OW is 2.5 minutes with a Standard Deviation of 4.5 minutes. About 50% of the OWs are less than one minute. However, 16.3% of OWs are more than five minutes, i.e., more than the average waiting per station.

One explanation is that OW might be a result of routine procedures, like room cleaning, writing reports, etc. However, as verified by our partner at XYZ, room cleaning or report writing typically do not take that long, so while these may explain the shorter OWs observed, they do not explain OWs of over 1-2 minutes.

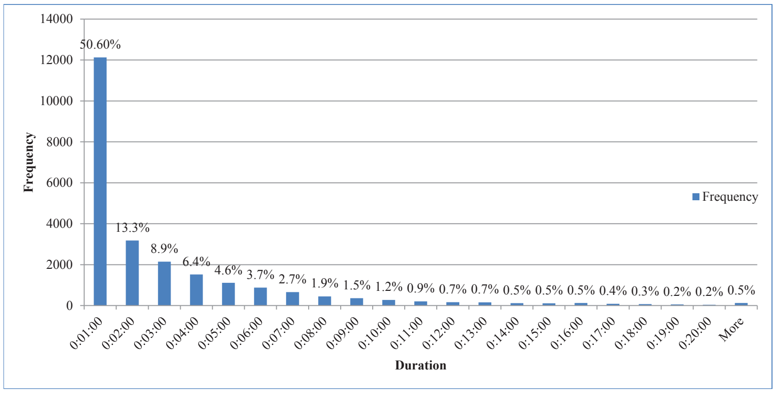

Figure 4 The Histogram of Overlapped Waiting.

As discussed earlier, SI can be an effective strategy for reducing the occurrence of long waits within the process. Thus given the importance placed by XYZ on minimizing the number of "red faces", an alternative explanation is that longer OWs provide the evidence of the usage of SI by the schedulers at XYZ. Of course, the mere presence of OW does not necessarily indicate that SI is being used. However, the prevalence of longer OWs in situations where SI policy would be most useful does provide support for this hypothesis. Below, we use statistical hypothesis testings to demonstrate that the timing of longer OWs provides evidence for the usage of SI at XYZ.

4.6.1. Statistical Evidence of the Use of Strategic Idleness Based on our intuition and the earlier discussion, if SI is indeed used, we would expect this to be reflected in OWs as follows. When station A is ready to serve customer i, and customer i's next station is station B, which is congested, station A would use longer OWs to balance customer i's waiting time at both stations. The congestion at station B can be indicated by the fact that station B is a problematic station (Fitness Test, Review with Doctor, and Audio Visual Test), or that station B's previous customer (served just before customer i) experienced a long waiting time (e.g., ≥ 15 minutes) just before entering station B. In the later case, we say that station A precedes a "potential long wait". Similar logic suggests that to avoid wasting capacity in problematic stations, these stations should use shorter OWs than other stations. Specifically, we anticipate:

1. OWs at problematic stations are shorter than OWs at non-problematic stations;

2. OWs at stations preceding problematic stations are longer than OWs at stations preceding non-problematic stations;

3. OWs at stations preceding “potential long waits” are longer than OWs at other stations.

The statements above can be investigated using the standard t-tests. First, we test if problematic stations have different mean OW than non-problematic stations:

Null Hypothesis (H_0): The difference between the mean OW at problematic stations and the mean OW at non-problematic stations is zero;

Alternative Hypothesis (H_A): The difference between these two mean OWs is not zero.

The two-tailed t-test (with p-value 2.9×10^{-15}) indicates that H_0 is rejected, and the non-problematic stations have significantly longer mean OW (with mean 2:42) than the problematic stations do (with mean 2:13).

Second, we test if the last station preceding problematic stations have different mean OW than stations preceding non-problematic stations:

H_0: The difference between the mean OW at stations preceding problematic stations and the mean OW at stations preceding non-problematic stations is zero;

H_A: The difference between these two mean OWs is not zero.

The two-tailed t-test (with p-value 1.3×10^{-18}) indicates that H_0 is rejected, and the stations preceding problematic stations have significantly longer OWs (with mean 2:55) than the stations preceding non-problematic stations (with mean 2:23).

Third, we test if stations preceding “potential long waits” have different mean OW than other stations:

H_0: The difference between the mean OW at stations preceding “potential long waits” and the mean OW at other stations is zero;

H_A: The difference between these two mean OWs is not zero.

The two-tailed t-test (with p-value 3.2×10^{-19}) indicates that H_0 is rejected, and the stations preceding ‘potential long waits’ have significantly longer OWs (with mean 3:38) than other stations do (with mean 2:09).

