

Appointment Scheduling with Discrete Random Durations*

Mehmet A. Begen[†]

Maurice Queyranne[‡]

Abstract

We consider the problem of determining optimal appointment schedule for a given sequence of jobs (e.g., medical procedures) on a single processor (e.g., operating room, examination facility), to minimize the expected total underage and overage costs when each job has a random processing duration given by a joint discrete probability distribution. Simple conditions on the cost rates imply that the objective function is submodular and L-convex. Then there exists an optimal appointment schedule which is integer and can be found in polynomial time. Our model can handle a given due date for the total processing (e.g., end of day for an operating room) after which overtime is incurred and, no-shows and emergencies.

1 Introduction and Motivation.

Our research concerns appointment scheduling of jobs on a highly utilized processor when the processing durations are stochastic, and jobs are not available before their appointment dates¹. We came across this problem in appointment scheduling of oncologist consultations and radiation therapy treatments for cancer patients as well as in surgery scheduling. There are many other challenging and important real-life applications for this setting including healthcare diagnostic operations (such as CAT scan, MRI) and physician appointments, as well as project scheduling, container vessel and terminal operations, gate and runway scheduling of aircrafts in an airport. For example, in surgery scheduling, patients or surgeries are the jobs, the operating room (OR) and associated resources are the processor, and the surgeon or the hospital is the scheduler. Some of these applications may have a specific due date for the end of processing, e.g., end of day for an OR, whereas there may not be

a particular due date for others, e.g., project scheduling. The need for a good schedule is crucial, and savings from such a schedule can be significant. In most cases, an appointment schedule needs to be prepared before any processing starts. It assigns each procedure an allocated duration by specifying the appointment date at which the required personnel and equipment, and the job or patient will be available. However, due to the uncertain processing durations, some jobs may finish earlier, whereas some others may finish later, than the appointment date of the next job. As the appointment dates have to be determined in advance, there are only limited recourse options when the actual duration of a job differs from its planned value. When a procedure finishes earlier than the next procedure's appointment date, the processor and other resources remain idle until the appointment date of the next job. This results in resource under-utilization. On the other hand, if a job finishes later than the next job's appointment date, the next job has to wait for the preceding procedure to complete and will start later than its original appointment date. This results in waiting for the next job and may cause overtime for the processor and resources at the end of the schedule. Therefore, there is an important trade-off between under-utilization, overtime and job waiting times. We are interested in generating an appointment vector² that minimizes the expected total cost of resource under-utilization, overtime and job waiting times. Finding such a schedule is more challenging but more valuable and useful when processing durations have more variability. Figure 1 shows an instance with 3 jobs G, B, R to be processed in this order. An appointment schedule (A_G, A_B, A_R) is given. Once the processing starts, due to the random processing durations, some jobs may be early whereas some others may be late as shown in Figure 1.

This problem can be modeled as a multistage stochastic program, but there are significant computational difficulties due to the need for multidimensional numerical integration (see Section 2). To our best knowledge, all the analytical studies we are aware of, even the ones with discrete epochs for job arrivals,

*This research was supported by Discovery Grant from Natural Sciences and Engineering Research Council (NSERC) of Canada to the second author, and a PGSD3 NSERC scholarship to the first author.

[†]University of British Columbia, Sauder School of Business, 2053 Main Mall, Vancouver, BC V6T1Z2, Canada.

[‡]University of British Columbia, Sauder School of Business, 2053 Main Mall, Vancouver, BC V6T1Z2, Canada.

¹To conform with scheduling terminology, we use the term date to denote a point in time. In most applications of appointment scheduling, the appointment "dates" are actually appointment times within the day for which the jobs are being scheduled.

²We use appointment schedule and appointment vector interchangeably.

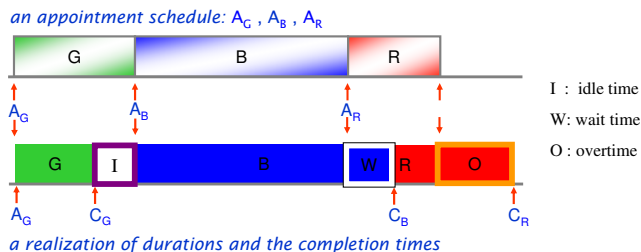


Figure 1: A three-job instance, and a realization of the processing durations.

use continuous processing duration distributions. For a given sequence of jobs, only small instances can be solved to optimality, larger instances require heuristics.

We study a discrete time version of the model and establish discrete convexity properties of the objective function. We prove that the objective function is L-convex under mild assumptions on cost coefficients. Furthermore, we show that there exists an optimal integer appointment schedule minimizing our objective. This result is important as it allows us to optimize only over integer appointment schedules without loss of optimality. All these results on the objective function and optimal appointment schedule enable us to develop a polynomial time algorithm, based on discrete convexity, that, for a given processing sequence, finds an appointment schedule minimizing the total expected cost. This algorithm invokes a sequence of submodular *set* function minimizations, for which various algorithms are available, see e.g., Fujishige [6], Iwata [7], McCormick [9] and Orlin [11].

When processing durations are stochastically independent we evaluate the expected cost for a given processing order and an integer appointment schedule, efficiently both in theory (in polynomial time) and in practice (computations are quite fast in our preliminary computational experiments). Independent processing durations lead to faster algorithms.

Our modeling framework can include a given due date for the end of processing (e.g., end of day for an operating room) after which overtime is incurred, instead of letting the model choose an end date. We also extend our analysis to include no-shows and emergency jobs. The expected benefits of this research effort include reduced job waiting times, reduced overtime and improved capacity utilization.

Our paper is organized as follows. We start with a literature summary in Section 2. Section 3 states our assumptions, introduces our notation and formally defines the problem and objective function. Section 4 gives some basic properties of the objective function and optimal solutions. We show the existence of an optimal

appointment vector which is integer in Section 5. Section 6 establishes the submodularity and L-convexity of the objective function under a mild condition on cost coefficients. We show that the total expected cost can be minimized efficiently and give the complexity of this minimization in Section 7. In Section 8, we briefly discuss our results in the case of independent processing durations, objective function with a due date, no-shows and emergency jobs. Section 9 discusses the current work and future work, and it concludes the paper.

2 Related Literature.

There are many studies done in the last 50 years about appointment scheduling, especially in healthcare. Here, we present the ones that we believe are the most relevant to our research. The use of appointment systems is not limited to service industries but also extends to other areas, such as project management, manufacturing and transportation.

Sabria and Daganzo [15] consider scheduling of arrivals of container vessels at a seaport. Weiss [19] recognized that appointment scheduling problem has a closed form solution when there are only two jobs, and it coincides with the well known newsvendor solution from inventory theory. Robinson et al. [14] extends this result to three jobs by obtaining optimality conditions. Zipkin [20] presents an analysis on the structure of a single-item multi-period inventory system, closely related to newsvendor problem, by using discrete convexity. Cayirli and Veral [4] review the literature on appointment systems of outpatient scheduling. The authors classify the literature in terms of methodologies and modeling aspects considered, and provide a discussion of performance measures. The authors conclude that the existing literature provides very situation-specific solutions and does not offer generally applicable and portable methodologies for appointment systems design in outpatient scheduling.

