Title
Incentives and Strategic Choices In The Secretary Problem

Author
Truong, Nguyen Le

Publication Date
2013

Incentives and Strategic Choices In The Secretary Problem

by

Nguyen Le Truong

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Engineering - Industrial Engineering and Operations Research

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Ilan Adler, Chair
Professor Shachar Kariv
Professor Zuo-Jun Max Shen

Fall 2013

Incentives and Strategic Choices In The Secretary Problem

Abstract

Incentives and Strategic Choices In The Secretary Problem

by

Nguyen Le Truong

Doctor of Philosophy in Engineering - Industrial Engineering and Operations Research

University of California, Berkeley

Professor Ilan Adler, Chair

Optimal policies for various secretary problems have an undesirable trait: they would interview applicants for the position, but those *earlier* ones are guaranteed to not get selected. Therefore, early applicants have incentive to not come in for their scheduled interviews, and as a direct consequence, the employer's intention to learn from the population becomes useless. Prior works have been done that tried to mitigate this issue, where the employer sacrifices her overall probability of selecting the best applicant by assigning equal selection probability to all interview slots. Among our results, we show such approaches can be costly for an employer with objectives different from the classical one. Furthermore, we generalize the classical setting to allow applicants to make independent choices with regard to their time of availability. This new game-theoretic approach solves the interviewing-without-hiring problem that arose earlier, and surprisingly, improves the employer's probability of selecting the best applicant from one obtained in the classical setting.

To Mom, Dad, and Bé Ti.

Contents

Acknowledgments

Words alone are not enough to convey how thankful I am for the following people in my life. This dissertation, in one way or another, was shaped by their generosity, patience, love, and friendship.

My advisor, Ilan Adler, for allowing me to follow my interests and explore on my own. His advices have served as my trustworthy compass, and his insights my terrain map for navigating this jungle of academia in the last 6 years.

My committee members, Shachar Kariv and Max Shen, for teaching me fundamental ideas in game theory and optimization, both central to this work. Their courses are also among the few that I thoroughly enjoyed while being a student here at Berkeley. Thank you for being great teachers!

My office mates and Berkeley friends, Gemma Berenguer, Dan Bu, Huaning Cai, Shea Chen, Yifen Chen, Kelly Choi, Shiman Ding, Qiaochu He, Tuan Hoang, Phoebe Lai, Stewart Liu, Cheng Lu, Amber Richter, Gah-yi Vahn, K.C. Yang, Peng Yi, and Yan Zhang for listening to my rants, holding midnight conversations on random subjects, hiking known explored wilderness, running in the rain, and countless other experiences that best not be brought up here. I had fun with you all, and I hope you did with me, too!

My sister, for being a steady cheerleader throughout.

And my parents, for everything that they have done, and would have done, for me.

Thank you all! I love you!

Applause Here.

Chapter 1

Introduction

1.1 A Mathematical Puzzle

We begin by first describing what is known as the secretary problem. It first appeared as a puzzle in Martin Gardner [6]'s Scientific American column in 1960, and had been studied extensively in the mathematical research community ever since. Its description below can be found in [5]:

- There is one secretarial position available.
- The number n of applicants is known.
- The applicants are interviewed sequentially in random order, each order being equally likely.
- It is assumed that you can rank all the applicants from best to worst without ties. The decision to accept or reject an applicant must be based only on the relative ranks of those applicants interviewed so far.
- An applicant once rejected cannot later be recalled.
- You are very particular and will be satisfied with nothing but the very best. (That is, your payoff is 1 if you choose the best of the n applicants and 0 otherwise.)

What should you do?

1.2 Solution To Puzzle

We present here a solution proposed by Gilbert and Mosteller [7]. Suppose we are interviewing the ith applicant. We should choose this ith applicant now if the probability that he is **best overall** *exceeds* the same probability obtain by the **best** strategy available by continuing on. It is clear that, at each state, we need only consider the best applicant **so far**, as any other cannot be the best overall.

1. Let $f(i) = \Pr[\text{select best overall with } i\text{th applicant} \mid i\text{th applicant is best so far}]$. Observe that $f(i) = \frac{i}{n}$, and hence, is increasing in i.
2. Let $g(i+1) = \Pr[\text{select best overall with best strategy from } (i+1)\text{st applicant on}]$. Observe that $g(i)$ is decreasing in i, as we can withhold from choosing in an earlier period and employ the best strategy in a later period, guaranteeing a result equally as good.

With $f(i)$ an increasing function in i, $g(i)$ a decreasing in i, and we choose an applicant whenever $f(i) > g(i+1)$, it follows that the optimal strategy has the form *bypass the first* $(s-1)$ *applicants, and choose the best one thereafter*. Here, s is the smallest value such that $f(s) > g(s+1)$.