Thus, the statistical results strongly suggest that the schedulers are indeed attempting to reduce the number of red faces by using SI.

5. Evaluation of Dynamic Scheduling Policies and Strategic Idleness

The service system operated by XYZ, just as many real-life service systems, is inherently transient (each day starts with an empty system and ends after processing 49 customers on average) and it is not clear that the steady-state regime is ever achieved. Since analytical methods run into significant difficulties when analyzing transient behavior, we developed a simulation model of XYZ to gain better understanding of the system and to test the performance of different policies described in Section 3.3 and 3.4 above. The simulation model was implemented in MATLAB and validated using the empirical data. Specifically, we used real service and arrival times in the model. With the simulation model, we are able to: 1) Analyze the transient behavior of XYZ's service network. By simulating the system operations over one day for 100 times (from arrival of the first customer and until the departure of the last customer at the end of the day), we can ensure that the distribution of system's performance is captured by the simulation model. 2) Simulate scheduling decisions that depend on the state of the system. Both the customer and station assignments rules can be replaced with one of the policies described in Section 3.3. 3) Measure the macro-level and micro-level SLMs: the expected average total system time, the expected probability of total system time longer than four hours, and the number of red faces. 4) Simulate the performance of scheduling policies that incorporate strategic idleness (as described in Section 3.4).

The simulation recorded the resulting three SLMs for each scheduling policy. For each working day, we keep the available resources (station availabilities and opening times), customers' service needs and customers' service times at different stations the same as in the real data, and only change the scheduling policy.

In Section 5.1, we investigate the performances of DSPs without SI modifications (work-conserving) to evaluate the benefit of automated DSPs with respect to the global performance measures. In Section 5.2, we compare the performance of the actual scheduling policies used by XYZ with and without OWs. Next in Section 5.3, we compare DSP+TBP policies with the actual ones.

5.1. Performance of Work-conserving Dynamic Scheduling Policies

The simulated performance of various dynamic scheduling policies described in Section 3.3 is presented in Table 3. The third row of the table contains the Empirical Data (ED), i.e., the results for the actual scheduling policy used by XYZ. We start by comparing the various Dynamic Scheduling Policies (DSPs) without the Strategic Idleness (SI) modification introduced in Section 3.4 - we call these the "non-idle" versions of the respective policies. The results for all policies can be found in rows 5-10 of the table.

We first observe that the Shortest Expected Remaining Processing Time first (SERP) policy dominates the LAW, LCW, and LERP policies: SERP has a lower average total system time, a lower probability of spending more than four hours in the system, and a lower number of red faces. Comparing SERP policy with the Longest System time first (LS) policy shows that LS outperforms SERP with respect to the number of red faces, while SERP policy performs better on the other two SLMs.

We also observe that all of the DSPs outperform the actual (ED) scheduling policy with respect to both macro-level SLMs used by XYZ: they reduce the average total system time by about 40 minutes (16%), and the proportion of customers experiencing total system times of over four hours from 52.5% to around 21%.

However, for all DSPs the incidence of red faces is substantially higher than for the ED policy. Even the best-performing Longest Mean Overage Processing time first (LMOP) policy experiences an increase in this measure by 34.2% to 612 vs. the ED policy - a clearly undesirable outcome. We suspect that the reason that the DSPs we proposed perform poorly with respect to the number of red faces measures is that they are work-conserving (non-idle) policies, while, as discussed in Section 4.6.1, it appears that the XYZ's schedulers are actively using Strategic Idleness to manage the number of red faces. We further investigate this issue in the following section.

5.2. Effect of Overlapped Waiting Times: Another Indication of Strategic Idleness

To investigate the usage of Overlapped Waiting (OW), we define the "non-idle" version of the actual scheduling policy by eliminating the OWs. For example, suppose customer A finishes service at station i. To find the next customer for station i, we examine the data and find that customer B is the next customer of station i. If customer B is in the waiting room and station i is her next station (based on data), she is immediately taken to station i; otherwise, station i stays idle until the arrival of customer B. We follow a similar rule when choosing the next station for customer A. We call this non-idle version of the actual scheduling policy "ED+Non-idle". The corresponding results are presented on row 4 of Table 3.

