Scheduling Arrivals to a Stochastic Service Delivery System using Copositive Cones

Qingxia Kong * Chung-Yee Lee † Chung-Piaw Teo ‡ Zhichao Zheng §

Abstract

We develop a convex programming approach to the appointment system design problem in a single server facility, using a (stochastic) network flow model to capture the waiting time performance measure of each patient in the system. We solve a robust min-max problem, using a representative “worst case” distribution matching the prescribed means and covariance estimates of the service durations of the patients, to determine the optimal schedule. Using this approach, the scheduling problem (finding the arrival time of the patients) can be determined by solving a semidefinite programming relaxation.

Our analysis and numerical results yield several interesting insights on the nature of the optimal appointment system design - the scheduling decisions obtained by planning against the “worst case” distribution performed exceedingly well even for several other distributions with the same mean and covariance parameters. Furthermore, for multi-class appointment system, if patients with higher variability are to be seen first before patients with lower variability, the optimal schedule generally exhibits “Bailey’s Rule + break” structure, with first few patients coming “close together” at the beginning, and a higher than usual interval for the last patient in the first class. Our analysis also reveals an interesting characteristic of the optimal appointment system - except for the first and last few slots (those with zero consultation interval allocated to them), the chances of waiting for consultation service is identical for patients choosing all other slots in the system!

1 Introduction

In many service delivery systems, the core operational activities are largely planned around the arrival times of the customers. The ability to regulate the arrival of customers, through a suitable appointment system, is thus central to the performance of these systems. The fastpass service of Disney is a well known example. Customers in the park can obtain a pass to ensure fast track service at certain rides if they return on the stipulated time. The temple of Tirumala

*Department of Decision Sciences, NUS. Email: qingxia@nus.edu.sg
†Department of Industrial Engineering and Logistics Management, HKUST. Email: cylee@ust.hk
‡Department of Decision Sciences, NUS. Email: bizteocp@nus.edu.sg
§Department of Decision Sciences, NUS. Email: zhichao@nus.edu.sg
in India has also used an online appointment system to convert its long waiting line into a virtual queue\(^1\). This has helped improve service delivery and generated spillover economic benefits to businesses in the vicinity of the temple.

The appointment design problem is also a core problem for healthcare facilities such as outpatient clinics and surgery theaters. The appointment system is used to regulate the usage of the costly equipments and precious resources in the system. In one eye-care facility that we have visited, there are two consultation sessions per day, each lasting four hours, and the number of doctors available per session is normally around two to seven. Each doctor has to handle 20 to 30 patients per session. The patients can be classified into “New” (20%) and “Repeat” (80%) patient types. The mean and variance of the consultation times of new patients are noticeably higher than those of repeat patients, as the conditions of the new patients are hitherto unknown prior to the visit. There are also various operational details that complicate the issues. For instance, patients often have to go for dilation test prior to seeing the doctor. This process adds to the complexity of finding an optimal appointment strategy for the system.

The key performance indicator in this system is the “Turnaround Time” (TAT), defined to be the time from the moment the patient walks into the clinic, to the moment the patient leaves the clinic. Figure 1 shows the overall median TAT, service time and waiting time of patients arriving in different time slots for the two different sessions in the clinic. Clearly the patients are experiencing long turn around time, with waiting time far exceeding the actual service time.

![Figure 1: Median Time from Registration to Payment](image)

We note that there are several pertinent features in this system: (i) New patients often have to undergo a series of checks (such as visual acuity, and/or other advanced tests) after the consultation, some of which can take as much as two and half hours. To make sure that

\(^1\)For a thorough discussion, see http://www.iimahd.ernet.in/publications/data/2005-08-02nravi.pdf
all the tests and consultations can be performed within the same day, the doctors prefer to see new patients in the early portion of the morning session. Consequently, early morning slots are reserved primarily for new patients, i.e., new patients are seen before repeat patients in the morning session. (ii) The current appointment strategy is to allocate 5 minutes per slot for one hour, followed by a half hour break to make up for any delay from the packed first hour. This allows each doctor to see around 36 patients in each 4 hour session.

This leads to the central question in this paper: Are there (near) optimal strategy/pattern to schedule the arrival of patients, to minimize the waiting time of the patients and overtime work of the doctor? Furthermore, are there “distributionally robust” solutions that perform well for a wide range of service duration distributions?

The research on appointment system design over the past few decades has been driven largely by these issues. However, these problems are notoriously difficult. Standard queueing theory does not apply as we are interested in the transient performance measures of the system. It is technically challenging to calculate the expected waiting time of the \( n \)th patient in the sequence, due to the difficulty to propagate the impact of earlier events on this patient. Recently, Begen & Queyranne (2009) showed that the scheduling problem is solvable in polynomial time (in the size of the representation of the discrete distributions). Unfortunately, this method works well for discrete distributions with small number of distinct values. To the best of our knowledge, simulation and stochastic programming methods are still the preferred approaches used to tackle the appointment design problem. Unfortunately, the solutions obtained are often sensitive to the samples used to develop the schedules, and hence very little is known about the structure of the optimal policies, even in the simplest environment with one doctor, and when patients arrive punctually.

1.1 Contributions

In this paper, we develop a convex programming approach to solve the appointment scheduling problem. This problem can be suitably reformulated as a two-stage stochastic optimization problem. In the second stage, we construct a (stochastic) network flow model to capture the waiting time characteristics of each patient, given the scheduling policies (decided in the first stage) and the service duration distributions. Our novelty comes in the solution to the first stage problem, which is a technically challenging problem. Instead of using given service duration distributions to design the schedule, we use a “canonical” set of distributions, optimally chosen by our model to maximize some performance measures, at the same time matching the first and second moment conditions prescribed by the system. We therefore develop the schedule that is optimal for this set of canonical distributions for the service durations. This approach allows us to exploit recent advances in the conic programming to transform the stochastic
appointment scheduling problem into a single deterministic copositive program (COP)\(^2\). With a tractable approximation to COP, the scheduling problem can now be solved as a semidefinite programming problem (SDP). Using standard SDP packages, we can now solve practical size appointment scheduling problems.

The schedule obtained using this approach is optimal for a set of canonical service duration distributions, suitably chosen by the model to match the moment conditions and to optimize the expected value of the objective function. Surprisingly, our numerical results show that this schedule also works reasonably well for other service duration distributions with the same moment conditions. This phenomenon is similar to what one would expect in a M/G/1 system, where the expected waiting time is determined purely by the first and second moments of the arrival and service processes, independent of the distributional form of the service duration.

Our approach can be used to obtain interesting structural properties of the optimal schedule. For instance, a pertinent question for the patient is to choose an appointment slot that would minimize the probability of waiting for service. Our analysis shows that when the appointment system is operating under the optimal schedule, other than the first and last few slots (where the consultation intervals allocated are zero or close to zero), the chances of waiting for service is identical for all other slots!

In a congested system with two types of patients, as in our eye clinic case, the optimal schedule often exhibits the pattern: “Bailey’s Rule + Break” - the optimal schedule allocates near zero time slot to the first few patients, which resembles the well known “Bailey’s Rule”, and a break is often inserted before switching from a class of patients with higher variability to another class of patients with lower variability. We use this observation and solution from the SDP model to develop simple practical schedule for the eye clinic. Compared to the naive approach of allocating equal interval to each patient (current practice in the clinic), we find that the total cost of the system can often be reduced by more than 50% in the eye clinic, by a simple modification of the schedule developed using the SDP model. This approach has thus the potential of producing near optimal appointment schedule that can be deployed in practice.

Finally, we extend this approach to incorporate sequencing decision into the appointment design problem. We show that the problem can be solved approximately as a 0-1 SDP problem. Our numerical results give several insights into the structure of the optimal sequencing decisions. In particular, we observed that the optimal sequence need not follows the smallest variance first rule. In fact, in some instances, a U-shape rule is more efficient. This is surprising as it is counter-intuitive to put a high variability patient in front of the queue to minimize the total expected waiting time.

\(^2\)A copositive program is a linear program over the convex cone of copositive matrices. Details of this optimization problem are discussed later in this paper.
2 Literature Review

Since the pioneering work of Bailey (1952), there have been extensive studies on the appointment design problem in the past 50 years. See Cayirli & Veral (2003), Gupta (2007), Gupta & Denton (2008) and Erdogan & Denton (2009) for excellent literature reviews on this line of research.

When patients are homogenous, the issues are simpler since scheduling rules are now the only concern. In practice, however, patients are distinct due to patient’s classification (e.g., new/repeat, ages, types of procedures performed etc.). Patients in different classifications tend to give rise to different means and variability in consultation/service time durations. Higher percentage of more complicated cases (e.g., new patients) normally translates into higher variability in the system performance, and thus proper sequencing of patients become more valuable (cf. Vanden & Dietz (2000) and Cayirli et al. (2008)). Weiss (1990) is arguably the first to study the optimal sequencing problem analytically. He explored the optimal starting time and sequencing of surgical procedures, to best utilize medical resources like surgeons and operating rooms. He showed that sequencing lower-variance procedure first is optimal in the case of 2 procedures under exponential/uniform service time. Weiss also conjectured that the smaller-variance-first rule might be optimal in more complicated systems. Similar results were later reported for local-scale distributions like normal and uniform distributions (cf. Gupta (2007)).

When the sequence of arrival of the patients is given, the scheduling problem reduces to one of finding the optimal consultation intervals to minimize the waiting time of each patient, and to reduce the idle and overwork time of the doctor. Kaandorp & Koole (2007) assumed that the service durations follow the exponential distribution and that the patient arrivals can only be scheduled at discrete intervals. They use results in queueing theory to calculate the objective function for a given schedule of starting times and use a local search algorithm to find the optimal solution. Begen & Queyranne (2009) went a step further and argue that under mild assumptions, the discrete time version of the appointment scheduling problem can be solved in polynomial time, by showing that the objective function is a $L$-convex function.

Given the analytical and computational difficulties of the appointment scheduling problem, we address the issue from a different angle, utilizing the concept of robust optimization. Evolving from the minimax theorem established by John Von Neuman in 1928, the concept was first brought into operations research area by Scarf (1958). Scarf solved an inventory problem with random demand by assuming only the mean and variance of the demand instead of a specific form of distribution. Noting that there could be multiple distributions that satisfies a given mean and variance, Scarf identified a worst case distribution that would result in the highest expected total system cost, and found an inventory strategy to minimize this maximal cost. That is why another popular term describing this concept is called distributionally robust. Such concept has recently been extensively studied and extended, and one stream of the research is to exploit the connection between the theory of moments and semidefinite programming (SDP)
(cf. Bertsimas et al. (2004), Bertsimas et al. (2006), Bertsimas et al. (2008), Vandenberge et al. (2007), etc.). Recently, Natarajan et al. (2009) showed that a robust mix 0-1 linear program under objective uncertainty is equivalent to a convex conic program, which would be helpful in dealing with second stage recourse function in a two stage stochastic programming framework.

3 The Model

To isolate the impact of sequencing and scheduling on the system performance, we rule out the presence of other disruptions in the system. The basic assumptions are listed as follows: (1) Patients arrive punctually at the scheduled appointment times. (2) There is a single doctor in the facility. The doctor arrives punctually and only serves the scheduled patients during the session. No break is taken during the time serving one patient. (3) Patients in the same class are homogenous in the distribution of consultation durations. (4) Walk-in and emergencies are not considered.