Let us now focus only on these threshold-type strategies. The probability that we select the best overall applicant given that we bypassed the first $(s-1)$ is $\frac{s-1}{n} \sum_{k=s}^{n} \frac{1}{k-1}$. To see this, simply observe that in order to select the best overall, this best applicant must appear in a slot $s \leq k \leq n$, and the best of the first $(k-1)$ applicants must come in the first $(s-1)$ slots. This event occurs with probability $\frac{s-1}{k-1} \cdot \frac{1}{n}$, and summing over the range for k yields the desired result.

The above discussion leads us to the following theorem.

Theorem 1. *In the secretary problem, the optimal strategy is to skip over the first* $(s-1)$ *applicants, and choose the best one thereafter. Here,* s *is the minimum value such that*

$$\frac{s}{n} \geq \frac{s}{n} \sum_{k=s+1}^{n} \frac{1}{k-1} \iff 1 \geq \sum_{k=s+1}^{n} \frac{1}{k-1}$$

When n *becomes sufficiently large, the optimal threshold is approximately* $\frac{n}{e}$*, and the probability for selecting the best overall applicant is about* $\frac{1}{e}$*.*

where the last two claims in the theorem can be derived using integral calculus.

1.3 The Online Decision-Making Framework And Other Applications

In reality, an employer oftentimes does not have to make a hiring decision immediately after an interview. As such, one may cast doubt on the general framework of the secretary problem as being unrealistic. We concur with this assessment, and will delve deeper into the issue later in this dissertation. In the mean time, we also make the observation that applicants generally do not give the employer unlimited time to make her decision either, and the classical setting of the secretary problem provides a structured, albeit extreme, way to approach this problem. It is a nice starting point to begin our analysis.

We also note that other real world applications are highly relevant to the secretary problem, and present two below. Both belong to the e-commerce sector of the business world, and have been studied extensively in the literature. We will restrict ourselves to the hiring setting throughout this dissertation, but one should be aware of potential applications of the secretary problem to other places.

eBay is best known in the United States as an auction house where everything imaginable on Earth is being put up for sale daily. One of the key features for this online site is the *Best Offer* option, where arriving customers have an opportunity to offer a specific amount of money in return for the listed item. Once an offer is received, the seller can choose to accept it and end the listing, or reject it and continue with the process. Inherent to this problem is the sequential nature of customers' arrivals, which lends its similarity to assumptions in the secretary problem. One objective that the seller may have on mind is to maximize the probability of selecting the highest paying customer, a goal shared by the employer in our classical secretary setting. Since customers generally allow the seller sometime to make her decision, our game-theoretic framework (to be presented in Chapter 4) fits nicely into this general setting.

The secretary problem has also been used in other papers to model online auctions in various settings. We mention here the setup for an online auction problem studied in [1]. Consider a seller who wishes to auction off a single item to n bidders. Each bidder i has an arrival time $a_i \in [0, T]$ and a valuation v_i, both of which are private information. The number of bidders n, and the time horizon T, are common knowledge to everyone. The bidder may arrive at any time $t_i \geq a_i$, and places a bid b_i for the item, which can be different from his valuation v_i. The mechanism then gets to decide whether to allocate the good to this bidder, and if so, the price this bidder will be charged with. It can be assumed that the bidder's utility function takes the form $v_i - p_i$, where p_i is the amount he has to pay. By placing certain assumptions on the set of arrival times $\{a_1, a_2, \ldots, a_n\}$ and valuations $\{v_1, v_2, \ldots, v_n\}$, the authors proceed to design and optimize secretary-based auction mechanisms that are truthful with respect to both valuation and arrival at the assigned time. The maximization objective can be used for efficiency (probability that the mechanism allocates the good to the highest bidder) or revenue (the expected price charged).

1.4 What's Next?

The general outline for this dissertation is as follows. In Chapter 2, we will present linear programs that can be used to solve two different variants of the secretary problem, and show how to obtain such linear programs using two different approaches: from first principles, and from tools readily available in Markov Decision Processes. Variables in these linear programs allow us to interpret decisions probabilistically, which are fundamental in introducing incentives to participants of a sequential decision process.

Incentive compatibility is taken from works by Buchbinder et al. [2], and extended to another well-known variant of the secretary problem in Chapter 3. From this rank-based variant, we obtain conclusions which contrast those found by [2] for the classical setting. One such result, in particular, shows a jump from constant objective value to $\Theta(\log(n))$ when incentive compatible constraints are added. Another shows the same $\Theta(\log(n))$ objective value even when incentive compatibility constraints are relaxed further.