Our simulation results indicate that if the clinic was operated under ED+Non-idle policy during the two month in our data, the average total system time would drop by 18 minutes to 3:46:37, and the proportion of customers with system time of over four hours would decrease by 14.3% to 38.3%. Thus, both of the macro-level SLMs would improve substantially (though the improvements still fall far short of those observed under the DSPs described earlier). However, the total number of red faces would increase to 1094 (140% increase). Out of the 456 red faces observed in the data, 217 disappear in the simulation under the ED+Non-idle policy, while 855 new ones emerge. Shorter waiting time at previous stations, i.e., elimination of SI in the form of OW, causes 705 of these new red faces.

The mean and standard deviation of the waiting time from the simulation result for ED+Non-idle policy are 3.5 minutes and 7.5 minutes respectively. Figure 5 shows the histogram of W (waiting time) for the ED+Non-idle policy alongside the histogram for the ED policy. We see that the histogram of W from ED+Non-idle policy is much smoother than the histogram from the actual scheduling policy. Those odd jumps at the $t = 16$ or 21 minutes observed in the empirical data disappear, while the number of waits of more than 15 minutes is at the same level (about 2000) for both ED and ED+Non-idle policies.

Comparing the ED+Non-idle policy with the ED policy indicates that by intentionally holding back customers at non-problematic stations when a customer is likely to experience a long wait at the problematic station downstream, the ED policy effectively re-distribute the waiting times more evenly within the network. These observations further support the conclusions from our statistical hypothesis tests that XYZ's schedulers are using SI.

To summarize, the results above indicate that the reason that ED policy outperforms those proposed policies in the number of red faces is that these policies are non-idle policies,

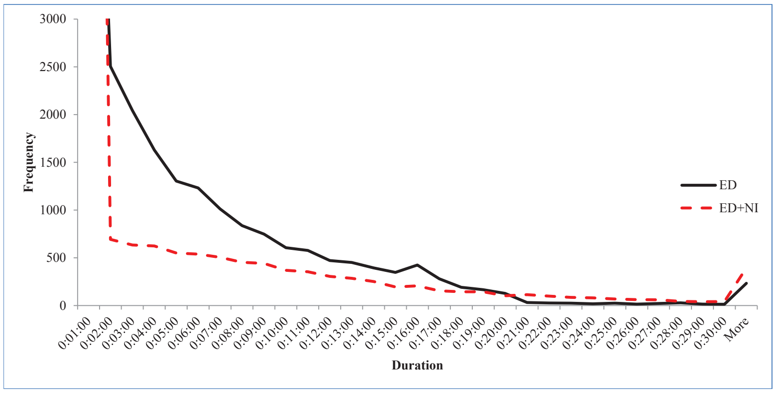

Figure 5 Histogram of Waiting Times under Different Scheduling Policies.

while the XYZ's schedulers are using SI in their scheduling policy. However, the schedulers of XYZ are using their own intuitions to insert OW. There were no official policies to this effect - as discussed earlier, the upper management seemed to be unaware of this practice.

In the following section, we introduce SI into our proposed DSPs by using the Maximum Workload Threshold Based Policy (TBP) as discussed in Section 3.4 earlier.

5.3. Performance of DSPs with Strategic Idleness Modification

In this section, we add the SI to the different DSPs. Specifically, we use the generalized TBP modification to the LAW, LS, LMOP, LCW, SERP and LERP policies. We demonstrate that this modification can reduce the number of red faces substantially without a significant deterioration in the values of the macro-level SLMs. We employed the "Maximum Workload TBP", described in Section 3.4, to the six proposed DSPs, and implemented the Overtake-free SI. This ensures that, after the SI period, customers would not be delayed once station resumes working. The results can be found in the last six rows of Table 3.

Comparing rows 5-10 to rows 11-16 reveals that in all six DSPs, the SI policy results in a small 5-7% increase in total system times (by about 11 minutes in the LAW, LS, LMOP and SERP policies, and about 13 minutes in LCW and LERP policies) and 8-10% increases in the probability of experiencing total system times of over four hours versus the non-idle version of these policies. However, the red face measure (i.e., the number of waits over 20 minutes) is reduced by around 24%. This is in line with the theoretical analysis presented