Let $\mathcal{U}_i$ denote the random consultation time of patient $i$ and assume that $\mathcal{U}_i$ follows a distribution with mean $\mu_i$ and standard deviation $\sigma_i$. $\mathcal{W}_i$ denotes the waiting time of the $i^{th}$ patient in the sequence. It is reasonable to assume that the session starts at time zero, i.e., $\mathcal{W}_1 = 0$. Let $s_i$ denote the consultation interval allocated to the $i^{th}$ patient in this sequence, and

$$Z_i = \mathcal{U}_i - s_i, \quad i = 1, \ldots, n.$$ 

$Z_i$ is the difference of the actual consultation time and the allocated consultation interval of the $i^{th}$ patient in the sequence.

The waiting time of subsequent patients are given by the following recursions (c.f. Denton & Gupta (2003)):

$$W_i = \max\{0, W_{i-1} + Z_{i-1}\}. \quad (1)$$

Let $\mathcal{O}$ denote doctor’s overtime. If there is an additional patient $i = n + 1$ arriving after the $n$ patients, then the waiting time of this patient is the doctor’s overtime, i.e.,

$$\mathcal{O} = W_{n+1} = \max\{0, W_n + Z_n\}. \quad (2)$$

3 This assumption can be relaxed. In the paper, we demonstrate how to incorporate late/early arrivals and no-shows into our model as the extension.

4 Note that in a typical appointment scheduling problem, it is common for patients to choose the appointment slots in a dynamic fashion, and their characteristics are known only at the time of booking. The model described above matches more the surgery scheduling environment. However, in certain appointment scheduling environment, patients are classified into distinct classes and each appointment slot in a single clinical session is pre-assigned to a dedicated class of patients. The slots are filled up when patients call in for appointment and their classifications are revealed. We assume that the clinic has enough volume to fill up the slots available in each day. In this way, the scheduling problem described here essentially addresses the design of the appointment system based on patient classifications, but not on the characteristics of individual patients.
In this paper, we will use the total patient’s waiting time and doctor’s overtime, i.e.,
\[ \sum_{k=1}^{n} W_k \]
and
\[ W_{n+1} \]
as the key performance indicators of the appointment system. In this way, the waiting time of each patient in the sequence can be written as
\[ W_i = \max \left\{ 0, Z_{i-1}, Z_{i-1} + Z_{i-2}, \cdots , \sum_{k=1}^{i-1} Z_k \right\} . \quad (3) \]

The objective of the appointment scheduling problem is to minimize the expectation of the weighted sum of the patients’ waiting times and the doctor’s overtime, i.e.,
\[ \mathbb{E} \left[ \sum_{i=1}^{n} w_i W_i + w_{n+1} W_{n+1} \right] , \]
where \( w_i, i = 1, 2, \ldots , n + 1 \) are corresponding weights. Unless otherwise stated, we assume throughout that \( w_i = 1 \) for all \( i = 1, \ldots , n + 1 \).

This problem is challenging, in part because of the technical difficulty associated with the computation of
\[ \mathbb{E} |W_i| = \mathbb{E} \left[ \max \left\{ 0, Z_{i-1}, Z_{i-1} + Z_{i-2}, \cdots , \sum_{k=1}^{i-1} Z_k \right\} \right] . \]

4 Copositive Cone Programming

In this section, we show that the appointment scheduling problem can be formulated as a two-stage stochastic optimization problem. In the first stage, the appointment scheduling decisions are made with the objective to minimize the expected total waiting time cost. In the second stage, the patients’ service durations are realized and the system performance is determined.

4.1 The Copositive Cone Formulation

Let \( N = \{1, 2, \ldots , n\} \) be the set of all patients, and \( U_i \) be the random service time of patient \( i, i = 1, 2, \ldots , n \). We define \( s = \{s_1, s_2, \cdots , s_n\}^T \), where \( s_i \) represents the length of time slots scheduled for \( i^{th} \) patient in the sequence. Therefore, the appointment time of the patients in the sequence is given by \( \{0, s_1, s_1 + s_2, \ldots , \sum_{i=1}^{n-1} s_i\} \).

\(^5\)Note that our results can be easily extended to include doctor’s idle time as well, since the consultation interval \( T \) is pre-determined. The total idle time is thus \( T + O - \sum U_i \).

\(^6\)In the rest of the paper, the phrase “total waiting time (cost)” means “waiting time (costs) of all customers and the overtime (cost)”. 

7
4.1.1 Second Stage Problem

Given the schedule of the customers (i.e., \( s \) is known), the total waiting time can be computed via solving a network flow problem on a directed graph shown in Figure 2. The cost on arc \((i, s)\) is 0, whereas the cost on arc \((i + 1, i)\) is \( c_i(s) = Z_i = U_i - s_i \), and the capacity for all the arcs are infinite.

![Figure 2: Network flow representation of appoint scheduling problem](image)

Let \( y_i, i = 1, 2, \ldots, n \), be the flows on arc \((i + 1, i)\), and \( z_i, i = 1, 2, \ldots, n + 1 \) be the flows on arc \((i, s)\). Then we have the following proposition.

**Proposition 1** Given the schedule, the optimal cost of the following maximum cost flow problem equals the total waiting time of the system:

\[
\begin{align*}
    f(s) = & \max \sum_{i=1}^{n} c_i(s) \cdot y_i \\
    \text{s.t.} & \quad y_i - z_1 = -1 \\
                 & \quad y_i - y_{i-1} - z_i = -1, \forall i = 2, 3, \ldots, n \\
                 & \quad -y_n - z_{n+1} = -1 \\
                 & \quad y_i \geq 0, \forall i = 1, 2, \ldots, n \\
                 & \quad z_i \geq 0, \forall i = 1, 2, \ldots, n + 1
\end{align*}
\]

**Proof.** Recall Equation (3), the waiting time of the \( i^{th} \) patient in the appointment system is given by

\[
W_i = \max \left\{ 0, Z_{i-1}, Z_{i-1} + Z_{i-2}, \ldots, \sum_{k=1}^{i-1} Z_k \right\}.
\]

In the optimal network flow solution, the unit supply from node \( i \) will find a path to destination \( s \), by maximizing the flow cost among the paths

\[
(i \to s), (i \to i - 1 \to s), \ldots, (i \to i - 1 \to \ldots 1 \to s).
\]
Hence the flow cost attained by the supply from node $i$ is just $W_i$. 

Finally, by removing redundant constraints if necessary, we rewrite $f(s)$ using matrix notations for the ease of exposition:

$$f(s) = \max_{y, z \geq 0} c^T(s) y$$

subject to:

$$a^T_j y - e^T_j z = -1, \forall j = 1, 2, \ldots, n$$

where $c(s) = (c_1(s), c_2(s), \ldots, c_n(s))^T$, $y = (y_1, y_2, \ldots, y_n)^T$, and $z = (z_2, z_3, \ldots, z_{n+1})^T$; and $e_j \in \mathbb{R}^n$ is the unit vector with its $j^{th}$ entry being one; and

$$\begin{pmatrix}
  a^T_1
  
  a^T_2
  
  \vdots
  
  a^T_n
\end{pmatrix} = \begin{pmatrix}
  -1 & 1 & 0 & \ldots & 0 & 0 \\
  0 & -1 & 1 & \ldots & 0 & 0 \\
  : & : & : & \ddots & : & : \\
  0 & 0 & 0 & \ldots & -1 & 1 \\
  0 & 0 & 0 & \ldots & 0 & -1
\end{pmatrix} \in \mathbb{R}^{n \times n}.
$$

When the service durations become stochastic, with given moment conditions, the worst case expected total waiting time can be written as:

$$(P) \quad \mathbb{Z}_P = \sup_{U \sim (\mu, \Sigma)^+} \{E[f(s)]\}$$

where $U \sim (\mu, \Sigma)^+$ denotes the set of feasible multivariate distributions supported on $\mathbb{R}_+^n$ with finite first moment $\mu$ and finite second moment $\Sigma$, and this set is assumed to be nonempty. The challenge to solve $(P)$ reduces to the following: can one find a set of random variables $c$ in such a way that

$$c \geq 0 \text{ with probability 1, } \quad E[c] = \mu, \quad E[cc^T] = \Sigma,$$

and a corresponding optimal solution $x(c)$ which is feasible solution to $f(s)$ in $(P)$, such that $E[c^T x(c)]$ attains the maximum $\mathbb{Z}_P$? In general, if the maximum cannot be attained, can one find a limiting set of random variables so that $\mathbb{Z}_P$ is attained asymptotically?

4.1.2 Transformation to copositive cone

Before showing the main theorem, we introduce some necessary notations and briefly review the related subjects. A more detailed review can be found in Appendix A.

**Notations**

The trace of a matrix $A$, denoted by $tr(A)$, is the sum of diagonal entries of matrix $A$. The inner product between two matrices of appropriate dimensions $A$ and $B$ is denoted as $A \bullet B = tr(A^T B)$. $I_n$ represents the identity matrix of dimension $n \times n$, while $O_n$ represents the zero matrix of dimension $n \times n$. And $0$ is used to denote the zero vector of appropriate dimension. For any convex cone $K$, the dual cone is denoted as $K^*$. $S_n$ denotes the cone of $n \times n$ symmetric
matrices, and $\mathcal{S}_n^+$ denotes the cone of $n \times n$ positive semidefinite matrices. $A \succeq 0$ indicates that matrix $A$ is positive semidefinite and $B \succeq A$ indicates $B - A \succeq 0$. Two cones of special interest are the cone of completely positive matrices and the cone of copositive matrices. The cone of $n \times n$ completely positive matrices is defined as:

$$\mathcal{CP}_n := \left\{ A \in \mathcal{S}_n \mid \exists V \in \mathbb{R}^{n \times k}_+, \text{ such that } A = VV^T \right\}.$$  

The cone of $n \times n$ copositive matrices is defined as:

$$\mathcal{CO}_n := \left\{ A \in \mathcal{S}_n \mid \forall v \in \mathbb{R}^n_+, \ v^T Av \geq 0 \right\}.$$  

$A \succeq_{cp}\ (\succeq_{co}) 0$ indicates that matrix $A$ is completely positive (copositive). These two cones are dual of each other, i.e.,

$$\mathcal{CP}_n^* = \mathcal{CO}_n, \text{ and } \mathcal{CO}_n^* = \mathcal{CP}_n.$$  

It is well known that there exists a hierarchy of linear and semidefinite representable cones that approximate the copositive and completely positive cones (cf. Bomze et al. (19952), Klerk et al. (2002), Parrilo (2000)). In this paper, we restrict our attention to the simple relaxations of CPP and COP for the numerical experiments:

$$\begin{cases}  
A \succeq_{cp} 0 \approx A \succeq 0, \text{ and } A \succeq 0, \\
A \succeq_{co} 0 \approx \exists A_1 \succeq 0, \text{ and } A_2 \succeq 0, \text{ such that } A = A_1 + A_2. 
\end{cases}$$  

(CPP) min $C \cdot X$

s.t. $A_i \cdot X = b_i, i = 1, 2, \ldots, m$

$X \succeq_{cp} 0$

The dual of the above CPP is

(COP) max $\sum_{i=1}^{m} b_i y_i$

s.t. $C - \sum_{i=1}^{m} y_i A_i \succeq_{co} 0$

Natarajan et al. (2009) showed the following fundamental result:

**Theorem 1** [Natarajan et al. (2009)] The stochastic optimization problem:

$$Z_1 = \sup_{\hat{c} \sim (\mu, \Sigma)^+} \mathbb{E}[Z(\hat{c})]$$

where

$$Z(\hat{c}) = \max \ \hat{c}^T x$$

s.t. $a_i^T x = b_i, \forall i = 1, 2, \ldots, m$

$x \geq 0$

can be reformulated as a CPP:
apply this result directly, since in our model the random objective coefficients \( \tilde{c} \) correspond optimal solution \( \tilde{x} \). Completely Positive Cross Moment Model. It produces a set of scenarios, i.e., \( Z \). The fundamental result. The scheduling decisions, i.e., \( Z \). Define a new random vector \( \tilde{x} \). When the random vector \( \tilde{c} \) is bounded below, i.e., \( \exists 0 < \tilde{c} < \infty \) such that \( Pr(\tilde{c} + \tilde{x} \geq 0) = 1 \). For more detailed discussion on the definition of \( x_j(\tilde{c}) \) and the complete treatment of multiple optimal issues, please refer to Natarajan et al. (2009)

\[
Z_2 = \sup \quad tr(Y) \\
\text{s.t.} \quad \tilde{a}_i^T\tilde{x} = b_i, \forall i = 1, 2, \ldots, m \\
\tilde{a}_i^TX\mu = b_i^2, \forall i = 1, 2, \ldots, m \\
\begin{pmatrix}
1 & \mu^T & x^T \\
\mu & \Sigma & Y^T \\
x & Y & X
\end{pmatrix} \succeq 0
\]

\( i.e., Z_1 = Z_2 \).