In Chapter 4, we present new game-theoretic models for the classic variant of the secretary problem, one being an extensive game with imperfect information, and the other a strategic, simultaneous-move game. In the extensive game, we present the dominant strategy for all applicants, and show an improvement in the employer's objective value when everyone plays this strategy. In the strategic game, among results that we will show is the existence of a pure-strategy equilibrium for all cases of $n \neq 4$. Our new game-theoretic models also address concerns for incentive compatibility brought forth in [2].

Chapter 5 wraps up our dissertation.

Chapter 2

Linear Programming And The Secretary Problem

2.1 Background

Buchbinder et al. [2] introduced a linear programming representation for several variants of the secretary problem (henceforth will be referred to as BJS-approach). In their formulations, decision variables are probabilistic, and each can be interpreted as the probability a feasible hiring mechanism will select someone satisfying certain conditions. This chapter is built on their works, where we show how to obtain such linear programs for two important variants of the secretary problem. One is fundamental for our analysis in Chapter 3, the other plays an important role in the game-theoretic approach in Chapter 4. Furthermore, we will obtain these representations using two different techniques. One approach uses first principles as demonstrated by [2], whereas another uses tools readily available in the theories of linear programming and Markov Decision Processes. We believe our insight will be helpful in cases where it is intuitively hard to come up with decision variables as in [2], due to the fact that our proposed process is mechanical in nature.

2.2 The Rank-Based Secretary Problem

An employer with the classic objective of hiring the best overall applicant is being picky in an extreme way. She is never satisfied with anyone but the best, and may come away from the hiring process empty-handed. This may be okay in certain hypothetical situations, but is not necessary in many others.

In the rank-based secretary problem, there are n applicants who apply for one available job. If we are allowed to observe them all, we would be able to rank them individually, from best (rank 1) to worst (rank n). Suppose we assign these applicants to interview slots in a random order, and when the ith applicant is interviewed, we can only observe his rank relative to those who came in before he did. We must make our hiring decision online: either accept him and end the job search, or reject him and continue with the job search. Our goal in this rank-based setting is to find a hiring strategy which minimizes the expected rank of the hiree.

Implicit in the statement of the problem is that not hiring anyone would yield a rank of ∞ or $n+\epsilon$, where $\epsilon > 0$ is any positive constant. In other words, hiring no one is worse than hiring the worst applicant. All together, this objective function leads to a neat, easy-to-show (to be done in chapter 3) property in the optimal hiring policy: that it will always select someone. In this dissertation, we choose to assign an objective value of $n+1$ in the event no one is selected. Choosing $n+1$ over others, say ∞, will have implications in incentive compatible settings to be explored in a later chapter.

Lindley [8] was the first to consider this version of the secretary problem. He was able to derive a recurrence equation characterizing the optimal threshold policy, which is of the form:

while interviewing for the rth applicant, stop and accept if her apparent rank $s \leq s^*(r)$, continue if $s > s^*(r)$.

The recurrence equation that needs to be solved to get $s^*(r)$ is fairly complex, and it was Chow et. al. [3] who successfully showed the optimal stopping policy chooses an applicant with expected rank $\prod_{j=1}^{\infty}\left(\frac{j+2}{j}\right)^{1/(j+1)} = 3.8695$ as $n \to \infty$.

In the rank-based secretary problem, we seek to minimize the expected rank of the selected applicant. Observe that this objective is *equivalent* to *maximizing the expected utility* of the selected applicant, where the utility for hiring an ith-rank applicant is $(n+1-i)$. Let r_n^* denote the optimal expected rank of the hiree when there are n applicants, and u_n^* the optimal expected utility of the hiree when there are n applicants. It is evident that the optimal expected rank $r_n^* = n+1-u_n^*$. We choose to work with this alternative problem of maximizing the expected utility from now on.

LP Formulation For The Utility-Based Secretary Problem

Our first objective is to derive the linear programming formulation for the incentive compatible utility-based secretary problem. We will give two different approaches to this result. The first uses known results in the theory of Markov Decision Process and linear programming, and is shown below. From the recurrence for this optimal stopping problem, we will construct an equivalent linear program (D'). We then make an appropriate substitution of variables to obtain a new linear program (D). Taking the dual of the linear program (D) will yield the desired LP (P) below. We also note that (P) is the linear program obtained directly by using the approach as shown in [2]. This second, alternative approach will be expounded in a later section.

From our approach, stop-continue binary decisions in the recursion are transformed into probabilistic decision variables. Although the nature of the secretary problems dictates a 0-1 solution, the newer probabilistic variables come in handy when we consider adding incentive compatible constraints, whose optimal solutions will no longer be binary. We will delay this discussion until the next chapter.