In a queuing based study, Wang [17] develops a model to find appointment dates of jobs in a single server system to minimize expected customer delay and server completion time with identical jobs and costs, and exponential processing duration distributions. In his numerical studies, the optimal allocated time for each job shows a “dome” structure, i.e., it increases first and then decreases. In another study, Wang [18] investigates the sequencing problem with the same setting but with distinct exponential distributions. He conjectures that sequencing with increasing variance is optimal. Bosch, Dietz and Simeoni [3] present a model with i.i.d. Erlang processing durations and identical cost coefficients. In their model, customers can arrive only at one of discrete potential arrival epochs, which

are equally spaced, and the decision variable is the number of customers to be scheduled at each potential arrival epoch. In a similar paper, Kaandorp and Koole [8] study outpatient appointment scheduling with exponential processing durations with no-shows. They take advantage of the exponential distribution in their computations and define a neighborhood structure and an exact search method. However for large instances, they had to develop a heuristic due to high computation times of their search method.

An other important stream of appointment scheduling research is based on stochastic programming. Denton and Gupta [5] develop a two-stage stochastic linear program to determine the optimal appointment dates for a given surgery sequence and due date for the end of processing horizon. The authors use general, i.i.d. and continuous processing durations, and identical server idling cost coefficients for all jobs. They infer from stochastic programming results that their model is a convex minimization problem, and they develop an algorithm with sequential bounding for solving small sized instances. They develop heuristics to solve larger instances. In a similar study, Robinson and Chen [13] develop a stochastic linear program for finding appointment dates for a fixed sequence of surgeries and propose a Monte-Carlo based solution method. Due to the high computational requirements of Monte-Carlo integration, they have to develop heuristics in which they use the “dome” structure of the optimal policy as reported in Wang [17].

We aim to develop a sufficiently generic and portable framework to solve the appointment scheduling problem efficiently both in theory and practice.

3 Assumptions and Notation.

There are $n + 1$ jobs that need to be sequentially processed on a single processor. The processing sequence is given. An appointment schedule needs to be prepared before any processing can start. Jobs will not be available before their appointment dates. When a job finishes earlier than the next job’s appointment date, the system experiences some cost due to under-utilization. We refer to this cost as the *underage* cost. On the other hand, if a job finishes later than the next job’s appointment date, the system experiences *overage* cost due to the overtime of the current job and the waiting of the next job.

The processing durations are given by their joint discrete distribution. Assuming independent discrete processing durations lead to faster algorithms as briefly discussed in Section 8. We assume that this joint distribution is known. Complete information of distributions is reasonable in most settings, but we relax this

assumption in Begen et al. [1]. Our next assumption is a natural one: all cost coefficients and processing durations are non-negative and bounded. We also assume that processing durations are integer valued³. Although we obtain some of our results without this assumption, it is important for our main results.

We assume job 1 starts on-time, i.e., the start time for the first job is zero, and there are n real jobs. The $(n + 1)^{th}$ job is a dummy job with a processing duration of 0. The appointment time for the $(n + 1)^{th}$ job is the total time available for the n real jobs. We use the dummy job to compute the overage or underage cost of the n^{th} job.

Let $\{1, 2, 3, \dots, n, n + 1\}$ denote the set of jobs. We denote the random processing duration of job i by p_i and the random vector⁴ of processing durations by $\mathbf{p} = (p_1, p_2, \dots, p_n, 0)$. Let \underline{p}_i and \bar{p}_i denote the minimum and maximum possible value of processing duration p_i , respectively. The maximum of these \bar{p}_i ’s is $\bar{p}_{\max} = \max(\bar{p}_1, \dots, \bar{p}_n)$. The *underage cost rate* u_i of job i is the unit cost (per time unit) incurred when job i is completed at a date C_i before the appointment date A_{i+1} of the next job $i + 1$. The *overage cost rate* o_i of job i is the unit cost incurred when job i is completed at a date C_i after the appointment date A_{i+1} . Thus the total cost due to job i completing at date C_i is $u_i(A_{i+1} - C_i)^+ + o_i(C_i - A_{i+1})^+$ where $(x)^+ = \max(0, x)$ is the positive part of real number x . We define $\mathbf{u} = (u_1, u_2, \dots, u_n)$ and $\mathbf{o} = (o_1, o_2, \dots, o_n)$. We denote unit vectors in \mathbb{R}^{n+1} as $\mathbf{1}_i$ where the i^{th} component is 1 and all other components are 0.

The underage cost may be interpreted as the idling cost and/or opportunity cost of the resources, whereas the overage cost may be thought as the waiting cost of the next job and/or the overtime of the current job. The overage cost of the last job may include the overtime cost for the whole facility at the end of the schedule after a specified due date.

Next we introduce our decision variables. Let a_i be the allocated duration and A_i the appointment date for job i . Then we have $A_1 = 0$ and $A_{i+1} = A_i + a_i$ for $i = 1, \dots, n$. Thus we may equivalently use the *allocated duration vector* $\mathbf{a} = (a_1, a_2, \dots, a_{n-1}, a_n)$ or the *appointment vector* $\mathbf{A} = (A_1, A_2, \dots, A_n, A_{n+1})$ (with $A_1 = 0$) as our decision variables; we choose to work with the appointment vector \mathbf{A} . We introduce additional variables which help define and compute the objective function. Let S_i be the start date and C_i the completion date of job i . Since job 1 starts on-time we have $S_1 = 0$ and $C_1 = p_1$. The other start

³We can restrict ourselves to integer appointment schedules without loss of optimality by Theorem 5.1.

⁴We write all vectors as row vectors.

times and completion times are determined as follows: $S_i = \max\{A_i, C_{i-1}\}$ and $C_i = S_i + p_i$ for $2 \leq i \leq n+1$. Note that the dates S_i and C_i are random variables which depend on the appointment vector \mathbf{A} .

Let $F(\mathbf{A}|\mathbf{p})$ be the total cost of appointment vector \mathbf{A} given processing duration vector \mathbf{p} :

$$(3.1) \quad F(\mathbf{A}|\mathbf{p}) = \sum_{i=1}^n (o_i(C_i - A_{i+1})^+ + u_i(A_{i+1} - C_i)^+).$$

The objective to be minimized is the expected total cost $F(\mathbf{A}) = \mathbb{E}_{\mathbf{p}} [F(\mathbf{A}|\mathbf{p})]$ where the expectation is taken with respect to random processing duration vector \mathbf{p} . We simplify notations by defining the lateness $L_i = C_i - A_{i+1}$ of job i , its tardiness $T_i = (L_i)^+$, and its earliness $E_i = (-L_i)^+$. The objective $F(\mathbf{A})$ can now be written as

$$F(\mathbf{A}) = \mathbb{E}_{\mathbf{p}} \left[\sum_{i=1}^n (o_i T_i + u_i E_i) \right] = \sum_{i=1}^n (o_i \mathbb{E}_{\mathbf{p}} T_i + u_i \mathbb{E}_{\mathbf{p}} E_i).$$

4 Basic Properties.

We start by making an observation about the completion times and expressing the objective function in a different form that is useful for deriving some of our later results. Since $C_i = S_i + p_i = \max\{A_i, C_{i-1}\} + p_i$, the completion time of job i may be seen as the length of the longest (or critical) path from some job j ($j \leq i$) to job $i+1$ in a corresponding ‘‘project network’’ (Pinedo [12]), namely:

LEMMA 4.1. (*Critical Path*) For all jobs $i = 1, \dots, n$,

$$C_i = \max_{j \leq i} \left\{ A_j + \sum_{k=j}^i p_k \right\}$$

$$F(\mathbf{A}|\mathbf{p}) = \sum_{i=1}^n \left(o_i \left(\max_{j \leq i} \left\{ A_j + \sum_{k=j}^i p_k \right\} - A_{i+1} \right)^+ + u_i \left(A_{i+1} - \max_{j \leq i} \left\{ A_j + \sum_{k=j}^i p_k \right\} \right)^+ \right).$$