The matrices \( Y \) and \( X \) attempt to encode the relationship \( Y_{ji} = E[c_j x_j(\tilde{c})] \) and \( X_{ij} = E[x_i(\tilde{c}) x_j(\tilde{c})] \), where \( x_j(\tilde{c}) \) denotes the value of \( x_j \) in an optimal solution to \( Z(\tilde{c}) \) under a specific realization of \( \tilde{c} \). The constraints \( \tilde{a}_i^TX\mu = b_i^2 \) arises from the observation that

\[
E[\tilde{a}_i^T(x(\tilde{c})x(\tilde{c})^T)a_i] = b_i^2,
\]

and the completely positive constraint arises from

\[
\begin{pmatrix}
1 & \mu^T & x^T \\
\mu & \Sigma & Y^T \\
x & Y & X
\end{pmatrix} = E\begin{pmatrix}
1 & \tilde{c}_i^T & x(\tilde{c})^T \\
\tilde{c}_i & \tilde{c}\tilde{c}_i^T & \tilde{c}x(\tilde{c})^T \\
x(\tilde{c}) & x(\tilde{c})\tilde{c}_i^T & x(\tilde{c})x(\tilde{c})^T
\end{pmatrix} = E\begin{pmatrix}
1 & \tilde{c}_i^T \\
\tilde{c}_i & x(\tilde{c})^T
\end{pmatrix}^T \succeq 0.
\]

The objective function \( E[\tilde{c}^T x(\tilde{c})] \) reduces to \( tr(Y) \) in this case.

Miraculously, the lifted constraints \( \tilde{a}_i^TX\mu = b_i^2 \), and the intersection with completely positive cone are sufficient to characterize the solution to the original stochastic optimization problem \( Z_1 \). The proof hinges on the following decomposition result:

If we let \( (x^*, Y^*, X^*) \) be an optimal solution to \( Z_2 \), then there exists a sequence of nonnegative random objective coefficient vectors \( \tilde{c}_\epsilon^* \) and feasible solutions \( x^*(\tilde{c}_\epsilon^*) \) that converge in moments to:

\[
\lim_{\epsilon \to 0} E\left[ \begin{pmatrix}
1 & \tilde{c}_\epsilon^* \\
x^*(\tilde{c}_\epsilon^*) \\
\tilde{x}^*(\tilde{c}_\epsilon^*)
\end{pmatrix} \begin{pmatrix}
1 & \tilde{c}_\epsilon^* \\
x^*(\tilde{c}_\epsilon^*) \\
\tilde{x}^*(\tilde{c}_\epsilon^*)
\end{pmatrix}^T \right] = \begin{pmatrix}
1 & \mu^T & x^*T \\
\mu & \Sigma & Y^*T \\
x^* & Y^* & X^*
\end{pmatrix}.
\]

The above optimization model to compute \( Z_2 \) is denoted as CPCMM, which stands for Completely Positive Cross Moment Model. It produces a set of scenarios \( c_\epsilon^* \) and the corresponding optimal solution \( x_\epsilon^* \), matching the moment conditions. Unfortunately, we could not apply this result directly, since in our model the random objective coefficients \( \tilde{c} \) is a function of the scheduling decisions, i.e., \( c_i(s) = U_i - s_i \). Hence the moment matrix \( \mu \) and \( \Sigma \) are variables in our formulation. To overcome this difficulty, we establish an important extension to this fundamental result.

When the random vector \( \tilde{c} \) is bounded below, i.e., \( \exists 0 < \tilde{c} < \infty \) such that \( Pr(\tilde{c} + \tilde{x} \geq 0) = 1 \). Define a new random vector \( c_+ = \tilde{c} + \tilde{x} \), which is nonnegative, and \( \mu_+ = E[c_+] = \mu + \tilde{c}, \Sigma_+ = E[c_+^Tc_+] = \Sigma + \mu\mu^T + \tilde{c}\tilde{c}^T \).
By mimicking the proof to Theorem 1, and exploiting the decomposition result of the optimal solution, we can prove that

**Proposition 2** With the notations defined above, the following stochastic optimization problem

\[
Z_1' = \sup_{c_+ \sim (\mu_+, \Sigma_+)^+} E[Z(c_+ - \bar{c})]
\]

where

\[
Z(c_+ - \bar{c}) = \max_{x \geq 0} (c_+ - \bar{c})^T x
\]

s.t. \( a_i^T x = b_i, \forall i = 1, 2, \ldots, m \)

is equivalent to

\[
Z_2' = \sup_{Y} tr(Y) - \bar{c}^T x
\]

s.t. \( a_i^T x = b_i, \forall i = 1, 2, \ldots, m \)

\[
\begin{pmatrix}
1 & \mu_+^T x^T \\
\mu_+ & \Sigma_+ Y^T \\
x & Y & X
\end{pmatrix} \succeq_{cp} 0
\]

i.e., \( Z_1' = Z_2' \).

To apply the above results on the appointment design problem, we incorporate decision variables \( s \) into the model in the following way:

**Theorem 2** \((P)\) can be solved as the following completely positive program, i.e., \( Z_P = Z_C \):

\[
(C) \quad Z_C = \max_{y, z} tr(X) - s^T y
\]

s.t. \( \begin{pmatrix}
a_j & y^T \\
-z & z^T
\end{pmatrix} = -1, \forall j = 1, 2, \ldots, n \)

\( \begin{pmatrix}
a_j & y^T \\
-z & z^T
\end{pmatrix} = 1, \forall j = 1, 2, \ldots, n \)

\( \begin{pmatrix}
Y & W^T \\
W & Z
\end{pmatrix} \succeq_{cp} 0 \)

where the decision variables are \( y, z \in \mathbb{R}^n, Y, W, Z, X, V \in \mathbb{R}^{n \times n} \).

This formulation models the expected waiting cost of the appointment system, when the scheduling decision \( s \) are fixed. Note that these decisions are used only in the objective function, which will be convenient for our dual reformulation. We offer a sketch of the proof to Theorem 2. The proof to Proposition 2 can be constructed using similar argument.

**Proof.** First of all, note that all the assumptions in CPCMM (except nonnegative support assumption) are trivially satisfied in our problem, i.e., bounded and nonempty feasible region. **Step 1:** \( Z_P \leq Z_C \), i.e., \( (C) \) is a relaxation of \( (P) \).
We show next that we could treat (C) as a relaxation of (P) by a suitable interpretation of the decision variables in (C). Thus $Z_P \leq Z_C$.

With a slight abuse of notation, $\forall i = 1, 2, \ldots, n$, define $y_i(U)$ to be the value of the variable $y_i$ in an optimal solution to $f(s, \sigma)$ obtained under a specific realization of $U$. When $U$ is random, $y_i(U)$ is also a random variable. Similar definitions apply to $z_i(U)$, $i = 1, 2, \ldots, n$.

The decision variables in (C) can be interpreted as:

\[
\begin{align*}
    y &= \mathbb{E}[y(U)], \\
    z &= \mathbb{E}[z(U)], \\
    Y &= \mathbb{E}[y(U)y(U)^T], \\
    W &= \mathbb{E}[y(U)z(U)^T], \\
    Z &= \mathbb{E}[z(U)z(U)^T], \\
    X &= \mathbb{E}[y(U)U^T], \\
    V &= \mathbb{E}[z(U)U^T].
\end{align*}
\]

With these interpretations, the constraints in (C) are simply the necessary conditions for the constraints in (P), and the objective of (C) becomes:

\[
X \cdot I_n - s^T y = \mathbb{E}[y(U)U^T] \cdot I_n - s^T \mathbb{E}[y(U)] = \mathbb{E}[U^T y(U)] - s^T \mathbb{E}[y(U)] = \mathbb{E}[c(s)^T y(U)] = \mathbb{E}[f(s)].
\]

To conclude, (C) is obtained from (P) using the same objective function but only keeping some necessary conditions for the feasibility conditions in (P). Hence, (C) is a relaxation of (P).

**Step 2:** $Z_P \geq Z_C$, i.e., (C) is a tight relaxation of (P).

By the decomposition theorem, we can find a sequence of nonnegative random objective coefficient vectors $\tilde{c}^*_e$ and feasible solutions $y^*(\tilde{c}^*_e)$ and $z^*(\tilde{c}^*_e)$ that converge in moments to:

\[
\lim_{\epsilon \downarrow 0} \mathbb{E} \left[ \begin{pmatrix} 1 \\ \tilde{c}^*_e \\ y^*(\tilde{c}^*_e) \\ z^*(\tilde{c}^*_e) \end{pmatrix} \right]^T \begin{pmatrix} 1 \\ \mu^T \\ \Sigma \\ X^* \\ Y^* \\ V^* \\ W^* \\ Z^* \end{pmatrix} = \begin{pmatrix} 1 \\ \mu^T \\ \Sigma \\ X^* \\ Y^* \\ V^* \\ W^* \\ Z^* \end{pmatrix}.
\]

Hence

\[
Z_C = tr(X^*) - s^T y^* = \lim_{\epsilon \downarrow 0} \mathbb{E} \left[ (\tilde{c}^*_e - s)^T y^*(\tilde{c}^*_e) \right] \leq \sup_{U \sim (\mu, \Sigma)^+} \{ \mathbb{E}[f(s)] \} = Z_P.
\]

We conclude that $Z_P = Z_C$. 

\[\square\]
4.1.3 First Stage Problem

We have shown that the second stage problem can be transformed into a completely positive program. Now back to the first stage, we want to obtain a schedule $\mathbf{s}$ such that the worst case expected total waiting time is minimized, i.e., we want to solve

$$
\min_{\mathbf{s}} \left\{ \sup_{U \sim (\mu, \Sigma)^+} \{ \mathbb{E}[f(\mathbf{s})] \} \right\}
$$

(5)

Knowing that the inner part of the above problem is a single maximization CPP given in (C), we can get the dual of this CPP, which is a minimization COP. Details of constructing the dual can be found in Appendix C. After that, Problem (5) can be consolidated as a single COP in a minimization form as follows$^8$:

$$(S^2) \quad Z_{S^2} := \min \left( \Sigma \bullet \Gamma + \mathbf{\mu}^T \mathbf{\beta} + \alpha \right)
$$

subject to

$$
\begin{bmatrix}
\sum_{j=1}^{n} (-u_j + v_j) + \alpha \\
\frac{\beta}{2} \\
\left( \frac{s}{2} \right) - \sum_{j=1}^{n} \left( \begin{array}{c}
\mathbf{a}_j \\
-\mathbf{e}_j
\end{array} \right)
\end{bmatrix}
\end{bmatrix}
\begin{bmatrix}
\beta \\
\Gamma \\
\left( \begin{array}{c}
-I_n/2 \\
O_n
\end{array} \right)
\end{bmatrix}
\geq_c 0
$$

and $\mathbf{s} \in \Omega_s$.

where the decision variables are $\alpha \in \mathbb{R}, \beta \in \mathbb{R}^n, \Gamma \in \mathbb{R}^{n \times n}, \mathbf{u} \in \mathbb{R}^n, \mathbf{v} \in \mathbb{R}^n$. The last constraint confines the choice of $\mathbf{s}$ to a feasible set $\Omega_s$. For example, $\mathbf{s} \in \Omega_s$ in our case is

$$
\sum_{i=1}^{n} s_i \leq \mathcal{T}, \text{ and } s_i \geq 0, \forall i = 1, 2, \ldots, n
$$

(6)

which means the time slots must be nonnegative and the total scheduled time cannot exceed the session time $\mathcal{T}$.