	System Time			Red Faces				
Policies \ Measures	Mean	Stdev	≥4hrs	#(≥ 15mins)	#(≥ 18mins)	#(≥ 20mins)	#(≥ 22mins)	#(≥ 25mins)
Empirical Data	4:04:26	1:00:51	52.5%	1645	749	456	397	327
ED+Non-idle	3:46:37	56:53	38.3%	1846	1340	1094	880	645
LAW+Non-idle	3:23:53	49:55	21.3%	958	822	730	661	541
LS+Non-idle	3:24:21	47:44	21.3%	873	743	665	607	520
LMOP+Non-idle	3:24:28	52:01	20.4%	807	683	612	543	457
LCW+Non-idle	3:24:19	50:07	21.2%	972	816	729	653	554
SERP+Non-idle	3:23:17	50:57	20.6%	929	784	687	606	507
LERP+Non-idle	3:25:22	50:58	20.9%	926	791	705	633	528
LAW+MaxWrkldTBP	3:34:32,	51:46	28.0%	828	647	551	477	393
LS+MaxWrkldTBP	3:35:03	51:29	28.2%	841	622	518	444	360
LMOP+MaxWrkldTBP	3:35:55	53:40	28.9%	707	539	476	427	355
LCW+MaxWrkldTBP	3:37:17	52:54	30.7%	845	640	551	478	380
SERP+MaxWrkldTBP	3:34:25	52:35	28.2%	792	618	525	464	380
LERP+MaxWrkldTBP	3:38:58	53:16	31.7%	878	692	609	541	445

Table 3 Performance of Different Scheduling Policies with and without SI.

in Baron et. al. (2013): the SI policy is able to significantly reduce the probability of long waits (an equivalent measure to red faces), while only slightly increasing the total system times. Note that the SERP+TBP policy again dominates LAW+TBP, LCW+TBP and LERP+TBP policies.

While the numbers of red faces under our DSPs with TBP modification are greater than under the ED policy (476 vs. 456), a significant reduction in total system times, as well as an automated DSP without any supervision presents an attractive trade-off to the decision-maker. Incidentally, the difference in the number of red faces versus the ED policy may be due to the fact that in real system expediting action appears to be taken by process managers somewhat before the waiting time reaches 20 minutes (which is natural, given that red faces incidents are used in performance reviews). If we redefine “excessive waits” to be 15 or 18 minutes, rather than the current 20 minutes, the incidence of such waits under all six DSP+SI policies falls below that in the empirical data.

Our main finding from this comparison is that the use of strategic idleness in conjunction with the dynamic scheduling policies, such as our LS, LAW and SERP policies, can be used to improve the SLM in practice. Moreover, due to the simple and transparent structure, the cost of implementing such policies should be low. In addition to supporting the main theme of the paper on the joint usage of DSPs and SI, this finding is also encouraging for the usage of decision support systems for managing service network in practice.

6. Summary and Open Questions

In this paper we investigated the performance of Dynamic Scheduling Policies in a stochastic open-shop service network. The unique feature of the system we examined is the need to balance between the more traditional "macro-level" service level measures, such as the total system time, and the customer-focused "micro-level" measure related to excessive waits within the system. The incidence of excessive waits can be managed by introducing "strategic idleness" where intentional (small) waits are introduced in upstream stations to prevent (longer) waits at busy downstream stations. Our work was motivated by the data from a real-life medical clinic. Through process analysis and statistical hypothesis tests we demonstrate that (largely unbeknownst to management), system schedulers appear to use strategic idleness to minimize instances of excessive waits.

We developed a flexible framework allowing us to represent "completely reactive" dynamic scheduling policies by defining simple scoring rules. We also showed how a given policy can be modified with a "strategic idleness" component allowing it to account for both micro- and macro-level measures. By developing a simulation model based on the real data for the XYZ system we showed that these automated scheduling policies appear to be quite promising: achieving substantial improvements on the macro-level measures while essentially matching the performance of actual policies on the micro-level measure.

Due to the complexity of the underlying system, our results are mostly computational. Analytical substantiation of some of our conclusions would be quite interesting. A step in this direction was taken by Baron et al. (2013) who investigated policies with strategic idleness analytically for a tandem queue. An extension of their results to more complex stochastic networks remains open.

References

Afèche, P. (2013) Incentive-compatible revenue management in queueing systems: optimal strategic delay. Manufacturing Service Oper. Management Forthcoming.

Alcaide, D., J. Sicilia, D. Vigo. (1997) A tabu search algorithm for the open shop problem. TOP (Trabajos de Investigación Operativa), 5 (2) (1997), pp. 283–29.

Alcaide, D., A. Rodriguez-Gonzalez, J. Sicilia. (2006) A heuristic approach to minimize expected makespan in open shops subject to stochastic processing times and failures. Int J Flex Manuf Syst, 17 (3) pp. 201–226.

Baron, O., O. Berman, D. Krass. (2008) Facility Location with Stochastic Demand and Constraints on Waiting Time. *Manufacturing & Service Operations Management* Vol. 10, #3, pp. 484-505.