Proof. The claim holds trivially for $i = 1$. By induction let the claim be true for $i = m$, i.e., $C_m =$

$\max_{j \leq m} \left\{ A_j + \sum_{k=j}^m p_k \right\}$. Then

$$\begin{aligned} C_{m+1} &= S_{m+1} + p_{m+1} = \max \{A_{m+1}, C_m\} + p_{m+1} \\ &= \max \left\{ A_{m+1}, \max_{j \leq m} \left\{ A_j + \sum_{k=j}^m p_k \right\} \right\} + p_{m+1} \\ &= \max \left\{ A_{m+1} + p_{m+1}, \max_{j \leq m} \left\{ A_j + \sum_{k=j}^{m+1} p_k \right\} \right\} \\ &= \max_{j \leq m+1} \left\{ A_j + \sum_{k=j}^{m+1} p_k \right\} \end{aligned}$$

where the first line is by definition and the second line follows by the inductive assumption. The expression for $F(\mathbf{A}|\mathbf{p})$ follows.

The next result is not only important on its own but also crucial for the existence of an optimal solution.

LEMMA 4.2. (*Continuity*) Functions $F(\cdot|\mathbf{p})$ and $F(\cdot)$ are continuous.

Proof. By expression Eq(3.1), $F(\cdot|\mathbf{p})$ is a weighted sum of piecewise linear continuous functions of \mathbf{A} , hence is itself piecewise linear continuous. Since we have a finite sample space, the expectation $F(\cdot) = \mathbb{E}_{\mathbf{p}} F(\cdot|\mathbf{p})$ is also continuous.

We next establish the existence of an optimal solution. The proof follows from the fact that our objective function is continuous (by Lemma 4.2), and we can restrict the appointment vector to a compact set without loss of optimality. Let $\underline{\mathbf{A}} = (\underline{A}_1, \dots, \underline{A}_{n+1})$ and $\overline{\mathbf{A}} = (\overline{A}_1, \dots, \overline{A}_{n+1})$ where $A_1^* = \underline{A}_1 = \overline{A}_1 = 0$, $\underline{A}_i = \sum_{j < i} \underline{p}_j$ and $\overline{A}_i = \sum_{j < i} \overline{p}_j$ for $i = 2, \dots, n+1$. We define the compact set \mathcal{K} as the cartesian product of the intervals $[\underline{A}_i, \overline{A}_i]$, i.e., $\mathcal{K} = \prod_{i=1}^{n+1} [\underline{A}_i, \overline{A}_i] = [\underline{\mathbf{A}}, \overline{\mathbf{A}}] \subseteq \mathbb{R}^{n+1}$.

LEMMA 4.3. (*Existence of an Optimal Vector*) There exists an appointment vector $\mathbf{A}^* \in \mathcal{K}$ such that $F(\mathbf{A}^*) \leq F(\mathbf{A})$ for any appointment vector \mathbf{A} .

Proof. We show that we can restrict, without loss of optimality, the appointment vector \mathbf{A} to the compact set $\mathcal{K} = [\underline{\mathbf{A}}, \overline{\mathbf{A}}]$ and recall that job 1 starts at time zero, i.e., $A_1 = 0 = \underline{A}_1 = \overline{A}_1$. Consider any appointment vector $\mathbf{A} \notin \mathcal{K}$ with $A_1 = 0$.

If $\mathbf{A} \not\geq \underline{\mathbf{A}}$ then define the appointment vector $\mathbf{A}' = \mathbf{A} \vee \underline{\mathbf{A}}$ with component $A'_i = \max\{A_i, \underline{A}_i\}$. For any realization \mathbf{p} of the processing durations, the completion times C'_i in the resulting schedule satisfy $C'_i = C_i \geq \underline{A}_{i+1}$. (Indeed, $C'_1 = p_1 = C_1 \geq \underline{A}_2$ and, by

induction $C'_i = \max\{A'_i, C'_{i-1}\} = \max\{A_i, \underline{A}_i, C_{i-1}\} = \max\{A_i, C_{i-1}\} = C_i \geq \underline{A}_{i+1}$. Then the resulting tardiness and earliness become: if $A_{i+1} \geq \underline{A}_{i+1}$ then $T'_i = (C'_i - A'_{i+1})^+ = (C_i - A_{i+1})^+ = T_i$ and $E'_i = (A'_{i+1} - C'_i)^+ = (A_{i+1} - C_i)^+ = E_i$; and, if $A_{i+1} < \underline{A}_{i+1}$ then $T'_i = (C'_i - A'_{i+1})^+ = (C_i - \underline{A}_{i+1})^+ \leq (C_i - A_{i+1})^+ = T_i$ and $0 \leq E_i = (A_{i+1} - C_i)^+ \leq (A'_{i+1} - C'_i)^+ = E'_i = 0$ (so $E'_i = E_i = 0$). Since all $u_i, o_i \geq 0$, it follows from Eq(3.1) that $F(\mathbf{A}'|\mathbf{p}) \leq F(\mathbf{A}|\mathbf{p})$ and thus $F(\mathbf{A}') \leq F(\mathbf{A})$. We have shown that for every \mathbf{A} there exists $\mathbf{A}' \geq \underline{\mathbf{A}}$ with $F(\mathbf{A}') \leq F(\mathbf{A})$.

Now, for any vector $\mathbf{A} \in \mathbb{R}^{n+1}$ satisfying $\mathbf{A} \geq \underline{\mathbf{A}}$, $A_1 = 0$ and $\mathbf{A} \notin \mathcal{K}$, let $i(\mathbf{A})$ denote the smallest index such that $A_i > \bar{A}_i$. Let $\mathbf{A} \in \mathbb{R}^{n+1}$ be a vector with largest $i(\mathbf{A})$ value satisfying $\mathbf{A} \geq \underline{\mathbf{A}}$, $A_1 = 0$ and $\mathbf{A} \notin \mathcal{K}$. We claim that there exists \mathbf{A}' satisfying $\mathbf{A}' \geq \underline{\mathbf{A}}$, $A'_1 = 0$, $F(\mathbf{A}') \leq F(\mathbf{A})$, and either $\mathbf{A}' \in \mathcal{K}$ or $i(\mathbf{A}') > i(\mathbf{A})$. Then after at most n such changes we obtain $\mathbf{A}'' \in \mathcal{K}$ satisfying $F(\mathbf{A}'') \leq F(\mathbf{A})$, which is what we wanted to show. We now prove the claim.