Proposition 1. *The following linear program is a formulation of the utility-based secretary problem.*

$$
\begin{array}{llll}
 & max & \sum_{i=1}^{n}\sum_{j=1}^{i}\left(\frac{(n+1)(i-j+1)}{i(i+1)}\right)q_i^j & \\
(P) & s.t. & q_i^j \leq 1-\sum_{l<i} p_l & 1 \leq j \leq i \leq n \\
 & & p_i = \frac{1}{i}\sum_{k=1}^{i} q_i^k & 1 \leq i \leq n \\
 & & p_i, q_i^j \geq 0 &
\end{array}
$$

Proof. Consider a Markov Decision Process where the underlying state (i, j) denotes the employer is interviewing the ith applicant given he is jth best so far. The employer has two possible actions, **stop** or **continue**. Let $V_{i,j}$ denotes the optimal value at state (i, j), then we must have the following (e.g. a simple modification found in [4]):

$$V_{i,j} = \max\left\{n+1-\frac{n+1}{i+1}j,\ \frac{1}{i+1}\sum_{l=1}^{i+1} V_{i+1,l}\right\}, \qquad 1 \leq j \leq i \leq n$$

and that $V_{n+1,j} = 0$ for $1 \leq j \leq n+1$.

The above recurrence can be formulated as the following linear program (D'):

$$
\begin{array}{lllll}
 & \text{min} & V_{1,1} & & \\
(D') & \text{s.t.} & V_{i,j} & \geq & n+1-\frac{n+1}{i+1}j \quad 1 \leq j \leq i \leq n \\
 & & V_{i,j} & \geq & \frac{1}{i+1}\sum_{l=1}^{i+1} V_{i+1,l} \quad 1 \leq j \leq i \leq n \\
 & & V_{n+1,j} & = & 0 \quad 1 \leq j \leq n+1
\end{array}
$$

Consider the transformation $i \cdot x_{i,j} = V_{i,j} - \frac{1}{i+1}\sum_{l=1}^{i+1} V_{i+1,l}$ for all $1 \leq j \leq i \leq n$. By induction, we can show that

$$V_{i,j} = i \cdot x_{i,j} + \sum_{l=i+1}^{n}\sum_{j=1}^{l} x_{l,j}, \qquad 1 \leq j \leq i \leq n$$

Let us denote $iy_i = -\sum_{l=i+1}^{n}\sum_{j=1}^{l} x_{l,j}$, then (D') can be re-written as:

$$
(D) \quad \begin{array}{llcll}
\min & \sum_{i=1}^{n}\sum_{j=1}^{i} x_{i,j} & & & \\
\text{s.t.} & x_{i,j} - y_i & \geq & \frac{(n+1)(i-j+1)}{i(i+1)} & 1 \leq j \leq i \leq n \\
& \sum_{l=i+1}^{n}\sum_{j=1}^{l} x_{l,j} + iy_i & = & 0 & 1 \leq i \leq n \\
& x_{i,j} \geq 0,\ y_i \text{ free} & & & 1 \leq j \leq i \leq n
\end{array}
$$

But observe that this newly transformed linear program is the dual of the desired LP (P) by letting q_i^j's to be dual variables corresponding to inequality constraints, and p_i's to be dual variables corresponding to equality constraints.

□

We note the necessity of introducing q_i^j's into the model, as the dynamic program requires knowing an applicant's relative rank at each stage to determine an optimal policy. In this way, different probability variables are needed for different problems. In particular, they must be dependent on the decision maker's objective function and constraints. We choose to delay elaborating on this point until the next section, where another example will be exhibited.

We also note that this proof does not allow us to directly interpret p_i's and q_i^j's as probabilities of selecting the ith applicant, and probabilities of selecting the ith applicant given he is jth best so far, respectively. To give proper justifications, we need to use the type of proof as outlined in the next section.

BJS-Derivation of LP For the Utility-Based Secretary Problem

The approach here is similar to that found in Buchbinder et al.'s paper [2]. The first step involves showing that all mechanisms must satisfy a certain set of linear constraints, and the corresponding linear program gives an objective value which is **at least** that of the mechanism's. The second step shows the converse, i.e. from a feasible solution to the linear program, construct a mechanism which selects an applicant with expected utility **at least** as high as that in the LP objective. These two steps then imply the problem of finding an optimal mechanism is equivalent to that of solving a particular linear program.