Baron, O., O. Berman, D. Krass., J. Wang (2013) Using Strategic Idleness to Improve Customer Service Experience in Service Networks. *Operations Research Forthcoming.*

Baron, O., J. Milner. (2009) Staffing to Maximize Profit for Call Centers with Alternate Service Level Agreements. *Operations Research*, 57, pp. 685-700.

Bertsimas, D., D. Gamarnik, J. Sethuraman (2003) From Fluid Relaxations to Practical Algorithms for High-multiplicity Job-shop Scheduling: The Holding Cost Objective, *Operations Research*, 51(5), 798-813.

Bouch, A., A. Kuchinsky, N. Bhatti. (2000) Quality is in the eye of the beholder: Meeting users' requirements for Internet quality of service. In \emph{Proc. of CHI2000 Conference on Human Factors in Computing Systems}, ACM Press, pp. 297-394.

Dai, D., W. Lin (2005) Maximum Pressure Policies in Stochastic Processing Networks, *Operations Research*, 53(2), 197-218.

de-Véricourt F., Y.-P. Zhou (2005) A routing problem for call centers with customer callbacks after service failure. *Operations Research* 53(6) 968-981.

de-Véricourt F., O. Jennings. (2011) Review on Nurse Staffing in Medical Units: A Queueing Perspective. *Operations Research* Vol. 59. No. 6, pp 1320-1331, 1547-1548.

Friedman, H.H., Friedman, L.W. (1997) Reducing the "wait" in waiting-line systems: waiting line segmentation. *Business Horizons*, 40 (4), 54-58.

Gans, N., G. Koole, A. Mandelbaum. (2003) Telephone Call Centers: Tutorial, Review, and Research Prospects. *Manufacturing & service Operations Management*, Vol. 5, No. 2, Spring, pp. 79-141

Harrison, J. M. (1996) The BIGSTEP approach to flow management in stochastic processing networks. F. P. Kelly, S. Zachary, I. Ziedins, eds. Stochastic Networks: Theory and Applications. Clarendon Press, Oxford, U.K., 57-90.

Larson R. C. (1987) Perspectives on Queues: Social Justice and the Psychology of Queueing. *Operations Research* Vol. 35, No. 6 (Nov-Dec), pp. 895-905.

Maglaras, C. (2000) Discrete-review policies for scheduling stochastic networks: Trajectory tracking and fluid-scale asymptotic optimality. *Ann. Appl. Probab.* 10(3) 897-929.

Masin, M., Y. Herer, E. M. Dar-el. (2010) SWIP: A Unified Model of Self-regulating Production Control Systems. *Working paper*, Technion.

Mehrotra, V., K. Ross, G. Ryder, Y.P. Zhou. (2012) Routing to Manage Resolution and Waiting Time in Call Centers with Heterogeneous Servers. *Manufacturing & Service Operations Management* Vol. 14, No. 1, Winter 2012, pp. 66-81.

Ouelhadj, D., S. Petrovic. (2009) A survey of dynamic scheduling in manufacturing systems. J. Scheduling, vol. 12, no. 4, pp.417-431.

Pinedo, M. (1984) A Note on the Flow Time and the Number of Tardy Jobs in Stochastic Open Shops. European Journal of Operational Research, Vol. 18, pp. 81–85.

Pinedo, M., S.M., Ross (1982) Minimizing Expected Makespan in Stochastic Open Shops. Advances in Applied Probability, Vol. 14, pp. 898–911.

Pinedo, M. (2012) Scheduling: Theory, Algorithms, and Systems. Springer, New-York.

Roemer, T.A. (2006) A note on the complexity of the concurrent open shop problem, Journal of Scheduling Vol. 9, pp. 389-396.

Soman, D., M. Shi (2003) Virtual Progress: The effect of path characteristics on perceptions of progress and choice. *Management Science* Vol. 49. No. 9, pp. 1229-1250.

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. *Operations Research* Forthcoming.

Smith W.E. (1956) Various optimizers for single-stage production. *Naval Res. Logist. Quart.* 3 59-66.

Stidham S. (2002) Analysis, Design, and Control of Queueing Systems. *Operations Research* **50**(1) 197-216.

Taylor, S. (1994) Waiting for service: the relationship between delays and evaluation of service. *Journal of Marketing*, 58 (April), 56-69.

Van Mieghem, J.A. (1995) Dynamic Scheduling with Convex Delay Costs: the Generalized $c\mu$ Rule. *Annals of Applied Prob.* 5(3) 809-833.