We have thus obtained the central result in this paper:

**Theorem 3**

$$
\min_{\mathbf{s}} \left\{ \sup_{U \sim (\mu, \Sigma)^+} \{ \mathbb{E}[f(\mathbf{s})] \} \right\} = Z_{S^2}
$$

(7)

---

$^8$Without loss of generality, we assume the Slater’s constraint qualification is satisfied. Hence the strong duality holds. For the technical details regarding the Slater’s conditions for conic optimization problems, the audience can refer to Boyd & Vandenberghe (2004).
Using the approximation
\[ A_{\geq 0} \approx \exists A_1 \geq 0, \text{ and } A_2 \geq 0, \text{ such that } A = A_1 + A_2, \]
we can solve a relaxation of \( Z_{S2} \) as semidefinite programming problem. The compact formulation of \( Z_{S2} \) allows us to study the structure of the optimal policy both analytically and numerically.

4.2 Extensions

4.2.1 General Waiting Time Costs

In the earlier discussion, we have assumed \( w_i = 1 \) for each patient \( i \). The network flow model used in the second stage problem can be extended to cope with general waiting time cost \( w_i \). This can be achieved by simply changing the in-flow at each node \( i \) from 1 to \( w_i \), and the out-flow at node \( s \) from \( n + 1 \) to \( \sum_{i=1}^{n+1} w_i \). The reader can easily verify that the total waiting time cost is now mapped to the maximum cost flow problem in the network with the new parameters.

4.2.2 Eye test before consultation

Suppose that the \( i^{th} \) patient in the sequence has to undertake a test prior to the consultation. The test is often handled by a nurse and can be administered immediately upon arrival. The duration of the test is random and denoted by the random variable \( L_i \). We define the waiting time of the patients to be the waiting time needed to consult the doctor after the test is administered. We assume also that the patients are seen by the doctor in the same sequence based on the appointment time, i.e., the sequence of the patients seen by the doctors is the same as the sequence of arrival. In this case, we can also use the network flow model to capture the impact of the test on the performance of the system. This is achieved by changing the cost on arcs \((i, s)\), \( i = 1, 2, \ldots, n \), from 0 to the random variables, \( L_i \). The readers can easily verify that the network flow solution in our model corresponds to the total waiting cost in the system, offset by \( \sum_{i=1}^{n} L_i \), i.e.,

\[
 f(s) = \max_{y, z} \sum_{i=1}^{n} c_i(s) \cdot y_i + \sum_{i=1}^{n} L_i z_i - \sum_{i=1}^{n} L_i \\
\text{s.t. } y_1 - z_1 = -1 \\
y_i - y_{i-1} - z_i = -1, \forall i = 2, 3, \ldots, n \\
y_n - z_{n+1} = -1 \\
y_i \geq 0, \forall i = 1, 2, \ldots, n \\
z_i \geq 0, \forall i = 1, 2, \ldots, n + 1
\]

To see this, note that when \( z_i = 1 \), then the \( i^{th} \) patient finishes the eye test and finds the doctor to be idling. This patient gets to consult the doctor at time \( L_i \) after arrival. The waiting time is thus zero. This starts a new busy period, with the initial consultation duration given
by \( \mathcal{L}_i + c_i(s) \). On the other hand, if after the test, the patient finds the doctor to be busy, then \( z_i = 0 \) in the network flow solution, and hence the waiting time is simply the length of the longest path originating from node \( i \) deducted by \( \mathcal{L}_i \).

### 4.2.3 Relationship to Scenario Planning

In our model, we assume that only the moments and covariance parameters are known. The model constructs a set of scenarios, the associated probability function, and a solution which attains the (worst case) performance objective under this set of scenarios. This approach can be augmented easily to include specific scenarios in the worst case decomposition. More specifically, suppose that the planner would like to construct the optimal schedule under the additional restriction to include the scenarios \( u^L \) with probability \( p_L \), such that \( \sum_{L=1}^{N} p_L = p < 1 \). Furthermore, the conditional mean and covariance parameters for the remaining scenarios are denoted by \( (\mu, \Sigma)^+ \). Our model reduces to

\[
Z_P = (1 - p) \sup_{U \sim (\mu, \Sigma)^+} \{ \mathbb{E}[f(s)] \} + \sum_{L=1}^{N} p_L f_L(s)
\]

where

\[
f_L(s) = \max \sum_{i=1}^{n} (u_i^L - s_i) \cdot y_i^L
\]

\[
\text{s.t. } y_i^L - z_i^L = 1 \\
y_i^L - y_{i-1}^L - z_i^L = -1, \forall i = 2, 3, \ldots, n \\
y_{n+1}^L - z_{n+1}^L = -1 \\
y_i^L \geq 0, \forall i = 1, 2, \ldots, n \\
z_i^L \geq 0, \forall i = 1, 2, \ldots, n + 1
\]

In this way, we use a small set of scenarios to ensure that the optimal solution constructed will not perform too badly for these typical scenarios, and hence will not be overly conservative. Note that the dual to the above second stage problem can be written using the approach described earlier, together with standard linear programming duality.

### 5 Model Analysis

Our model provides a single deterministic convex formulation to solve a two stage stochastic optimization problem. To the best of our knowledge, this model is the first of such kind. Furthermore, since the formulation of (C) is tight, and from the proof of Theorem 2, the optimal solution to (C) have natural probabilistic interpretation under the worst case distribution. Note that we can obtain the values of those variables in (C) by taking the dual of \((S^2)\). Together with the network flow intuitions, they provide us a rich resource to analytically obtain the insights into the structure of optimal appointment policy.
Before presenting the analysis, we first define the necessary dual variables of \((S^2)\). Let \(M\) be the dual variables of the copositivity constraint. Note that \(M\) is exactly the completely positive matrix in \((P)\), and in particular,

\[
\begin{aligned}
M_{1,n+2} &= y_1, & M_{1,n+3} &= y_2, & \ldots, & M_{1,2n+1} &= y_n, \\
M_{1,2n+2} &= z_2, & M_{1,2n+3} &= z_3, & \ldots, & M_{1,3n+1} &= z_{n+1}.
\end{aligned}
\]

Denote the dual variables of the constraints in \((6)\) by \(\theta\) and \(\lambda_i\), where \(\theta\) corresponds to the total session time limit constraint, whereas \(\lambda_i\) corresponds to the non-negativity constraint for \(s_i\).

Without loss of generality, We assume the constraints qualifications (in particular, the Slater’s condition) are satisfied. Thus the KKT conditions are both necessary and sufficient in characterizing the optimal solutions.

### 5.1 Structure of Optimal Schedules

In this section, we discuss some findings on the structure of the optimal schedules. We show first that if there is a need to bunch the arrival of patients together, then it is optimal to bunch the arrivals at the end of the session.

**Proposition 3** When the waiting time costs and overtime cost are strictly positive, in the optimal solution to Problem \((S^2)\), if the allocated service time slots are zero, \((i.e., s_i = 0, \forall i \in I \subseteq 1,2,\ldots,n)\), then the index \(i\) must be the last \(|I|\) numbers in \(\{1,2,\ldots,n\}\), i.e. \(I = \{n - |I| + 1,\ldots,n - 1, n\}\), where \(|I|\) is the cardinality of set \(I\).

**Proof.** The proof is based on the KKT conditions and the network structure in Figure 2. Assume in the optimal solution, \(s_i = 0\) and \(s_{i+1} > 0\) for some \(i \in \{1,2,\ldots,n - 1\}\). Then the cost on arc \((i+1,i)\) in the network is \(c_i = U_i - s_i = U_i \geq 0\). Due to the nature of maximal cost flow problem, any flow entering node \(i + 1\) will choose arc \((i+1,i)\) instead of arc \((i,s)\) whose cost is zero in any situations. Then we have \(z_{i+1}(U) = 0\) for any realization of \(U\), and consequently \(E[z_{i+1}] = 0\), i.e., in the optimal solution to Problem \((S^2)\), \(z_{i+1} = 0\).

Recall that \(w_i\) is the weight of the waiting time cost of the \(i^{th}\) patient in the sequence. From the following KKT conditions:

\[
\begin{aligned}
\lambda_i s_i &= 0, \\
\lambda_{i+1} s_{i+1} &= 0, \\
\lambda_{i+1} &\geq 0, \\
y_i &= \theta - \lambda_i, \\
y_{i+1} &= \theta - \lambda_{i+1}, \\
w_{i+1} + y_{i+1} &= y_i + z_{i+1},
\end{aligned}
\]

we get

\[
\begin{aligned}
s_{i+1} > 0 &\implies \lambda_{i+1} = 0 \\
&\implies y_{i+1} = \theta - \lambda_{i+1} = \theta \\
&\implies y_i = w_{i+1} + y_{i+1} - z_{i+1} = w_{i+1} + \theta.
\end{aligned}
\]
Since $y_i = \theta - \lambda_i$, we have

$$w_{i+1} + \theta = \theta - \lambda_i$$

$$\implies \lambda_i = -w_{i+1} < 0,$$

which contradicts $\lambda_i \geq 0$. Hence, the result follows.

The next proposition shows that the non-negativity constraint on the consultation slots can be removed without loss of generality, except for the last slot.

**Proposition 4** Suppose the consultation slots are allowed to be negative, i.e., the second set of constraints in (6) are removed. When the waiting time costs and overtime cost are strictly positive, in the optimal solution to Problem $(S^2)$, there is at most one negative schedule. Furthermore, if this negative schedule exist, it must be the last one, i.e., $s_i \geq 0, \forall i = 1, 2, \ldots, n-1$, and $s_n < 0$.

**Proof.** By a similar proof as in Proposition 3, all the negative schedules should be at the end. Hence, we only need to prove there is only one such schedule, which is $s_{n+1}$.

Assume in an optimal schedule, denoted by $s^{(1)}$, there are at least two negative schedules, i.e., $s^{(1)}_{n-1} < 0$ and $s^{(1)}_n < 0$. Consider a new schedule, $s^{(2)}$ defined as

$$s^{(2)}_i = s^{(1)}_i, \forall i = 1, 2, \ldots, n-2,$$

$$s^{(2)}_{n-1} = 0,$$

$$s^{(2)}_n = s^{(1)}_{n-1} + s^{(1)}_n.$$

Let $TC^{(1)}(U)$ and $TC^{(2)}(U)$ be the total waiting time cost under a specific service duration $U$ for the schedule $s^{(1)}$ and $s^{(2)}$, respectively. Note for any $U$, $U^{(k)}_{\sigma(n-1)} - s^{(k)}_{n-1} \geq 0$, and $U^{(k)}_{\sigma(n)} - s^{(k)}_n \geq 0$, $k = 1, 2$. Then considering the input of the last two nodes, i.e., $w_n$ entering node $n$ and $w_{n+1}$ entering node $n+1$, we get

$$TC^{(1)}(U) - TC^{(2)}(U) = w_n(U_{\sigma(n-1)} - s^{(1)}_{n-1}) + w_{n+1}(U_{\sigma(n)} - s^{(1)}_n + U_{\sigma(n-1)} - s^{(1)}_{n-1} - \{w_n(U_{\sigma(n-1)} - s^{(1)}_{n-1}) + w_{n+1}(U_{\sigma(n)} - s^{(1)}_n + U_{\sigma(n-1)} - s^{(1)}_{n-1})\})$$

$$= -w_n s^{(1)}_{n-1} > 0.$$

Thus, $s^{(1)}$ should not be optimal, and we reach a contradiction.