Lemma 1. *Take any mechanism π for selecting applicants. Let q_i^j denote the probability π selects the ith applicant given that she is jth best so far. Let p_i denote the probability π selects the ith applicant. Then the linear program below gives an upper bound to the expected utility of the applicant that π selects:*

$$
\begin{array}{lll}
max & \frac{1}{n}\sum_{i=1}^{n}\sum_{s=1}^{n}\sum_{j=1}^{s}(n+1-s)\frac{\binom{i-1}{j-1}\binom{n-i}{s-j}}{\binom{n-1}{s-1}}q_i^j & \\
s.t. & q_i^j \leq 1-\sum_{l<i} p_l & 1 \leq j \leq i \leq n \\
 & p_i = \frac{1}{i}\sum_{k=1}^{i} q_i^k & 1 \leq i \leq n \\
 & p_i, q_i^j \geq 0 &
\end{array}
$$

Proof. We shall first derive the objective function, and constraints afterward.

1. Let U_π be the random variable denoting the utility of hiring an applicant. Also define f_i^s to be the probability a mechanism π selects the ith applicant given she is sth best overall. Then:

$$
\begin{aligned}
\mathbb{E}[U_\pi] &= \sum_{i=1}^{n}\sum_{s=1}^{n}\mathbb{E}[U_\pi \mid \pi \text{ selects } i,\ \pi \text{ did not select } 1,\ldots,i-1;\ \ i \text{ is } s\text{th best overall}] \\
&\qquad \cdot \Pr[\pi \text{ selects } i \mid \pi \text{ did not select } 1,\ldots,i-1;\ \ i \text{ is } s\text{th best overall}] \\
&\qquad\qquad \cdot \Pr[\pi \text{ did not select } 1,\ldots,i-1;\ \ i \text{ is } s\text{th best overall}] \\
&= \sum_{i=1}^{n}\sum_{s=1}^{n}(n+1-s)\cdot \Pr[\pi \text{ selects } i \mid i \text{ is } s\text{th best overall}] \\
&\qquad\qquad \cdot \Pr[i \text{ is } s\text{th best overall}] \\
&= \frac{1}{n}\sum_{i=1}^{n}\sum_{s=1}^{n}(n+1-s)\cdot f_i^s
\end{aligned}
$$

Next, we will show $f_i^s = \sum_{j=1}^{s}\frac{\binom{i-1}{j-1}\binom{n-i}{s-j}}{\binom{n-1}{s-1}}q_i^j$, which completes the argument. Observe that:

$$
\begin{aligned}
f_i^s &= \Pr[\pi \text{ selects } i \mid i\text{th applicant is } s\text{th best overall}] \\
&= \sum_{j=1}^{s}\Pr[\pi \text{ selects } i \mid i\text{th applicant is } s\text{th best overall}, \\
&\qquad\qquad\qquad i\text{th applicant is } j\text{th best so far}] \\
&\qquad \cdot \Pr[i\text{th applicant is } j\text{th best so far} \mid i\text{th applicant is } s\text{th best overall}] \\
&= \sum_{j=1}^{s}\Pr[\pi \text{ selects } i \mid i\text{th applicant is } j\text{th best so far}] \\
&\qquad \cdot \Pr[i\text{th applicant is } j\text{th best so far} \mid i\text{th applicant is } s\text{th best overall}] \\
&= \sum_{j=1}^{s} q_i^j \cdot \frac{\binom{i-1}{j-1}\binom{n-i}{s-j}}{\binom{n-1}{s-1}}
\end{aligned}
$$

Here, the third equality follows because a mechanism can only discern at position i whether this applicant is jth best so far or not. The information that she is sth best overall is irrelevant.

To see the fourth equality, observe that out of $n-1$ positions, we must choose $s-1$ that are of lower rank than the ith position; there are $\binom{n-1}{s-1}$ ways to do this. For the numerator, with the ith element being jth best so far, among the first $i-1$ positions, choose $j-1$ to be of lower rank; there are $\binom{i-1}{j-1}$ ways. Also, among the other $n-i$ positions that come after the ith applicant, choose $s-j$ positions to be occupied by the rest of the applicants with smaller rank than the ith applicant. This gives $\binom{n-i}{s-j}$ ways, and completes the argument.