Let $i = i(\mathbf{A})$, $\varepsilon = A_i - \bar{A}_i > 0$, and define \mathbf{A}' with $A'_j = A_j$ for all $j \leq i - 1$ and $A'_j = A_j - \varepsilon$ for all $j \geq i$, so $A'_i = \bar{A}_i$. For every realization \mathbf{p} of the processing durations, the completion time C'_j in the resulting schedule satisfy $C'_j = C_j$ for all $j \leq i - 1$. Note that for all $j \leq i - 1$, $A_j \leq \bar{A}_j$ implies $C_j \leq \bar{A}_{j+1}$. Therefore $C_i = A_i + p_i$ and $C'_i = A'_i + p_i = A'_i + C_i - A_i = C_i - \varepsilon$. It follows that $C'_j = C_j - \varepsilon$ for all $j \geq i$. As a result, $E'_j = E_j$ and $T'_j = T_j$ for all $j \neq i - 1$, and $E'_{i-1} = E_{i-1} - \varepsilon$, $T'_{i-1} = T_{i-1} = 0$. Since $\varepsilon > 0$ and $u_{i-1} \geq 0$, $F(\mathbf{A}'|\mathbf{p}) \leq F(\mathbf{A}|\mathbf{p})$ and thus $F(\mathbf{A}') \leq F(\mathbf{A})$. Since $A'_j = A_j \leq \bar{A}_j$ for all $j \leq i - 1$ and $A'_i = \bar{A}_i$, then either $\mathbf{A}' \in \mathcal{K}$ or $i(\mathbf{A}') \geq i + 1 = i(\mathbf{A}) + 1$, establishing the claim. This shows that for any $\mathbf{A} \notin \mathcal{K}$ there exists a vector $\mathbf{A}'' \in \mathcal{K}$ with $F(\mathbf{A}'') \leq F(\mathbf{A})$.

As a result, since F is continuous, its minimum on compact set \mathcal{K} is attained and is therefore the global minimum.

The next lemma gives bounds on the difference between any two consecutive components of an optimal appointment vector, and from this we obtain a useful and intuitive result in Lemma 4.5.

LEMMA 4.4. *There exists an optimal appointment schedule $\mathbf{A}^* \in \mathcal{K}$ satisfying $\underline{p}_i \leq A^*_{i+1} - A^*_i \leq \sum_{j \leq i} \bar{p}_j - \sum_{j < i} \underline{p}_j$ for all $i = 1, \dots, n$.*

Proof. By Lemma 4.3, we immediately obtain $\underline{p}_1 \leq A^*_2 - A^*_1 \leq \bar{p}_1$ and $A^*_{i+1} - A^*_i \leq \sum_{j \leq i} \bar{p}_j - \sum_{j < i} \underline{p}_j$ for all $i = 2, \dots, n$. Next, we show that $\underline{p}_i \leq A^*_{i+1} - A^*_i$ holds for all $i = 2, \dots, n$. By contradiction, suppose $\underline{p}_k + A^*_k > A^*_{k+1}$ for some $k = 2, \dots, n$ then job k is late at least $(\underline{p}_k + A^*_k - A^*_{k+1})$ time units, so increasing

A^*_{k+1} to $\underline{p}_k + A^*_k$ will improve the objective function by $o_k(\underline{p}_k + A^*_k - A^*_{k+1}) \geq 0$. Therefore we must have $\underline{p}_i \leq A^*_{i+1} - A^*_i$ for all $i = 2, \dots, n$.

LEMMA 4.5. (**Non-Decreasing Appointment Dates**) *There exists an optimal appointment vector $\mathbf{A}^* \in \mathcal{K}$ with non-decreasing components, i.e., $A^*_i \leq A^*_{i+1}$ for all $i = 1, \dots, n$.*

Proof. By Lemma 4.4, $A^*_{i+1} - A^*_i \geq \underline{p}_i \geq 0$ ($1 \leq i \leq n$).

5 Optimality of an Integer Appointment Vector.

The existence of an optimal appointment vector which is integer is crucial. It implies that we can restrict attention to integer appointment vectors without loss of optimality. We establish this result in the Appointment Vector Integrality Theorem 5.1. Its proof is surprisingly non-trivial.

Let \mathbf{A}^* be any non-integer appointment vector and A^*_f the first non-integer component of \mathbf{A}^* . Knowing all the jobs which have the same fractional part as A^*_f is crucial, so we define J to be the set of all jobs $j \geq f$ such that $A^*_j - A^*_f$ is integer. Let \mathbb{Z} denote the set of integers, and $\lfloor x \rfloor = \sup\{n \in \mathbb{Z} : n \leq x\}$ and $\lceil x \rceil = \inf\{n \in \mathbb{Z} : n \geq x\}$ for $x \in \mathbb{R}$. Let $\varphi(x)$ be the distance to the nearest integer for $x \in \mathbb{R}$, i.e., $\varphi(x) = \min(x - \lfloor x \rfloor, \lceil x \rceil - x)$. Let Δ be a strictly positive scalar satisfying $0 < \Delta < \frac{1}{2} \min(\Delta_1, \Delta_2)$ where $\Delta_1 = \frac{1}{4} \min\{\varphi(|A^*_j - A^*_k|) : j \in J, k \notin J\} > 0$ and $\Delta_2 = \frac{1}{4} \min\{\varphi(|A^*_j - A^*_k|) : j \notin J, k \notin J, A_j - A_k \notin \mathbb{Z}\} > 0$. We use Δ to construct two new appointment schedules \mathbf{A}' and \mathbf{A}'' from \mathbf{A}^* : let $A'_j = A^*_j - \Delta$ if $j \in J$, and $A'_j = A^*_j$ otherwise; similarly, let $A''_j = A^*_j + \Delta$ if $j \in J$, and $A''_j = A^*_j$ otherwise. For any realization of the processing duration vector \mathbf{p} , denote the completion times of job j as C^*_j, C'_j, C''_j in schedules $\mathbf{A}^*, \mathbf{A}'$ and \mathbf{A}'' , respectively.

One of the main ideas in proving the Appointment Vector Integrality Theorem 5.1 is that Δ is small enough so that there is “no event change” when we move from schedule \mathbf{A}^* to schedules \mathbf{A}' and \mathbf{A}'' . When there is no event change, we show in Lemma 5.9 that our objective function changes linearly between schedules \mathbf{A}' and \mathbf{A}'' . To make the no event change concept precise, we define the following. Job i ($1 < i \leq n + 1$) is *late* if $C^*_{i-1} > A^*_i$ (strictly positive tardiness), *early* if $C^*_{i-1} < A^*_i$ (strictly positive earliness), *just-on-time* if $C^*_{i-1} = A^*_i$, and *on-time* if $C^*_{i-1} \leq A^*_i$. Then no event change means that if any job is late, early or just-on-time, respectively, in schedule \mathbf{A}^* then it is also late, early or just-on-time, respectively, in both schedules \mathbf{A}' and \mathbf{A}'' .

We consider all possible realizations \mathbf{r} of the random processing duration vector \mathbf{p} , so r_i is the corresponding

realization of the processing duration p_i . We start by establishing relationships between the completion times in the schedules \mathbf{A}' and \mathbf{A}^* , and \mathbf{A}'' and \mathbf{A}^* .

LEMMA 5.1. *For every realization of the processing durations and every $j = 1, \dots, n+1$, $C_j^* + \Delta \geq C_j'' \geq C_j^* \geq C_j' \geq C_j^* - \Delta$.*

Proof. Let $1 \leq j \leq n+1$ and let \mathbf{r} be a realization of \mathbf{p} . Then $A_j^* - \Delta \leq A_j' \leq A_j^* \leq A_j'' \leq A_j^* + \Delta$ by definition of \mathbf{A}' and \mathbf{A}'' . By the Critical Path Lemma 4.1, $C_j^* = \max_{k \leq j} \{A_k^* + \sum_{i=k}^j r_i\}$, $C_j' = \max_{k \leq j} \{A_k' + \sum_{i=k}^j r_i\}$ and $C_j'' = \max_{k \leq j} \{A_k'' + \sum_{i=k}^j r_i\}$. Hence, $A_j' \leq A_j^* \leq A_j''$ implies that $C_j' \leq C_j^* \leq C_j''$. On the other hand, $A_j^* - \Delta \leq A_j'$ implies that $C_j^* - \Delta = \max_{k \leq j} \{A_k^* - \Delta + \sum_{i=k}^j r_i\} \leq \max_{k \leq j} \{A_k' + \sum_{i=k}^j r_i\} = C_j'$ so $C_j^* - \Delta \leq C_j'$. Similarly $A_j^* + \Delta \geq A_j''$ implies that $C_j^* + \Delta = \max_{k \leq j} \{A_k^* + \Delta + \sum_{i=k}^j r_i\} \geq \max_{k \leq j} \{A_k'' + \sum_{i=k}^j r_i\} = C_j''$ so $C_j^* + \Delta \geq C_j''$. The result follows.