**Remark 1** The negative schedule will appear in the optimal solution if the session time is too tight such that the overtime is almost unavoidable. Under this situation, if the total waiting time cost is important, it is better to schedule some customers to arrive after the end of the session instead of squeezing them in the over-congested session. From another perspective, the appearance of negative schedule in the optimal solution is a strong signal to the system designer that extending the session time should be considered.
5.2 Optimality Conditions

What are the desired features in the optimal schedule? Are there universal properties that any optimal schedule in the appointment system must satisfy? In this section, we answer these questions. We point out next an important class of necessary conditions in any optimal schedule. The results are derived from the optimality conditions of the Problem (S\textsuperscript{2}) as well as the network structure shown in Figure 2.

Before proving the optimality condition, we need two preliminary results, the first of which deals with predicting the probabilistic performance of the system under the optimal sequencing and scheduling decisions obtained from solving Problem (S\textsuperscript{2}). Recall that \( w_i \) is the waiting time cost of the \( i \)-th patient.

**Proposition 5** When the \( i \)-th patient enters the clinic, the probability that she has to wait for service is given by

\[
Pr\{i^{th} \text{ patient has to wait}\} = \frac{y_{i-1}}{y_i + w_i}, \forall i = 2, 3, \ldots, n.
\]

**Proof.** From Figure 2, the flow \( y_i \) merges with \( w_i \) at node \( i \). The probability that this combined flow will go through arc \( (i, i-1) \) is exactly the probability that the \( i \)-th customer has to wait. Otherwise, the flow on arc \( (i, i-1) \) would be zero, which indicates that the waiting time cost is zero for the \( i \)-th patient since arc \((i, s)\) has zero flow cost. More precisely,

\[
E[y_{i-1}(U)] = E[E[y_{i-1}(U)|y_i(U)]] \\
= E[(y_i(U) + w_i) \cdot Pr\{i^{th} \text{ customer has to wait}\}] \\
= (E[y_i(U)] + w_i) \cdot Pr\{i^{th} \text{ customer has to wait}\} \\
\implies Pr\{i^{th} \text{ customer has to wait}\} = \frac{E[y_{i-1}(U)]}{E[y_i(U)] + w_i}.
\]

Using similar argument, we can easily obtain two corollaries regarding the interpretations of the variables in our model.

**Corollary 1** The optimal value of \( z_{n+1} \) divided by \( w_{n+1} \) is the probability that the system does not experience an overtime.

**Remark 2** Corollary 1 is just a special case of Proposition 5 by defining \( y_{n+1} = 0 \).

**Corollary 2** When \( w_i = 1, \forall i = 1, 2, \ldots, n \), the value of \( z_i \) in the optimal solution is the expected number of customers served in a busy period with \( i \)-th patient as the first patient served in this busy period.

\textsuperscript{9}All the interpretations of the variables presented in this section are exact in the worse case situation given that the optimal policy obtained from solving (S\textsuperscript{2}) is adopted. The numerical evidence discussed in the next section shows that they can be good performance surrogates under different distributions of the service durations.
The second result is derived from the KKT conditions of Problem \((S^2)\).

**Proposition 6** If in the optimal solution to Problem \((S^2)\), the allocated service time slots are strictly positive, \((i.e., s_i > 0, \forall i \in I \subseteq 1,2,\ldots,n)\), then the dual variables must satisfy \(y_i \equiv K\), \(\forall i \in I\), where \(K\) is some nonnegative constant.

**Proof.** The proof only uses part of KKT conditions, which are

\[
\begin{aligned}
-y_i + \theta - \lambda_i &= 0, \forall i = 1,2,\ldots,n, \\
\lambda_i s_i &= 0, \forall i = 1,2,\ldots,n, \\
\theta &\geq 0.
\end{aligned}
\]

When \(s_i > 0, \forall i \in I \subseteq \{1,2,\ldots,n\}\), from the second set of constraints in (8), we get

\[
\lambda_i = 0, \forall i \in I \subseteq \{1,2,\ldots,n\}.
\]

Then

\[
y_i = \theta \geq 0, \forall i \in I \subseteq \{1,2,\ldots,n\},
\]

Define the constant \(K = \theta \geq 0\).

Combining the propositions established thus far, we can derive an important optimality condition for an appointment system:

**Theorem 4** Suppose \(w_i = 1\) for all \(i\). If in the optimal solution to Problem \((S^2)\), the allocated consultation slots are strictly positive for the first \(k\) patients, \((i.e., s_i > 0, i = 1,2,\ldots,k, \) where \(k > 0\)), then probabilities of waiting for service are the same for all the patients from \(i = 2,\ldots,k\), under the optimal worst case distribution.

Note that by Proposition 3 and Proposition 4, patients with positive consultation slots are all clustered in front of the schedule. This optimality condition is independent of all sequences as long as the schedule is fixed. This property of the optimal schedule is particularly useful for the patients - there is little incentive to choose between the slots in the clinical session if the objective is to minimize the chances of waiting for service. While this observation is derived for the optimal schedule under the worst case distribution, we believe that a suitably re-interpretation of the proof will show that the same property holds for any continuous service duration distribution.

6 Computational Results

All the computational studies are carried out in MATLAB environment on a Dell desktop (Core 1.86 GHz and 3GB of RAM). We solve the simplest form of SDP relaxation of the COP and CPP shown in (4). In MATLAB environment, we use YALMIP as the programming
interface with SDPT3 as the underlying SDP solver (cf. Löfberg (2004), Toh et al. (1999), Tutuncu et al. (2003)).

Note that expressing a problem as a COP or CPP does not resolve the difficulty of the problem, because it is already NP-hard to computationally capturing these conditions (cf. Murty et al. (1987)). Even solving a large scale SDP can be computationally inhibitive. Since our model lifts the original problem into a cone with higher dimensions, the current computational power limits the size of the problem instance we can solve to around 26 patients. While it is an interesting challenge to push the computational limit of this approach further, we leave this to future research. Instead, we use extensive numerical experiments to provide a glimpse to the structure of the optimal scheduling solutions.

6.1 Comparison with near-optimal solutions

Denton & Gupta (2003) solve the appointment scheduling problem using a sequential bounding method. Table 1 lists the near optimal schedules given in that paper, for 7 jobs with identically independent distributed service time (i.e., Uniform(0, 2)) under different cost parameters and fixed session length \( T = 7 \). The waiting time casts are identical among all the patients. In their numerical results, the optimality gap is less than 1%. We compute CPCMM to obtain the optimal schedule that minimizes the worst-case cost under all distributions with mean 1 and standard deviation \( \frac{1}{\sqrt{3}} \). The results of our model are presented in Table 2.

Note that both Denton & Gupta (2003) and CPCMM allow negative schedules. As proved in previous section, negative schedule only appears for the last patients.

<table>
<thead>
<tr>
<th>((w_1, w_{n+1}))</th>
<th>(3,14)</th>
<th>(5,12)</th>
<th>(7,10)</th>
<th>(3,12)</th>
<th>(5,10)</th>
<th>(7,8)</th>
<th>(3,10)</th>
<th>(5,8)</th>
<th>(7,6)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(s_1)</td>
<td>0.61</td>
<td>0.83</td>
<td>1.06</td>
<td>0.65</td>
<td>0.88</td>
<td>1.14</td>
<td>0.72</td>
<td>1.00</td>
<td>1.25</td>
</tr>
<tr>
<td>(s_2)</td>
<td>1.09</td>
<td>1.18</td>
<td>1.27</td>
<td>1.11</td>
<td>1.22</td>
<td>1.34</td>
<td>1.13</td>
<td>1.25</td>
<td>1.38</td>
</tr>
<tr>
<td>(s_3)</td>
<td>1.08</td>
<td>1.20</td>
<td>1.26</td>
<td>1.11</td>
<td>1.24</td>
<td>1.31</td>
<td>1.12</td>
<td>1.25</td>
<td>1.38</td>
</tr>
<tr>
<td>(s_4)</td>
<td>1.09</td>
<td>1.20</td>
<td>1.27</td>
<td>1.13</td>
<td>1.22</td>
<td>1.32</td>
<td>1.13</td>
<td>1.25</td>
<td>1.38</td>
</tr>
<tr>
<td>(s_5)</td>
<td>1.07</td>
<td>1.10</td>
<td>1.21</td>
<td>1.05</td>
<td>1.14</td>
<td>1.25</td>
<td>1.08</td>
<td>1.19</td>
<td>1.35</td>
</tr>
<tr>
<td>(s_6)</td>
<td>0.94</td>
<td>1.00</td>
<td>1.16</td>
<td>0.96</td>
<td>1.01</td>
<td>1.20</td>
<td>0.94</td>
<td>1.07</td>
<td>1.24</td>
</tr>
<tr>
<td>(s_7)</td>
<td>1.14</td>
<td>0.50</td>
<td>-0.23</td>
<td>1.01</td>
<td>0.31</td>
<td>-0.56</td>
<td>0.89</td>
<td>-0.01</td>
<td>-0.98</td>
</tr>
</tbody>
</table>

Table 1: Optimal schedules from Denton & Gupta (2003) under different cost structures

Next, we calculate the total cost under the schedules given in Table 1 and 2 through Monte Carlo simulation. The service time of each patient in CPCMM is generated under 4 common distributions used in practice: uniform, normal, two-point and gamma distribution\(^{11}\), with

\(^{10}\)In Denton & Gupta (2003), the objective function is the weighted sum of total waiting time, idle time and overtime of the doctor, while in our paper the objective function does not include the cost of idle time. According to Proposition 1 in Denton & Gupta (2003), we can transform the optimal scheduling problem in Denton & Gupta (2003) equivalently into our problem by combining the cost of idle time and overtime.

\(^{11}\)Similar results are obtained under a large set of distributions as well. We only report 4 most commonly used ones here.
mean 1 and standard deviation $1/\sqrt{3}$. All the 9 cost scenarios are tested. 50,000 rounds of simulation are executed for each of the 36 $(4 \times 9)$ scenarios. The expected cost is calculated by taking the average of 50,000 simulated costs. The expected total costs under different scenarios are then compared with the corresponding benchmarks given by Denton & Gupta (2003) under the uniform distribution. As shown in Table 3 the schedules obtained from CPCMM work sufficiently well when evaluated against the benchmarks. The expected total costs under CPCMM is close to that of Denton & Gupta (2003) under different cost structures and distributions. The gaps are within 2% and most of them are less than only 1%.