2.

$$\begin{aligned} q_i^j &= \Pr[\pi \text{ selects } i \mid i\text{th applicant is } j\text{th best so far}] \\ &\le \Pr[\pi \text{ did not select } 1, 2, \ldots, i-1 \mid i\text{th applicant is } j\text{th best so far}] \\ &= 1 - \sum_{l<i} \Pr[\pi \text{ selects lth applicant} \mid i\text{th applicant is } j\text{th best so far}] \\ &= 1 - \sum_{l<i} \Pr[\pi \text{ selects lth applicant}] \\ &= 1 - \sum_{l<i} p_l \end{aligned}$$

3.

$$\begin{aligned} p_i &= \Pr[\pi \text{ selects } i] \\ &= \sum_{j=1}^{i} \Pr[\pi \text{ selects } i \mid i\text{th applicant is } j\text{th best so far}] \\ &\qquad \cdot \Pr[i\text{th applicant is } j\text{th best so far}] \\ &= \sum_{j=1}^{i} q_i^j \cdot \frac{1}{i} \end{aligned}$$

□

Lemma 1 shows any hiring mechanism π must satisfy feasibility for a particular linear program, and its performance is upper bounded by the objective function of that linear program. In the next lemma, we show how to construct a hiring mechanism from a solution of the linear program. Together, these two lemmas show a one-to-one correspondence between mechanisms and LP feasible solutions, and thus completes the proof for one-to-one correspondence between LP's feasible solutions and hiring mechanisms.

Lemma 2. *Let the pair (p_i, q_i^j) be a feasible solution to the linear program presented earlier. Consider the mechanism μ which, given it has not selected applicants $1, 2, \ldots, i-1$ and the ith applicant is jth best so far, picks the ith applicant with probability $\frac{q_i^j}{1-\sum_{l<i} p_l}$. Then the expected utility of the hired applicant for which μ selected is that of the objective value:*

$$\frac{1}{n} \sum_{i=1}^{n} \sum_{s=1}^{n} \sum_{j=1}^{s} (n+1-s) \frac{\binom{i-1}{j-1}\binom{n-i}{s-j}}{\binom{n-1}{s-1}} q_i^j$$

Proof. We first show the probability of μ selecting applicant i **given** that this ith applicant is jth best so far is q_i^j. Furthermore, the probability of μ selecting the ith applicant is $p_i = p$. We proceed by induction on i. Once proved, it is easily seen that μ selects an applicant with expected rank given by the objective value in the linear program, by the argument in the previous lemma.

- $i = 1$: This is trivially true, since no applicants appear before the 1st, and the 1st applicant must be the best so far, it follows that q_1^1 is the probability μ selects applicant 1 **given** applicant 1 is the best so far.
- $i \leq k$: Assume for all $1 \leq j \leq i \leq k$, q_i^j is the probability that μ selects the ith applicant **given** the ith applicant is jth best so far. Also assume p_i is the probability μ selects the ith applicant.
- $i = k+1$: Note that

$$\begin{aligned} q_{k+1}^j &\stackrel{?}{=} \Pr[\mu \text{ selects } (k+1)\text{th applicant} \mid (k+1) \text{ is } j\text{th best so far}] \\ &= \Pr[\mu \text{ selects } (k+1)\text{th applicant} \mid 1, 2, \ldots, k \text{ not selected}, \\ &\qquad\qquad\qquad\qquad (k+1)\text{th is } j\text{th best so far}] \\ &\qquad \cdot \Pr[\mu \text{ did not select } 1, 2, \ldots, k \mid (k+1)\text{th is } j\text{th best so far}] \\ &= \frac{q_{k+1}^j}{1 - \sum_{l<k+1} p_l} \cdot \left(1 - \sum_{l<k+1} p_l\right) \\ &= q_{k+1}^j \end{aligned}$$

where the third inequality comes from the inductive assumption. As such, q_i^j denotes the probability μ selects the ith applicant given he is the jth best so far. It is also clear from the interpretation of q_i^j's that since $p_{k+1} = \sum_{j=1}^{k+1} q_{k+1}^j \cdot \frac{1}{k+1}$, the LHS denotes the probability of μ selecting the $(k+1)$th applicant. This completes our proof.

□

Thus, we have reduced the problem of finding the optimal incentive compatible hiring mechanism to that of solving a linear program. We note that q_i^j's cannot be used right away to obtain hiring decisions that are made by solving a simple dynamic program, and instead needs to be manipulated by conditioning on events of one's rank is jth best so far out of i, and that the decision maker has not made her decision up to that point. The next subsection shows how to simplify this linear program further to obtain the form as officially presented in *Proposition 1.*

Simplification of Linear Program

Proposition 2. *For* $1 \leq j \leq i \leq n$*:*

$$\sum_{s=j}^{n}(n+1-s)\frac{\binom{i-1}{j-1}\binom{n-i}{s-j}}{\binom{n-1}{s-1}} = \frac{n(n+1)(i-j+1)}{i(i+1)}$$