The next two results are about late and early jobs. Lemma 5.2 below implies that if job k is late (resp., early) then its tardiness (resp., earliness) is strictly greater than 2Δ . Lemma 5.3 implies that if job k is late (resp., early) in schedule \mathbf{A}^* then it is also late (resp., early) in \mathbf{A}' and \mathbf{A}'' .

LEMMA 5.2. *For every realization of the processing durations and every $k = 2, \dots, n+1$, if $|C_{k-1}^* - A_k^*| > 0$ then $|C_{k-1}^* - A_k^*| > 2\Delta$.*

LEMMA 5.3. *For every realization of the processing duration and every $k = 2, \dots, n+1$, if $C_{k-1}^* > A_k^*$ then $C_{k-1}' > A_k'$ and $C_{k-1}'' > A_k''$, and if $C_{k-1}^* < A_k^*$ then $C_{k-1}' < A_k'$ and $C_{k-1}'' < A_k''$.*

Proof. By Lemma 5.2, $C_{k-1}^* > A_k^*$ implies $C_{k-1}^* - A_k^* > 2\Delta$. Note that $A_k^* - \Delta \leq A_k' \leq A_k^* \leq A_k'' \leq A_k^* + \Delta$ by definition, and $C_{k-1}^* + \Delta \geq C_{k-1}'' \geq C_{k-1}^* \geq C_{k-1}' \geq C_{k-1}^* - \Delta$ by Lemma 5.1. Then $C_{k-1}^* - A_k^* > 2\Delta$ implies $C_{k-1}' - A_k' \geq C_{k-1}^* - \Delta - A_k^* > \Delta$ and $C_{k-1}'' - A_k'' \geq C_{k-1}^* - A_k^* - \Delta > \Delta$. Similarly, by Lemma 5.2, $C_{k-1}^* < A_k^*$ implies $A_k^* - C_{k-1}^* > 2\Delta$. Note that $A_k^* - \Delta \leq A_k' \leq A_k^* \leq A_k'' \leq A_k^* + \Delta$ by definition, and $C_{k-1}^* + \Delta \geq C_{k-1}'' \geq C_{k-1}^* \geq C_{k-1}' \geq C_{k-1}^* - \Delta$ by Lemma 5.1. Then $A_k^* - C_{k-1}^* > 2\Delta$ implies $A_k' - C_{k-1}' \geq A_k^* - C_{k-1}^* - \Delta > \Delta$ and $A_k'' - C_{k-1}'' \geq A_k^* - C_{k-1}^* - \Delta > \Delta$. The result follows.

Just-on-time jobs require more care, and we need further definitions and results before we can establish similar results as Lemmata 5.2 and 5.3. Let a *block* $B[t, k]$ be a sequence of consecutive jobs, $[t, t+1, \dots, k]$ ($1 \leq t < k \leq n+1$) such that either $t = 1$ or job t is

early, i.e., $C_{t-1}^* < S_t^* = A_t^*$; no other job in the block is early, i.e., $S_{j+1}^* = C_j^* \geq A_{j+1}^*$ for $j = t+1, \dots, k$; and job k is just-on-time, i.e., $C_{k-1}^* = A_k^*$. Let $K = \{i : t < i \leq k \text{ and } C_{i-1}^* = A_i^*\}$ denote the set of just-on-time jobs in the block $B[t, k]$. So we have $S_t^* = A_t^*$ and $S_j^* = A_j^* = C_{j-1}^*$ for all $j \in K$. Our next result, Lemma 5.4, implies that the first (job t) and all just-on-time jobs in a block (i.e., elements of K) are either all in J or all outside J .

LEMMA 5.4. *If $B[t, k]$ is a block then either $\{t\} \cup K \subseteq J$ or $\{t\} \cup K \subseteq B[t, k] \setminus J$.*

Proof. Let $j \in K$. We have $C_{j-1}^* = A_t^* + \sum_{i=t}^{j-1} r_i$ since t is on-time, and there is no idle time between t and j . We obtain $0 = C_{j-1}^* - A_j^* = A_t^* + \sum_{i=t}^{j-1} r_i - A_j^*$. Since $\sum_{i=t}^{j-1} r_i$ is integer, $A_j^* - A_t^*$ must be integer. This implies that if $j \in J$ then $t \in J$, and if $j \notin J$ then $t \notin J$.

Lemma 5.5 will be used to prove Lemmata 5.6 and 5.7.

LEMMA 5.5. *Let $k \in \{2, \dots, n+1\}$ be such that $A_k^* \notin \mathbb{Z}$. Then for every realization of the processing durations such that $C_{k-1}^* = A_k^*$ there is an early job $j < k$.*

Proof. Let \mathbf{r} be a realization of \mathbf{p} . By contradiction, assume there is no early job before job k . Then $C_{k-1}^* = A_1^* + \sum_{i=1}^{k-1} r_i = A_k^*$. This implies $A_k^* \in \mathbb{Z}$ (since $A_1^* = 0$ and $\sum_{i=1}^{k-1} r_i$ are integer), a contradiction.

In Lemmata 5.6 and 5.7 below we prove that no event change occurs for any just-on-time job. Therefore Lemma 5.8 states that no event change occurs for any job.

LEMMA 5.6. *Let $k \in \{2, \dots, n+1\}$. For every realization of the processing durations such that $C_{k-1}^* = A_k^*$, if there exists an early job $j < k$ then $C_{k-1}' = A_k'$ and $C_{k-1}'' = A_k''$.*

LEMMA 5.7. *Let $k \in \{2, \dots, n+1\}$. For every realization of the processing durations such that $C_{k-1}^* = A_k^*$ we have $C_{k-1}' = A_k'$ and $C_{k-1}'' = A_k''$.*

Proof. If there is an early job before k then the result follows from Lemma 5.6. Otherwise, $B[1, k]$ is a block. Therefore $C_{k-1}^* = A_1^* + \sum_{i=1}^{k-1} r_i = A_k^*$. Furthermore, $A_k^* \in \mathbb{Z}$ by Lemma 5.5 so $k \notin J$. Therefore $\{1\} \cup K \subseteq B[1, k] \setminus J$ by Lemma 5.4, and hence the result follows from Lemma 5.6.

Our next result establishes that no event change occurs for any job and directly follows from Lemmata 5.3 and 5.7. We define the *sign* of a real number x as $\text{sign}(x) = 1$ if $x > 0$; 0 if $x = 0$; and -1 if $x < 0$.

LEMMA 5.8. For every job $j = 2, \dots, n+1$ and every realization of the processing durations, $\text{sign}(C'_{j-1} - A'_j) = \text{sign}(C''_{j-1} - A''_j) = \text{sign}(C^*_{j-1} - A^*_j)$.

Lemma 5.9 below gives a consequence on the objective function of this no event change result.

LEMMA 5.9. F changes linearly with Δ between \mathbf{A}' and \mathbf{A}'' .