<table>
<thead>
<tr>
<th>$(w_1, w_{n+1})$</th>
<th>(3,14)</th>
<th>(5,12)</th>
<th>(7,10)</th>
<th>(3,12)</th>
<th>(5,10)</th>
<th>(7,8)</th>
<th>(3,10)</th>
<th>(5,8)</th>
<th>(7,6)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s_1$</td>
<td>0.35</td>
<td>0.87</td>
<td>0.94</td>
<td>0.52</td>
<td>0.89</td>
<td>0.99</td>
<td>0.76</td>
<td>0.92</td>
<td>1.05</td>
</tr>
<tr>
<td>$s_2$</td>
<td>1.32</td>
<td>1.09</td>
<td>1.16</td>
<td>1.22</td>
<td>1.10</td>
<td>1.20</td>
<td>1.08</td>
<td>1.13</td>
<td>1.26</td>
</tr>
<tr>
<td>$s_3$</td>
<td>1.05</td>
<td>1.17</td>
<td>1.25</td>
<td>1.08</td>
<td>1.19</td>
<td>1.30</td>
<td>1.11</td>
<td>1.22</td>
<td>1.38</td>
</tr>
<tr>
<td>$s_4$</td>
<td>1.12</td>
<td>1.29</td>
<td>1.38</td>
<td>1.16</td>
<td>1.31</td>
<td>1.44</td>
<td>1.21</td>
<td>1.35</td>
<td>1.53</td>
</tr>
<tr>
<td>$s_5$</td>
<td>1.20</td>
<td>1.31</td>
<td>1.36</td>
<td>1.23</td>
<td>1.31</td>
<td>1.42</td>
<td>1.26</td>
<td>1.33</td>
<td>1.50</td>
</tr>
<tr>
<td>$s_6$</td>
<td>1.17</td>
<td>1.27</td>
<td>1.20</td>
<td>1.20</td>
<td>1.20</td>
<td>1.25</td>
<td>1.24</td>
<td>1.18</td>
<td>1.33</td>
</tr>
<tr>
<td>$s_7$</td>
<td>0.79</td>
<td>0.00</td>
<td>-0.29</td>
<td>0.58</td>
<td>0.00</td>
<td>-0.61</td>
<td>0.33</td>
<td>-0.14</td>
<td>-1.04</td>
</tr>
</tbody>
</table>

Table 2: Optimal schedules from CPCMM under different cost structures

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>(w1, w2)</th>
<th>(3,14)</th>
<th>(5,12)</th>
<th>(7,10)</th>
<th>(3,12)</th>
<th>(5,10)</th>
<th>(7,8)</th>
<th>(3,10)</th>
<th>(5,8)</th>
<th>(7,6)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Benchmark</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Uniform</td>
<td>23.32</td>
<td>27.03</td>
<td>28.50</td>
<td>21.42</td>
<td>24.51</td>
<td>25.02</td>
<td>19.43</td>
<td>21.69</td>
<td>20.94</td>
<td></td>
</tr>
<tr>
<td>Two point</td>
<td>24.00</td>
<td>28.64</td>
<td>30.20</td>
<td>21.95</td>
<td>25.81</td>
<td>26.56</td>
<td>20.23</td>
<td>22.91</td>
<td>21.89</td>
<td></td>
</tr>
<tr>
<td>Gamma</td>
<td>22.73</td>
<td>27.53</td>
<td>28.87</td>
<td>20.93</td>
<td>25.08</td>
<td>25.84</td>
<td>19.48</td>
<td>22.10</td>
<td>22.21</td>
<td></td>
</tr>
</tbody>
</table>

Table 3: Comparison of the total costs between scheduling obtained by CPCMM and Denton under different distributions

It is worthwhile to point out that the total costs of CPCMM schedules do not vary much under different distributions. This significant observation reinforces the argument that the performance of CPCMM is insensitive to the choice of distributions, and the performance of the system can be adequately captured by the first and second moments information.

6.2 Probability of waiting for service

In this section, we run simulations to test whether the system wide performance obtained under the worst case distribution is indicative of the behavior under more natural random distributions.

Through solving the CPCMM, we can estimate the probability that a patient has to wait for service in the system, and also the probability of overtime work of the doctor. In the following we test the accuracy of the predictions from CPCMM under different distributions. We run a series of simulations by varying the number of patients ($n$), moments of service time ($\mu, \Sigma$) and cost structures ($w_i, i = 1, 2, \ldots, n + 1$).
For illustration, we report the simulation results with \( n = 8 \) patients case, each with mean 5 and standard deviation 1, and two sets of cost parameters: (1) \( w_i = 1, \forall i = 1, 2, \ldots, n, n + 1 \); (2) \( w_i = 1, \forall i = 1, 2, \ldots, n, w_{n+1} = 10 \).

Figure 3 shows both the predicted and the simulated probabilities of waiting for each patient and the probability of overtime for the doctor (i.e., the 9th patient). When \( w_{n+1} = 1 \), the optimal appointment strategy spreads out the arrivals of the patients so that except the first patient, all the other patients have to wait for service with probabilities close to 0.5. The clinic will almost surely experience overtime work in this system. When \( w_{n+1} \) is increased to 10, the clinic places higher weight on the overtime. In order to minimize overtime, the optimal appointment strategy bunches the arrival of the patients to decrease the chance of overtime work to around 0.7. Unfortunately, in this case, the chances of waiting for the patients are increased to more than 0.8. Furthermore, the simulated results under 4 different distributions show that the prediction based on CPCMM is robust, as depicted by the figure. The predicted probabilities work surprisingly well, taking into account that the value of the variables are obtained through solving the SDP relaxation of the original CPCMM model.

![Figure 3: Comparison of the predicted and simulated waiting probabilities](image)

6.3 Eye Clinic

In this subsection, we present the numerical results based on the data collected from the eye clinic and discuss pertinent managerial insights from our model.

We observe the consultation durations of 1021 patients in the clinic for 7 working days. The
mean and variance of the consultation time of the repeat patients are 6.24 minutes and 6.0 minutes respectively, while both parameters for the new patients are noticeably higher, with a mean of 9.97 minutes and a variance of 7.6 minutes.

In the simulation, we assume that one session lasts for 150 minutes. This mimics the current practice with one hour block, followed by half an hour break, followed by another one hour block. During one session, 24 patients are scheduled to arrive in the clinic, with 5 new patients arriving before 19 repeat patients. The consultation durations follow the empirical data in the clinic.\(^{12}\) The patients’ waiting time costs \(w_i\) are assumed to be identical among all the patients and normalized to 1. We set the value of \(w_{n+1}\) to be 1, 20 and 40, respectively, and solve the SDP relaxation of the CPCMM to find the optimal scheduling strategies. Figure 4 reveals several common interesting features from three optimal schedules under different \(w_{n+1}\).

![Figure 4: Optimal schedule when \(w_{n+1}\) is equal to 1, 20 and 40, given \(w_1=1\)](image)

It is interesting to note that the optimal schedule exhibits the pattern of “Bailey’s rule + Break”. First, the optimal schedule allocates near zero time slot to the first patient. Although Proposition 3 indicates that all the zero slots in this case should be placed at the end of the session, the time slots for the first few patients are indeed strictly positive but extremely small, as the system is congested and the overtime costs are large enough to induce such scheduling rules. The second outstanding feature is that, after serving the new patients, a break is inserted before switching to the repeat patients with lower variability. To confirm this feature, we run another group of experiments with 3 classes of patients. Similar patterns are observed - breaks are inserted after serving the first and the second class of patients.

One drawback of the optimal schedule is that it is generally not practical and non-intuitive. We next use the above insights to develop a simple but effective appointment schedule. To mimic the current practice in the clinic for comparison, we assume that each patient is assigned an equal interval of 5 minutes and a 30 minutes break is inserted after seeing 12 patients. This

\(^{12}\)Note that the sum of mean service durations of all patients is 168.41 minutes, which is higher than the session time, which means the system is congested.
schedule is called “Current Practice”. We simply modify the “Current practice’ by shifting a 30 minutes break after serving all the new patients, i.e. after the 5\textsuperscript{th} patient. This schedule is named “Modified Practice”.

We also allow the allocated service intervals to vary in the “Varying Interval” schedule. To resemble the optimal schedule (under \(w_{n+1} = 1\)), we assign zero time slots to the first patient and the last six patients. Other patients are assigned with time slots by rounding up their mean service durations, i.e., 10 minutes for a new patient and 7 minutes for a repeat patient. The remaining time is combined and inserted after the 5\textsuperscript{th} patient as a break.

<table>
<thead>
<tr>
<th></th>
<th>Uniform</th>
<th>Normal</th>
<th>Two-points</th>
<th>Gamma</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimal Schedule</td>
<td>352.58</td>
<td>349.80</td>
<td>355.37</td>
<td>352.78</td>
</tr>
<tr>
<td>Current Practice</td>
<td>564.13</td>
<td>560.18</td>
<td>570.31</td>
<td>535.37</td>
</tr>
<tr>
<td>Modified Practice</td>
<td>485.36</td>
<td>479.95</td>
<td>491.95</td>
<td>462.44</td>
</tr>
<tr>
<td>Varying Interval</td>
<td>358.24</td>
<td>353.61</td>
<td>363.60</td>
<td>354.83</td>
</tr>
</tbody>
</table>

Table 4: Total cost under different schedules when the sequence is fixed to “New Patients First” and \(w_1 = 1, w_{n+1} = 1\).

As shown in Table 4, implementing a schedule resembling the optimal schedule dramatically decrease the total cost, by about 60\%, as compared to the current practice. Interestingly, it seems that one can significantly improve the performance of the system by simply inserting a break after serving one class of patients in the optimal scheduling, and implementing a variant of the Bailey’s rule at the front and end of the queue. The easily implemented “Varying Interval” strategy makes it quite attractive for practical considerations.

Note that the above simulation results are obtained under \(w_{n+1} = 1\). In most environment, the overtime cost \(w_{n+1}\) is likely to be large and should be proportional to the number of patients seen in the clinic. The choice of \(w_{n+1} = 1\) is thus a conservative estimate and assumes the doctor places small penalty on the overtime work. In what follows, we summarize the features of optimal schedules when \(w_{n+1}\) increases.

The pattern of “Bailey’s rule + break” seems to be quite robust no matter how the the overtime cost changes. Besides this pattern, Figure 4 indicates several interesting features as the cost of doctor’s overtime increases:

- More patients are assigned with near zero time slots at the beginning of the session;
- Less patients are assigned with zero time slots at the end of the session;
- Longer time slot is assigned to the last patient;

Intuitively, all these features benefit the clinic that prefers a shorter overtime. Consequently, patient’s waiting time is increased.
7 Sequencing Problem

We have shown that the scheduling problem can be effectively solved using a simple convex program. We discuss in the rest of this section some insights on the optimal sequence of arrival of the patients. We assume \( s_i = \mu_i \) to remove the needs to address the scheduling decision, and focus solely on the sequencing problem. We want to determine the sequence to minimize the total waiting time, with \( w_i = 1 \) for \( i = 1, \ldots, n + 1 \), in our appointment problem. In particular, we address the question: Is it optimal to sequence the patients with smaller variance to arrive earlier in the session?

The following example, unfortunately, shows that this is not true in general.

Example 1 Assume the service durations \( \{U_i\} \) are independent. Let \( U_1 = 0 \) or 2 with equal probability, \( U_2, U_3 = 0 \) or 4 with equal probability, and \( U_k = 0 \) or 6 with equal probability for \( k > 3 \). In this case, \( P(Z_1 = \pm 1) = \frac{1}{2} \), \( P(Z_j = \pm 2) = \frac{1}{2} \), for \( j = 2, 3 \), and \( P(Z_k = \pm 3) = \frac{1}{2} \), for \( k = 4, \ldots, n \). We compare the performance of the sequence \( \{Z_1, Z_2, \ldots, Z_n\} \) and another obtained by switching \( Z_1 \) and \( Z_2 \) in the sequence. Note that patients in the first sequence are ordered in non-decreasing order of the variances. We ran simulation and plot the difference in the performance, (i.e., the difference in total waiting time), as a function of the number of patients \( n \), in Figure 5.

![Figure 5: Difference in Performance w.r.t n](image)

As the number of patients is small, say \( n \) below 20, scheduling patients with smaller variance first is generally better in this example. Surprisingly, this behavior changes as \( n \) increases, and for a large enough \( n \), putting patient 2 in front of patient 1 is now beneficial in reducing total waiting time! Consequently, sequencing patients in increasing variance is no longer optimal.
The simulation result also suggests that the optimal sequence is affected by the number of patients in each class, and hence sequencing patients by looking at pair of patients in isolation through stochastic ordering is probably a futile attempt.