Proof. First, observe that $\frac{\binom{i-1}{j-1}\binom{n-i}{s-j}}{\binom{n-1}{s-1}}$ is the probability of the ith applicant being jth best so far **given** she is sth best overall. Using Bayes' rule, we obtain:

$$\begin{aligned}
\frac{\binom{i-1}{j-1}\binom{n-i}{s-j}}{\binom{n-1}{s-1}} &= \Pr[i\text{th applicant is } j\text{th best so far} \mid i\text{th applicant is } s\text{th best overall}] \\
&= \Pr[i\text{th applicant is } s\text{th best overall} \mid i\text{th applicant is } j\text{th best so far}] \\
&\quad \cdot \frac{\Pr[i\text{th applicant is } j\text{th best so far}]}{\Pr[i\text{th applicant is } s\text{th best overall}]} \\
&= \frac{\binom{s-1}{j-1}\binom{n-s}{i-j}}{\binom{n}{i}} \cdot \frac{n}{i}
\end{aligned}$$

Hence, it follows that:

$$\begin{aligned}
\sum_{s=j}^{n}(n+1-s)\frac{\binom{i-1}{j-1}\binom{n-i}{s-j}}{\binom{n-1}{s-1}} &= \sum_{s=j}^{n}(n+1-s)\frac{n}{i}\frac{\binom{s-1}{j-1}\binom{n-s}{i-j}}{\binom{n}{i}} \\
&= \frac{n(n+1)}{i}\sum_{s=j}^{n}\frac{\binom{s-1}{j-1}\binom{n-s}{i-j}}{\binom{n}{i}} - \sum_{s=j}^{n}\frac{sn}{i}\frac{\binom{s-1}{j-1}\binom{n-s}{i-j}}{\binom{n}{i}} \\
&= \frac{n(n+1)}{i} - \sum_{s=j}^{n}\frac{jn}{i}\frac{\binom{s}{j}\binom{n-s}{i-j}}{\binom{n}{i}} \\
&= \frac{n(n+1)}{i} - \frac{jn}{i}\frac{\binom{n+1}{i+1}}{\binom{n}{i}} \\
&= \frac{n(n+1)(i-j+1)}{i(i+1)}
\end{aligned}$$

In the fourth equality above, we used the well-known combinatorial identity $\sum_{s=j}^{n}\binom{s}{j}\binom{n-s}{i-j} = \binom{n+1}{i+1}$.

□

We can now obtain the desired linear program by using the previous proposition and through exchanging a double sum. The result is what we have been after: (P) is a linear programming representation of the utility-based secretary program, and its decision variables can be interpreted probabilistically as defined earlier. Also observe our and [2]'s approaches both arrive at the desired linear program. A desirable trait in this second proof above is a clear method for constructing hiring mechanisms from feasible linear programming solutions, which is not readily evident from our first proof technique.

2.3 The Secretary Problem With Backward Solicitation

It should be observed that the method presented earlier can be employed to derive BJS-style linear programs for other secretary problems. The main task in the process is finding an appropriate transformation of variables, which varies from problem to problem. In those cases when employing a BJS approach proves difficult, the technique above can still be used to gain insight to the problem at hand. In this section, we will illustrate the technique for a version of the secretary problem which allows backward solicitation. This variant is central to our later works in modeling a game-theoretic approach to the secretary problem.

Consider the classical secretary objective of maximizing the probability for choosing the best overall applicant. We assume throughout this section that the employer has the option to recall any previous applicant, and each will accept the employer's offer with probability $\alpha(s,i)$. Here, s is the current round of interview, and i is the round when the employer first interviewed the relatively best applicant. As an example, suppose the employer is currently interviewing the 5th applicant, and the applicant in the 3rd round is the best so far, then $s = 5$, and $i = 3$. If the employer decides to offer the job to the 3rd applicant, he will accept with probability $\alpha(5,3)$. If the applicant does not accept, the employer must move on to interview the next one. Here, we do allow for multiple offers to the same applicant in different rounds (as oppose to [12]). That is, if an applicant rejects an offer this round, the employer may still offer that same applicant in the next round, albeit the probability of acceptance may be different (depending on the structure of $\alpha(\cdot,\cdot)$).