Proof. There is no event change when moving from \mathbf{A}' to \mathbf{A}'' by Lemma 5.8. Therefore for every realization \mathbf{r} of the processing duration vector \mathbf{p} , $F(\cdot|\mathbf{p} = \mathbf{r})$ changes linearly with Δ between \mathbf{A}' and \mathbf{A}'' . Hence, $F(\cdot) = \mathbb{E}_{\mathbf{p}}[F(\cdot|\mathbf{p})]$, F also changes linearly with Δ between \mathbf{A}' and \mathbf{A}'' .

THEOREM 5.1. (**Appointment Vector Integrality**) If the processing durations are integer random variables then there exists an optimal appointment vector which is integer.

Proof. By Lemma 4.3 we know that there exists an optimal appointment schedule in the set $\mathcal{K} = \{\mathbf{A} \in \mathbb{R}^{n+1} : \underline{\mathbf{A}} \leq \mathbf{A} \leq \overline{\mathbf{A}}\}$. Let \mathcal{A} denote the set of all such optimal appointment vectors in \mathcal{K} , so \mathcal{A} is nonempty, bounded and closed, since by Lemma 4.2 F is continuous. For $\mathbf{A} \in \mathcal{A}$ let

$$I(\mathbf{A}) = \min\{A_j : j \in \{2, \dots, n+1\} \text{ and } A_j \notin \mathbb{Z}\}$$

if $\mathbf{A} \notin \mathbb{Z}^{n+1}$ and $I(\mathbf{A}) = h+1$ if $\mathbf{A} \in \mathbb{Z}^{n+1}$.

We claim $I(\cdot)$ is upper semi continuous (usc) on the compact set \mathcal{A} . If $\mathbf{A} \in \mathcal{A} \cap \mathbb{Z}^{n+1}$ then $I(\mathbf{A}) = h+1 \geq I(\mathbf{B})$ for all $\mathbf{B} \in \mathcal{A}$, implying that $I(\cdot)$ is usc at \mathbf{A} . Otherwise $\mathbf{A} \in \mathcal{A} \setminus \mathbb{Z}^{n+1}$, and let $I(\mathbf{A}) = A_f$. For any $\epsilon > 0$ let $\delta = \min\{\epsilon, I(\mathbf{A}) - \lfloor A_f \rfloor, \lceil A_f \rceil - I(\mathbf{A})\} > 0$. For all $\mathbf{B} \in \mathcal{A}$, $\|\mathbf{B} - \mathbf{A}\| < \delta$ implies $B_f > A_f - \delta \geq A_f - (I(\mathbf{A}) - \lfloor A_f \rfloor) = \lfloor A_f \rfloor$ and $B_f < A_f + \delta \leq A_f + \lceil A_f \rceil - I(\mathbf{A}) = \lceil A_f \rceil$. Therefore B_f is fractional so $I(\mathbf{B}) \leq B_f \leq A_f + \epsilon = I(\mathbf{A}) + \epsilon$. Therefore $I(\cdot)$ is usc at $\mathbf{A} \in \mathcal{A} \setminus \mathbb{Z}^{n+1}$. This completes the proof that $I(\cdot)$ is usc on \mathcal{A} .

The fact that $I(\cdot)$ is usc and \mathcal{A} is compact implies that there exists an element \mathbf{A}^* of \mathcal{A} maximizing $I(\cdot)$. By contradiction, assume $\mathbf{A}^* \notin \mathbb{Z}^{n+1}$. Let $f = \min\{i : A_i^* = I(\mathbf{A}^*)\}$, so for all $j < f$, $A_j^* < I(\mathbf{A}^*)$ and thus $A_j^* \in \mathbb{Z}$. Let \mathbf{A}' and \mathbf{A}'' be the schedules derived from \mathbf{A}^* as defined at the beginning of this section. By optimality $F(\mathbf{A}^*) \leq F(\mathbf{A}')$ and $F(\mathbf{A}^*) \leq F(\mathbf{A}'')$. But by Lemma 5.9, $F(\mathbf{A}^*)$ changes linearly with Δ between \mathbf{A}' and \mathbf{A}'' . Hence we must have $F(\mathbf{A}^*) = F(\mathbf{A}') = F(\mathbf{A}'')$. Note that $\mathbf{A}'' \geq \mathbf{A}^* \geq \underline{\mathbf{A}}$ and, for every $j \in J$, $A''_j = A^*_j + \Delta < \lceil A^*_j \rceil \leq \overline{A}_j$ so

$\mathbf{A}'' \leq \overline{\mathbf{A}}$. This shows that $\mathbf{A}'' \in \mathcal{K}$ and therefore $\mathbf{A}'' \in \mathcal{A}$. But $I(\mathbf{A}^*) = A^*_f < A^*_f + \Delta = A''_f = I(\mathbf{A}'')$, i.e., $I(\mathbf{A}^*) < I(\mathbf{A}'')$, a contradiction with the definition of \mathbf{A}^* .

REMARK 5.1. Linear overage and underage costs are essential for the integrality of an optimal appointment vector. Consider the following example with quadratic costs. Let $n = 1$ and $F(\mathbf{A}) = \mathbb{E}_{\mathbf{p}} \left[o_1 ((C_1 - A_2)^+)^2 + u_1 ((A_2 - C_1)^+)^2 \right]$ with $o_1 = u_1 = 1$; and $\text{Prob}\{p_1 = 1\} = \text{Prob}\{p_1 = 2\} = \frac{1}{2}$. Then $F(\mathbf{A}) = \mathbb{E}_{\mathbf{p}} [2(C_1 - A_2)^2]$, $C_1 = p_1$, and the optimum is $A_2^* = \mathbb{E}_{\mathbf{p}}(p_1) = \frac{3}{2}$ which is not integer.

6 L-convexity.

We start by investigating an important property of our objective function, submodularity, see Murota [10] and Topkis [16]:

DEFINITION 6.1. A function $f : \mathbb{Z}^q \rightarrow \mathbb{R}$ is submodular iff $f(z) + f(y) \geq f(z \vee y) + f(z \wedge y)$ for all $z, y \in \mathbb{Z}^q$ where $z \vee y = (\max(z_i, y_i) : 0 \leq i \leq q) \in \mathbb{Z}^q$, $z \wedge y = (\min(z_i, y_i) : 0 \leq i \leq q) \in \mathbb{Z}^q$ (Murota [10]).

We now define a property of an appointment vector and a realization of the processing durations that will play an important role in this section.

DEFINITION 6.2. A quadruple (i, j, k, l) is a submodularity obstacle for appointment schedule \mathbf{A} and a realization \mathbf{r} of the processing durations if

- $1 \leq i < j < k < l \leq n+1$;
- the cost coefficients satisfy $o_{j-1} + u_{j-1} + \sum_{j \leq t < k-1} o_t < u_{k-1}$;

and, in schedule $\mathbf{A}|\mathbf{p} = \mathbf{r}$

- both jobs i and j are on-time;
- job l is the last job which starts on-time before job $n+1$;
- there is no idle time between jobs i and j ;
- there is positive idle time between jobs j and l ; and
- job k is the first early job after j .

PROPOSITION 6.1. For any realization \mathbf{r} of the processing durations, the function $F(\cdot|\mathbf{p})$ is submodular if and only if there is no submodularity obstacle for any integer appointment vector \mathbf{A} and realization \mathbf{r} .

COROLLARY 6.1. If there is no submodularity obstacle for any integer appointment vector \mathbf{A} and processing duration realization \mathbf{r} then F is submodular.