To add to the perplexity of the results, we show next that under the deterministic model where $s_i$ is set to a constant (e.g. patients scheduled to arrive in constant interval), then knowing the service duration $U_i$ in advance does not make the sequencing problem any easier! In fact, under this deterministic model, Vanden (1997) have shown earlier that when the objective coefficients $w_i$ are allowed to take arbitrary values, to determine the optimal sequence is equivalent to solve a nonlinear knapsack problem and is thus NP-hard. Surprisingly, we show next that the problem in fact remains NP-hard even when $w_i$’s are identical.

**Theorem 5** The appointment scheduling problem is NP-hard in the strong sense, even if the allocated appointment interval $S_j$ is constant for all $j$, and $w_i = 1$ for all $i$.

We refer the readers to Appendix B for a formal proof of this result.

In the rest of this section, we describe how the proposed approach can be used to address this class of sequencing problem, even when scheduling decision has to be made in conjunction with the sequencing decision. Let

\[(P)' \quad Z_P' = \sup_{U \sim (\mu, \Sigma)} \{E[f(s, \sigma)]\}\]

where $f(s, \sigma)$ is the cost of the second stage network flow model after fixing schedule $s$ and sequence $\sigma$.

To apply our approach on the appointment design problem, we introduce scheduling and sequencing decision variables $s$ and $P = (p_{ij})$ into the model in the following way:

**Theorem 6** $(P)'$ can be solved as the following completely positive program, i.e., $Z_P' = Z_C'$:

\[(C)' \quad Z_C' = \max_{s.t.} \quad X \cdot P - s^T y \]

\[
\begin{pmatrix}
\mu^T \\
\sum_i X^T \\
y
\end{pmatrix}
\begin{pmatrix}
y^T \\
y^T \\
Z^T \\
Z^T \\
\end{pmatrix} \geq \phi 0
\]

where the decision variables are $y, z \in \mathbb{R}^n$, $Y, W, Z, X, V \in \mathbb{R}^{n \times n}$, and $P = (p_{ij}) \in \{0, 1\}^{n \times n}$ is a known permutation matrix given by

\[
p_{i,j} = \begin{cases} 
1 & \text{if } \sigma(j) = i \\
0 & \text{otherwise}
\end{cases} , \quad i, j = 1, 2, \ldots, n.
\]
This formulation models the expected waiting cost of the appointment system, when the sequencing decision \( P = (p_{i,j}) \) and scheduling decision \( s \) are fixed. It replaces the objective function \( tr(X) \) by \( X \bullet P \), due to the sequencing consideration. The corresponding copositive cone program is now:

\[
(S^2)' \quad Z'_{S^2} := \min \nabla \bullet \Gamma + \mu^T \beta + \alpha
\]

\[
s.t. \quad \begin{cases}
\sum_{j=1}^n (-u_j + v_j) + \alpha & \beta^T = \frac{(s \ 0) - \sum_{j=1}^n u_j (a_j)^T}{\tau} \\
\frac{\beta}{\tau} & \Gamma = \begin{pmatrix}
-P/2 \\
-\sum_{j=1}^n v_j (a_j - e_j)^T
\end{pmatrix} \begin{pmatrix}
-P/2 \\
-\sum_{j=1}^n v_j (a_j - e_j)^T
\end{pmatrix}^T
\end{cases}
\]

\[
\sum_{i=1}^n p_{i,j} = \sum_{j=1}^n p_{j,i} = 1, \forall i = 1, 2, \ldots, n
\]

\[
p_{i,j} \in \{0, 1\}, \forall i, j = 1, 2, \ldots, n
\]

When the sequencing becomes parts of the decision variables, due to its discrete nature (\( n^2 \) binary variables), the time consumed in searching for optimal sequence (e.g. using a Branch and Bound (B&B) type method) increases exponentially in the size of the instance. We developed a simple B&B code to take advantage of the special structure of the problem by adding some symmetry breaking constraints, and can solve the sequencing problem for up to 8 customers efficiently.

### 7.1 Numerical Results

Our earlier numerical examples have debunked the conjecture on the optimality of the smaller-variance-first rule. However, what is the structure of the optimal sequencing policy? We use a set of numerical experiments to provide a glimpse to the answer of this question.

The numerical example assumes 6 patients in the system, with identical mean consultation duration of 5 time units, but with different standard deviations ([1 1 2 2 3 3]). By fixing the allocated service time to be the mean (\( \mu = 5 \)), we obtain the optimal sequences by solving the CPCMM model. In Figure 6, we observe that when the ratio between waiting time cost (\( w_i \)) and overtime cost (\( w_{n+1} \)) is 1 : 1, smaller-variance-first rule is indeed optimal. However, as the overtime cost is sufficiently high as compared to the waiting time cost (e.g., 1:100), the optimal sequence appears to be “U-shaped” in terms of the variability of the service durations. Namely, the patients with larger variances are either assigned to the beginning or the end of the session.

Monte Carlo simulation results indeed show that under the “U-shape” sequencing rule, the expected total cost is smaller than that under the smaller-variance-first rule. The above
structure continues to hold for many different sets of parameters in the experiments. We conjecture that it holds in general:

**Conjecture.** When the allocated service time for each patient is set to the mean service time and the overtime cost is sufficiently high, the optimal sequence exhibits a U-shape pattern.

We leave the resolution to the above conjecture to future research.

8 Conclusion

We propose a novel approach to deal with the difficult patient scheduling and sequencing problem. Instead of planning against a fixed service distribution, we plan against a canonical set of service distributions with the same mean and covariance parameters. The canonical distribution is “constructed” via a copositive cone program. In this way, we reduce a difficult two stage stochastic programming problem into a single stage convex programming problem. Through extensive simulations we show that the optimal schedules solved under the “worst case” give near-optimal solutions when the objective is to minimize expected total cost. This approach allows us to shed some light on the structure of the optimal schedule and sequence, which we can readily modify to obtain practical and efficient schedule and sequence.

The approach can be generalized to deal with the situation when the patients need to undergo a test (random duration) prior to the consultation, a pertinent feature in many eye clinic. The network flow approach can also be conceivably extended to deal with other practical considerations in a clinical environment. There are however several limitations with this approach - the computational difficulty associated with solving large scale SDP limits our ability to solve
large scale scheduling problem. Furthermore, we need to devise a specialized B&B algorithm to deal with the case when sequencing decision is involved. We are however hopefully that larger instances of the sequencing problem can be solved if we move to a commercial platform to deal with the 0-1 problem. We leave this for future research.

Acknowledgment

We would like to thank Prof John Buzacott, Gideon Weiss, and Diwakar Gupta for their comments on earlier version of the paper. We would like to thank Jiajie Liang for providing the data.

Appendix I. Copositive and Completely Positive Programs

Another equivalent definition for completely positive matrices is that $A$ is completely positive if and only if there exist vectors $v_1, v_2, \ldots, v_k \in \mathbb{R}_+^n$ such that

$$A = \sum_{i=1}^{k} v_i v_i^T.$$  \hfill (9)

Clearly, the factorization of a completely positive matrix is not unique. The representation (9) is called a rank 1 representation of $A$. The decomposition of $A$ into the sum of rank 1 matrices is referred to as a completely positive decomposition. The minimal $k$ for which there exists a rank 1 representation is called the cp-rank of $A$. From the definition, it is clear that for every CP matrix $A$, cp-rank($A$) $\geq$ rank($A$), where the equality holds when $n \leq 3$, or when rank($A$) $\leq 2$. An upper bound on cp-rank of $A$ in terms of rank($A$) (when rank($A$) $\geq 2$) is

$$cp\text{-rank}(A) \leq \frac{\text{rank}(A)(\text{rank}(A) + 1)}{2} - 1.$$

Since its introduction, copositive programming has been closely related to quadratic and combinatorial optimization\textsuperscript{13}. It has been shown that the standard quadratic problem can be exactly reformulated as a COP:

\begin{align*}
\text{(StQP)} \quad & \min \ x^T Q x \\
& s.t. \ e^T x = 1 \\
& \quad x \geq 0
\end{align*}

\begin{align*}
= \min \ Q \bullet X \\
& s.t. \ e e^T \bullet X = 1 \\
& \quad X \succeq_{co} 0
\end{align*}

where $e$ denotes the vector of all ones.

Then Burer (2009) developed the much more general result that every quadratic problem with a mixture of continuous and binary variables can be rewritten as a CPP under an easily enforceable condition on the feasible region:

\textsuperscript{13}For a comprehensive survey on the history and applications of COP, please refer to Dür (2009).
max $x^T Q x + 2c^T x$

\[ \begin{align*}
\text{s.t. } & a_i^T x = b_i, \quad \forall i = 1, \ldots, m \\
& x \geq 0 \\
& x_j \in \{0, 1\} \quad \forall j \in \mathcal{B}
\end{align*} \]

max $Q \bullet X + 2c^T X$

\[ \begin{align*}
\text{s.t. } & a_i^T x = b_i^2, \quad \forall i = 1, \ldots, m \\
& X_{jj} = x_j, \quad \forall j \in \mathcal{B} \\
& \left( \begin{array}{cc}
1 & x^T X \\
x & X
\end{array} \right) \succeq c_p 0
\end{align*} \]

Expressing the problem as a COP or CPP does not resolve the difficulty of the problem, because capturing these cones is generally suspected to be difficult (Bomze et al. (1995), Burer (2009), Dür (2009)). For instance, the problem of testing if a given matrix is copositive is known to be in the co-$\mathcal{NP}$-complete class (Murty et al. (1987)). However, such a transformation shifts the difficulty into the respective convex cones, so that whenever something more is known about copositive or completely positive matrices, it can be applied uniformly to many $\mathcal{NP}$-hard problems (Burer (2009)). One such result is a well-known hierarchy of linear and semidefinite representable cones that approximate the copositive and completely positive cone, which is elaborated next.

Klerk et al. (2002) showed that there exists a series of linear and semidefinite representable cones approximating the copositive cone $\mathcal{CO}_n$ from the inside, i.e.

\[ \begin{align*}
\exists \text{ closed convex cones } \{ \mathcal{K}_n^r : r = 0, 1, 2, \ldots \} \\
\text{such that } \mathcal{K}_n^r \subseteq \mathcal{K}_n^{r+1}, \forall r \geq 0 \text{ and } \bigcup_{r \geq 0} \mathcal{K}_n^r = \mathcal{CO}_n,
\end{align*} \]

where $X$ denotes the closure of set $X$. The dual cones $\{(\mathcal{K}_n^r)^* : r = 0, 1, 2, \ldots \}$ approximate the completely positive cones $\mathcal{CP}_n$ from outside, i.e.

\[ \begin{align*}
(\mathcal{K}_n^r)^* \supseteq (\mathcal{K}_n^{r+1})^*, \forall r \geq 0 \text{ and } \bigcap_{r \geq 0} (\mathcal{K}_n^r)^* = \mathcal{CO}_n^* = \mathcal{CP}_n.
\end{align*} \]

The simplest approximation we consider in this paper is the case where $r = 0$:

\[ \mathcal{K}_n^0 = \left\{ A \in \mathcal{S}_n \mid \exists X \in \mathcal{S}_n^+, \exists Y \in \mathbb{R}_+^{n \times n}, \ A = X + Y \right\}, \]

\[ \left(\mathcal{K}_n^0\right)^* = \left\{ A \in \mathcal{S}_n^+ \mid A \in \mathbb{R}_+^{n \times n} \right\}. \]

It can be shown that when $n \leq 4$, the above two approximations are exact, i.e. $\mathcal{K}_n^0 = \mathcal{CO}_n$ and $\left(\mathcal{K}_n^0\right)^* = \mathcal{CP}_n$.