Proof. The result holds since submodularity is preserved under expectation, $F(\cdot) = \mathbb{E}_{\mathbf{p}} [F(\cdot|\mathbf{p})]$, and $F(\cdot|\mathbf{p})$ is submodular if there is no submodularity obstacle for any integer appointment vector \mathbf{A} and processing duration realization \mathbf{r} by Proposition 6.1.

Submodularity obstacle is a very specific configuration, and it does not exist with reasonable cost structures such as nonincreasing u_i 's ($u_{i+1} \leq u_i$ for all i) or nonincreasing $(o_i + u_i)$'s ($o_{i+1} + u_{i+1} \leq o_i + u_i$ for all i). To capture these cost structures we define the following.

DEFINITION 6.3. *The cost coefficients (\mathbf{u}, \mathbf{o}) are α -monotone if there exists reals α_i ($1 \leq i \leq n$) such that $0 \leq \alpha_i \leq o_i$ and $u_i + \alpha_i$ are non-increasing in i , i.e., $u_i + \alpha_i \geq u_{i+1} + \alpha_{i+1}$ for all $i = 1, \dots, n-1$.*

The following Lemma establishes a relation between submodularity obstacle and α -monotonicity.

PROPOSITION 6.2. *If the cost coefficients (\mathbf{u}, \mathbf{o}) are α -monotone then there is no submodularity obstacle for any integer appointment vector \mathbf{A} and processing duration realization \mathbf{r} .*

THEOREM 6.1. (*Submodularity*) *If the cost vectors (\mathbf{u}, \mathbf{o}) are α -monotone then F is submodular.*

Proof. If the cost vectors (\mathbf{u}, \mathbf{o}) are α -monotone then there is no submodularity obstacle for any integer appointment vector \mathbf{A} and processing duration realization \mathbf{r} by Proposition 6.2. Hence the result follows from Corollary 6.1.

The objective function is not only submodular but also L-convex, an important discrete convexity property. Before we show L-convexity results, we give the definition of L-convexity.

DEFINITION 6.4. *$f : \mathbb{Z}^q \rightarrow \mathbb{R} \cup \{\infty\}$ is L-convex iff $f(z) + f(y) \geq f(z \vee y) + f(z \wedge y) \quad \forall z, \forall y \in \mathbb{Z}^q$ and $\exists r \in \mathbb{R} : f(z + \mathbf{1}) = f(z) + r \quad \forall z \in \mathbb{Z}^q$ (Murota [10]).*

PROPOSITION 6.3. *For any realization \mathbf{r} of the processing durations, the function $F(\cdot|\mathbf{p})$ is L-convex if and only if there is no submodularity obstacle for any integer appointment vector \mathbf{A} and realization \mathbf{r} .*

Proof. If there is no submodularity obstacle for any integer appointment vector \mathbf{A} and realization \mathbf{r} then $F(\cdot|\mathbf{p})$ is submodular for any realization \mathbf{r} of the processing durations by Proposition 6.1 and submodularity is the first property of L-convexity definition.

Recall that $F(\mathbf{A}|\mathbf{p}) = \sum_{i=1}^n (o_i T_i + u_i E_i)$, $T_i = (C_i - A_{i+1})^+$ and $E_i = (A_{i+1} - C_i)^+$. Consider

$F(\mathbf{A} + \mathbf{1}|\mathbf{p}) = \sum_{i=1}^n (o_i T_i^1 + u_i E_i^1)$, where $x_i^1 =$ quantity of interest of job i with appointment vector $\mathbf{A} + \mathbf{1}$ for $x \in \{S, C, T, E\}$. Then $S_i^1 = S_i + 1$ and $C_i^1 = C_i + 1$ hence $T_i^1 = T_i$ and $E_i^1 = E_i$. Therefore $F(\mathbf{A} + \mathbf{1}|\mathbf{p}) - F(\mathbf{A}|\mathbf{p}) = r = 0$. This gives us the second property of L-convexity definition.

Conversely, if $F(\cdot|\mathbf{p})$ is L-convex then $F(\cdot|\mathbf{p})$ must be submodular for any realization \mathbf{r} of the processing durations and if $F(\cdot|\mathbf{p})$ is submodular then there is no submodularity obstacle for any integer appointment vector \mathbf{A} and realization \mathbf{r} by Proposition 6.1.

COROLLARY 6.2. *If there is no submodularity obstacle for any integer appointment vector \mathbf{A} and realization \mathbf{r} then $F(\cdot)$ is L-convex.*

Proof. The claim holds since L-convexity is preserved under expectation, $F(\cdot) = \mathbb{E}_{\mathbf{p}} [F(\cdot|\mathbf{p})]$, and $F(\cdot|\mathbf{p})$ is L-convex if there is no submodularity obstacle for any integer appointment vector \mathbf{A} and realization \mathbf{r} by Proposition 6.3.

THEOREM 6.2. (*L-convexity*) *If the cost vectors (\mathbf{u}, \mathbf{o}) are α -monotone then $F(\mathbf{A})$ is L-convex.*

Proof. If the cost coefficients (\mathbf{u}, \mathbf{o}) are α -monotone then there is no submodularity obstacle for any integer appointment vector \mathbf{A} and processing duration realization \mathbf{r} by Proposition 6.2. Therefore the result follows from Corollary 6.2.

7 Algorithms.

Using algorithmic results for minimizing L-convex functions, we can minimize expected cost F using a polynomial number of expected cost computations and submodular set minimizations.

Assume the input to our problem consists of the number n of jobs, the cost vectors \mathbf{u} and \mathbf{o} , the horizon h over which F is to be minimized. Assume also that the processing times are integer and that we have an oracle which computes the expected cost $F(\mathbf{A})$ for any given integer appointment vector \mathbf{A} .

THEOREM 7.1. (*Polynomial Time Algorithm*) *If the cost vectors (\mathbf{u}, \mathbf{o}) are α -monotone and the processing durations are integer then there exists an algorithm which minimize F using polynomial time and a polynomial number of expected cost evaluations.*

Proof. The Appointment Vector Integrality Theorem 5.1 implies that we only need to consider integer appointment vectors to minimize F . If the cost vectors (\mathbf{u}, \mathbf{o}) are α -monotone then F is an L-convex function by the L-convexity Theorem 6.2. Then F can be minimized

in $O(\sigma(n)EO n^2 \log(\lceil h/2n \rceil))$ time by Iwata's steepest descent scaling algorithm (Section 10.3.2 of Murota [10]) where $\sigma(n)$ is the number of function evaluations required to minimize a submodular set function over an n -element ground set and EO is the time needed for an expected cost evaluation.

8 Independent Processing Durations, Objective Function with a Due Date., No-shows and Emergency Jobs.

When the processing durations are independent, expected cost of an integer appointment vector can be evaluated efficiently. We use recursive equations for the probability distributions of the start time, completion time, tardiness and earliness of each job and compute F at an integer point \mathbf{A} in $O(n^2 \bar{p}_{\max}^2)$ time and minimize F in $O(n^9 \bar{p}_{\max}^2 \log \bar{p}_{\max})$.

Our modeling framework can handle a given due date D for the total processing (e.g., end of day for an operating room) after which overtime is incurred, instead of letting the model choose an end date A_{n+1} . We assume D is integer and $0 \leq D \leq \sum_{i=1}^n \bar{p}_i$. Define $\tilde{\mathbf{A}} = (A_1, A_2, \dots, A_n)$ then our new objective $F^D(\tilde{\mathbf{A}})$ becomes

$$\mathbb{E}_{\mathbf{P}} \left[\sum_{j=1}^{n-1} \left(o_j (C_j - A_{j+1})^+ + u_j (A_{j+1} - C_j)^+ \right) + o_n (C_n - D)^+ + u_n (D - C_n)^+ \right]$$

We immediately observe that $F(\tilde{\mathbf{A}}, D) = F^D(\tilde{\mathbf{A}})$. Like F , F^D many properties such as discrete convexity (F^D is L^{\natural} -convex), optimal vector integrality and existence of a polynomial time minimization algorithm.

No-shows and emergency jobs may have important practical applications and implications. For example, no-shows can be quite important in certain outpatient exams such as MRI scans. Similarly, emergencies, such as emergency surgeries or examinations, can be a huge factor affecting the planned appointment schedules. With minor modifications and assumptions, our model can handle both no-shows and the insertion of randomly arriving new jobs (e.g., emergency jobs) in finding an optimal appointment schedule.

9 Current Work, Future Work and Conclusion.

After developing our modeling framework and proving that we can find an optimal appointment schedule in polynomial time, we are now focusing on practical implementation issues. Our objective as a function of continuous appointment vector is non-smooth but we have shown that it is convex, and we characterized its

subdifferential. We have obtained closed form formulas for the subdifferential as well as for any subgradient. This characterization is very useful, it allows us to develop two very important extensions.

In the first extension, Begen et al. [1], we relax the perfect information assumption on the probability distributions of processing durations, i.e., we assume that processing duration distributions are not known and can only be statistically estimated on the basis of past data or statistical sampling. Our approach is non-parametric, and we assume no (prior) information about processing duration distributions. We develop a sample-based approach to determine the number of independent samples required to obtain a provably near-optimal solution with high confidence, i.e., the cost of the sample-based optimal schedule is with high probability no more than $(1 + \epsilon)$ times the cost of an optimal schedule determined from knowing the true distributions. This result has important practical implications, as the true processing duration distributions are often not known and only their past realizations or some samples are available.

In another study, Begen and Queyranne [2], we use the subdifferential characterization with independent processing durations and compute a subgradient in polynomial time for any given appointment schedule. This is not a trivial task as the subdifferential formulas include exponentially many terms, and some of the probability computations are complicated. We also obtain an easily computable lower bound on the optimal objective value. Furthermore, we extend computation of the expected total cost (in polynomial time) for any (real-valued) appointment vector. These allow us to use non-smooth convex optimization techniques to find an optimal schedule. Although we already have a polynomial time algorithm to find an optimal appointment schedule, it is not clear at the moment which technique will work faster in practice. We are also considering hybrid algorithms based on both discrete convexity and non-smooth convex optimization combined with a special-purpose integer rounding method. Preliminary versions of these algorithms have been developed. The rounding algorithm takes any fractional solution and rounds it to an integer one with the same or improved objective value. We are planning to implement our algorithms and compare different approaches in computational experiments.

There are many exciting future directions for this research. One is to find an optimal sequence and appointment schedule simultaneously, i.e., given the jobs, determine a sequence and a job appointment schedule minimizing the total expected cost. This problem is likely to be hard, but it may be possible to de-

velop heuristic algorithms with performance guarantees. Studying some special cases for this problem may shed light on the general case. Another one is to put our findings into practice. We are in contact with local health-care organizations to apply our results with real data and compare the appointment schedules determined by of our methods with current practices.

In this paper, we study a discrete time version of the model and establish discrete convexity properties of the objective function. We prove that the objective function is L-convex under mild assumptions on cost coefficients. Furthermore, we show that there exists an optimal integer appointment schedule minimizing the objective. This result is important as it allows us to optimize only over integer appointment schedules without loss of optimality. All these results on the objective function and optimal appointment schedule enable us to develop a polynomial time algorithm, based on discrete convexity, that, for a given processing sequence, finds an appointment schedule minimizing the total expected cost. When processing durations are stochastically independent we evaluate the expected cost for a given processing order and an integer appointment schedule, efficiently both in theory (in polynomial time) and in practice (computations are quite fast in our preliminary computational experiments). Independent processing durations lead to faster algorithms. Our modeling framework can handle a given due date for the total processing (e.g., end of day for an operating room) after which overtime is incurred, instead of letting the model choose an end date. We also extend our model and framework to include no-shows and emergencies. We believe that our framework is sufficiently generic so that it is portable and applicable to many appointment systems in healthcare as well as in other areas.

References

- [1] M. A. Begen, R. Levi, and M. Queyranne, *A sampling-based approach to appointment scheduling*, Working Paper, Sauder School of Business, University of British Columbia, (2008).
- [2] M. A. Begen and M. Queyranne, *Minimizing a discrete-convex function for appointment scheduling*, Working Paper, Sauder School of Business, University of British Columbia, (2008).
- [3] P. M. V. Bosch, D. C. Dietz, and J. R. Simeoni, *Scheduling customer arrivals to a stochastic service system*, Naval Research Logistics, 46 (1999), pp. 549–559.
- [4] T. Cayirli and E. Veral, *Outpatient scheduling in health care: A review of literature*, Production and Operations Management, 12 (2003).
- [5] B. Denton and D. Gupta, *A sequential bounding approach for optimal appointment scheduling*, IIE Transactions, 35 (2003), pp. 1003–1016.
- [6] S. Fujishige, *Submodular Functions and Optimization*, Elsevier, 2005.
- [7] S. Iwata, *Submodular function minimization*, Math. Program., 112 (2008), pp. 45–64.
- [8] G. C. Kaandorp and G. Koole, *Optimal outpatient appointment scheduling*, Health Care Man. Sci., 10 (2007), pp. 217–229.
- [9] S. T. McCormick, *Submodular function minimization. a chapter in the handbook on discrete optimization*, Elsevier, K. Aardal, G. Nemhauser, and R. Weismantel, eds, (2006).
- [10] K. Murota, *Discrete Convex Analysis*, SIAM, 2003.
- [11] J. B. Orlin, *A faster strongly polynomial time algorithm for submodular function minimization*, to appear in Math Programming, (2007).
- [12] M. Pinedo, *Scheduling: Theory, Algorithms, and Systems*, Prentice Hall, 2001.
- [13] L. W. Robinson and R. R. Chen, *Scheduling doctors' appointments: optimal and empirically-based heuristic policies*, IIE Transactions, 35 (2003), pp. 295–307.
- [14] L. W. Robinson, Y. Gerchak, and D. Gupta, *Appointment times which minimize waiting and facility idleness*, Working Paper, DeGroote School of Business, McMaster University, (1996).
- [15] F. Sabria and C. F. Daganzo, *Approximate expressions for queuing systems with scheduling arrivals and established service order*, Transportation Science, 23 (1989), pp. 159–165.
- [16] D. M. Topkis, *Minimizing a submodular function on a lattice*, Oper. Res., 26 (1978), pp. 305–321.
- [17] P. P. Wang, *Static and dynamic scheduling of customer arrivals to a single-server system*, Naval Research Logistics, 40 (1993), pp. 345–360.
- [18] —, *Sequencing and scheduling n customers for a stochastic server*, European Journal of Operations Research, 119 (1999), pp. 729–738.
- [19] E. N. Weiss, *Models for determining estimated start times and case orderings in hospital operating rooms*, IIE Transactions, 22 (1990), pp. 143–150.
- [20] P. Zipkin, *On the structure of lost-sales inventory models*, Oper. Res., Published Online, (2008).