The higher order approximation ($r \geq 1$) becomes much more complicated. For instance, when $r = 1$, Parrilo (2000) showed that

\[ \mathcal{K}_n^1 = \left\{ A \in \mathcal{S}_n \mid \exists M^{(i)} \in \mathcal{S}_n, i = 1, 2, \ldots, n \text{ such that } \begin{cases} A - M^{(i)} \succeq 0, i = 1, 2, \ldots, n \\
M^{(i)}_{ii} = 0, i = 1, 2, \ldots, n \\
M^{(i)}_{jj} + M^{(j)}_{jj} + M^{(j)}_{ji} = 0, i \neq j \\
M^{(i)}_{jk} + M^{(j)}_{ik} + M^{(k)}_{ij} \succeq 0, i \neq j \neq k \end{cases} \right\}. \]
Appendix II. Hardness of Deterministic Variant

Assume that there are \( n \) patients to be scheduled. For each patient \( J_j \), \( j = 1, 2, \cdots, n \), let \( p_j \) denote the allocated appointment interval and \( q_j \) the corresponding actual consultation time.

Let \( \mathcal{P} = \{p_1, p_2, \cdots, p_n\} \) and \( \mathcal{Q} = \{q_1, q_2, \cdots, q_n\} \). For any given sequence \( \sigma \) and \( \mathcal{P} \), let \( C_j(\mathcal{P}, \sigma) \) denote the scheduled completion time for patient \( J_j \). Therefore, the appointment time given to patient \( J_j \) under \( \sigma \) is:

\[
a_j(\mathcal{P}, \sigma) = C_j(\mathcal{P}, \sigma) - p_j. \tag{10}
\]

Let \( C_j(\mathcal{P}, Q, \sigma) \) denote the actual completion time for patient \( J_j \). The waiting time for this patient is

\[
w_j = C_j(\mathcal{P}, Q, \sigma) - q_j - a_j(\mathcal{P}, \sigma) = (C_j(\mathcal{P}, Q, \sigma) - q_j) - (C_j(\mathcal{P}, \sigma) - p_j). \tag{11}
\]

The objective of the appointment sequencing problem now is to find a sequence \( \sigma \) such that the total patient’s waiting time \( \sum_{j=1}^{n} W_j \) is minimized. Interestingly, this problem is related to the following well-known NP-complete problem (cf. Gary & Johnson (1979)).

Numerical 3-Dimensional Matching (N3DM)

**Instance:** Given three disjoint sets \( W, X, Y \), each containing \( m \) elements, a size \( s(a) \in \mathbb{Z}^+ \) for each element \( a \in W \cup X \cup Y \), and a bound \( B \in \mathbb{Z}^+ \).

**Question:** Can \( W \cup X \cup Y \) be partitioned into \( m \) disjoint sets \( A_1, A_2, \cdots, A_m \) such that each \( A_i \) contains exactly one element from each of \( W, X \) and \( Y \) and such that, for \( i = 1, 2, \cdots, m \), \( \sum_{a \in A_i} s(a) = B \)?

**Proof.** We show that given an instance of N3DM, we can construct an instance of the appointment scheduling problem as follows. Suppose there are \( 3m + 1 \) jobs such that \( p_j = 5M \) for \( j = 1, 2, \cdots, 3m + 1 \); and

\[
\begin{align*}
q_i &= 6M + s(i) & \text{if } i \in W, \\
q_j &= 4M + s(j) & \text{if } j \in X, \\
q_k &= 5M - B + s(k) & \text{if } k \in Y, \\
q_{3m+1} &= (m^2 + 5)M
\end{align*}
\]

where \( M \) is a number much larger than \( B \), e.g., \( M > m^2B \). The question is: can we find a sequence \( \sigma \) to the appointment scheduling problem such that,

\[
\sum_{j=1}^{3m+1} w_j \leq Z = mM + 2 \sum_{i \in W} s(i) + \sum_{j \in X} s(j) + Z.
\]

For notational convenience, we call jobs corresponding to set \( W, X, Y \), \( W \)-type, \( X \)-type and \( Y \)-type respectively. To start with, suppose that the N3DM problem has a feasible solution. We then can obtain a sequence containing the following \( m + 1 \) consecutive blocks, where the
jobs in Block $i$, $i = 1, \cdots, m, m + 1$ correspond to $a \in A_i$, and within each $A_i$ jobs are sequenced in $W, X, Y$ order. The $m + 1^{th}$ block contains only $J_{3m+1}$. By definition, in the above sequence, $C_j(P, \sigma) = 5jM$ for $j = 1, 2, \cdots, 3m + 1$. Also, in the actual processing, there is no idle time for the machine. In each block, the waiting time is zero for the $W$-type job, $M + s(i)$ for $X$-type job and $s(i) + s(j)$ for $Y$-type job, where $s(i)$ corresponds to the $W$-type job processed in the first position and $s(j)$ corresponds to the $X$-type job processed in the second position. Also, the waiting time is zero for $J_{3m+1}$. Hence, the total waiting time equals to

$$mM + 2\sum_{i \in W} s(i) + \sum_{j \in X} s(j) = Z.$$ 

Then, suppose that there exists a sequence $\sigma$ to the appointment scheduling problem such that $\sum_{j=1}^{3m+1} w_j \leq Z = mM + 2\sum_{i \in W} s(i) + \sum_{j \in X} s(j)$. We first claim that in such a sequence, $J_{3m+1}$ must be processed in the last position. Otherwise, after $J_{3m+1}$ there are still other patients left and the waiting time is at least $m^2M$, which is larger than $Z$. This is a contradiction to our assumption. Note that the waiting time of any patient that is scheduled immediately after a $W$-type patient is at least $M + s(i)$. Therefore, the total waiting time for those patient is at least $mM + \sum_{i \in X} s(i)$. Now we show that $X$-type patient must be scheduled immediately after a $W$-type patient. If not, what follows $W$-type patient must be $W$-type or $Y$-type patient. Firstly, if it is $W$-type patient who is scheduled immediately after, then the total waiting time should be at least $(m + 1)M + \sum_{i \in X} s(i)$, which is bigger than $Z$. Otherwise, if $Y$-type patient would be put immediately after $W$-type patient, then total waiting time for all patients should be at least $(m + 1)M - B$, which is also bigger than $Z$ and a contradiction. In such a case, in each block, the sequence is $W$-type, $X$-type, $Y$-type and the total waiting time of all the patients is $mM + 2\sum_{i \in W} s(i) + \sum_{j \in X} s(j) = Z$. In the actual processing, there will not be any idle time for the machine. These three consecutive jobs $W, X, Y$ form a block with total processing time $15M$, to ensure that a $W$-type job in the next block can start without waiting. Namely, in each block we have three jobs and $s(i) + s(j) + s(k) = B$. 

### Appendix III. Constructing the Dual of Problem (C)

To construct the dual, we first transform Problem (C) into a standard primal CPP by introducing a matrix variable $M \in \mathbb{R}^{(3n+1) \times (3n+1)}$ such that

$$M = \begin{pmatrix}
1 & \mu^T & y^T & z^T \\
\mu & \sum X & V \\
y & X^T & Y & W \\
z & V^T & W^T & Z
\end{pmatrix} \succ_{cp} 0.$$
Then the moment requirements can be treated as some linear constraints on \( M \), i.e.,

\[
\begin{pmatrix}
1 & 0^T & 0^T & 0^T \\
0 & O_n & O_n & O_n \\
0 & O_n & O_n & O_n \\
0 & O_n & O_n & O_n \\
0 & e_j^T & 0^T & 0^T \\
\varepsilon_j & O_n & O_n & O_n \\
0 & O_n & O_n & O_n \\
0 & 0^T & 0^T & 0^T \\
0 & e_{j1,j2} & O_n & O_n \\
0 & O_n & O_n & O_n \\
0 & O_n & O_n & O_n \\
\end{pmatrix} \cdot M = 1,
\]

\[
\begin{pmatrix}
1 & 0^T & 0^T & 0^T \\
0 & O_n & O_n & O_n \\
0 & O_n & O_n & O_n \\
0 & O_n & O_n & O_n \\
0 & e_j^T & 0^T & 0^T \\
\varepsilon_j & O_n & O_n & O_n \\
0 & O_n & O_n & O_n \\
0 & 0^T & 0^T & 0^T \\
0 & e_{j1,j2} & O_n & O_n \\
0 & O_n & O_n & O_n \\
0 & O_n & O_n & O_n \\
\end{pmatrix} \cdot M = \mu_j, \forall j = 1, 2, \ldots, n,
\]

\[
\begin{pmatrix}
1 & 0^T & 0^T & 0^T \\
0 & O_n & O_n & O_n \\
0 & O_n & O_n & O_n \\
0 & O_n & O_n & O_n \\
0 & e_j^T & 0^T & 0^T \\
\varepsilon_j & O_n & O_n & O_n \\
0 & O_n & O_n & O_n \\
0 & 0^T & 0^T & 0^T \\
0 & e_{j1,j2} & O_n & O_n \\
0 & O_n & O_n & O_n \\
0 & O_n & O_n & O_n \\
\end{pmatrix} \cdot M = \Sigma_{j_1,j_2}, \forall j_1, j_2 = 1, 2, \ldots, n,
\]

where \( E_{j_1,j_2} = (\varepsilon_{i,j}) \in \mathbb{R}^{n \times n} \) with

\[
\varepsilon_{i,j} = \begin{cases} 1 & \text{if } i = j_1, j = j_2, \\ 0 & \text{otherwise.} \end{cases}
\]

We can also transform the constraints and the objective function in (C) into some linear functions on \( M \), e.g.,

\[
\begin{pmatrix} a_j \\ -e_j \end{pmatrix}^T \begin{pmatrix} y \\ z \end{pmatrix} = -1, \forall j = 1, 2, \ldots, n \iff \begin{pmatrix} 1 & 0^T & a_j^T & -e_j^T \\
0 & O_n & O_n & O_n \\
0 & a_j & O_n & O_n \\
0 & e_j & O_n & O_n \\
\end{pmatrix} \cdot M = 0, \forall j = 1, 2, \ldots, n.
\]

Then we introduce the dual variables for the standardized formulation of Problem (C) in a minimization form as follows:
Next, utilizing the standard conic duality, we can uncover the dual of Problem (C),
Simplifying the above formulation using vector and matrix notations, and transforming it back into the minimization form by letting
\[ \alpha = -\alpha, \beta = -\beta, \text{ and } \Gamma = -\Gamma, \]
we get the following dual problem of (C),
\[
\begin{aligned}
\min & \quad \Sigma \cdot \Gamma + \mu^T \beta + \alpha \\
\text{s.t.} & \quad \frac{\sum_{j=1}^n (-u_j + v_j) + \alpha}{\frac{\beta}{2}} = \frac{\beta^T}{T} \\
& \quad \begin{pmatrix}
\Sigma & \frac{-I_n/2}{\Omega_n} \\
\frac{-I_n/2}{\Omega_n} & \frac{-I_n/2}{\Omega_n}
\end{pmatrix}
\begin{pmatrix}
\frac{s}{0} \\
\frac{-\sum_{j=1}^n a_j}{2}
\end{pmatrix}
\begin{pmatrix}
\frac{-\sum_{j=1}^n a_j}{2} \\
\frac{-\sum_{j=1}^n a_j}{2}
\end{pmatrix}
\begin{pmatrix}
\frac{-\sum_{j=1}^n a_j}{2} \\
\frac{-\sum_{j=1}^n a_j}{2}
\end{pmatrix}
\end{aligned}
\]
Finally, Problem (5) can be consolidated as a copositive programming problem in minimization form shown as Problem \( (S^2) \).

References