Dynamic Programming Formulation

Suppose we are at the position (s,i). That is, the employer is interviewing the sth applicant, and the relatively best applicant is at position i. As in [12], let $\pi_f(s,i)$ be the probability of the employer hiring the overall best applicant if she decides to interview the next applicant without solicitating the then current relatively best applicant. Also, let $\pi_b(s,i)$ be the probability of hiring the overall best applicant if she decides to hire the then current relatively best applicant. Let $\pi(s,i)$ be the probability of hiring the overall best applicant. Then we must have the following:

$$\begin{aligned}
\pi_f(s,i) &= \tfrac{1}{s+1}\pi(s+1,s+1) + \tfrac{s}{s+1}\pi(s+1,i) \\
\pi_b(s,i) &= \tfrac{s}{n}\cdot\alpha(s,i) + \pi_f(s,i)\cdot(1-\alpha(s,i)) \\
\pi(s,i) &= \max\{\pi_f(s,i), \pi_b(s,i)\} \\
\pi(n,i) &= \alpha(n,i)
\end{aligned}$$

Solving the above dynamic problem will yield the optimal policy that the employer should follow. From the dynamic program, we can readily form the following equivalent linear program.

$$
\begin{array}{llcl}
 & \min & \pi(1,1) & \\
(D'_{BS}) & \text{s.t.} & \pi(s,i) & \geq \frac{1}{s+1}\pi(s+1,s+1) + \frac{s}{s+1}\pi(s+1,i) \\
 & & \pi(s,i) & \geq \frac{s}{n}\alpha(s,i) + \left(\frac{1}{s+1}\pi(s+1,s+1) + \frac{s}{s+1}\pi(s+1,i)\right)(1-\alpha(s,i))
\end{array}
$$

We will next proceed to illustrate the steps needed to convert this initial linear program into another in line with BJS's representation.

Linear Programming Formulation

A BJS-style proof can be used to obtain the linear programming formulation, but we will delay this until the next chapter. There, the linear program is used to compute payoffs for applicants in a model of strategic secretary game. We show here the approach derived directly from the dynamic program above.

Proposition 3. *The secretary problem with backward solicitation can be solved using the linear program below.*

$$
\begin{array}{lll}
 & max & \frac{1}{n}\sum_{s=1}^{n} s \cdot p_s \\
(P_{BS}) & s.t & s \cdot p_{s,i} \leq \alpha(s,i) \cdot \left(1 - \sum_{l<i} p_l - \sum_{l\geq i}^{s-1} l \cdot p_{l,i}\right) \quad \forall\ 1 \leq i \leq s \leq n \\
 & & p_s = \sum_{i=1}^{s} p_{s,i} \quad \forall\ 1 \leq s \leq n \\
 & & p_{s,i} \geq 0,\ \ p_s \ \mathit{free}
\end{array}
$$

Proof. Consider the linear program (D'_{BS}) and use the transformation $s \cdot x_{s,i} = \pi(s,i) - \frac{1}{s+1}\pi(s+1,s+1) - \frac{s}{s+1}\pi(s+1,i)$ to obtain the relationship

$$\pi(s,i) = s\sum_{k=s}^{n} x_{k,i} + \sum_{k=s+1}^{n}\sum_{j=k}^{n} x_{j,k}$$

The linear program above can then be transformed into

$$
\begin{array}{lllll}
 & \text{minimize} & \sum_{k=1}^{n}\sum_{j=k}^{n} x_{j,k} & & \\
(D_{BS}) & \text{subject to} & x_{s,i} & \geq 0 & \forall\ 1 \leq i \leq s \leq n \\
 & & \frac{1}{\alpha(s,i)} s \cdot x_{s,i} + s\sum_{k>s}^{n} x_{k,i} + \sum_{k>s}^{n}\sum_{j=k}^{n} x_{j,k} & \geq \frac{s}{n} & \forall\ 1 \leq i \leq s \leq n
\end{array}
$$

Letting $p_{s,i}$ to be dual variables of the second set of constraints (which corresponds to the **stopping** action on state (s,i)), and form the dual linear program to (D_{BS}). A straightforward simplification yields (P_{BS}).

□

General Strategy For Formulations

In general, if we have two possible actions to take, **continue** or **stop**, we should first derive a linear program directly from the set of dynamic programming constraints, then form a BJS-style linear program through the use of a variable transformation for the **continue** constraints. This would make the right hand side of these constraints to be 0, and thus guarantee non-negativity for the transformed variables. As such, when we form the dual linear program, only variables corresponding to the **stop** action would remain in place. These are the p's and q's in BJS's and our works. Having derived necessary tools for our later analysis, we next consider incentive compatibility in the secretary problem.

Chapter 3

Participant Incentives and Consequences

3.1 Motivation

An inherent problem with the secretary problem's optimal threshold policy is that it completely ignores applicants' motives. Imagine yourself spending an entire day interviewing, only to find out later that you are not selected for the position. Worse still, this has nothing to do with your qualifications, but rather is due to the employer's hiring scheme. If you are one of those applicants in earlier slots, you are a part of the *learning* phase and are being used as guinea pigs in this selection process. Would you participate in an interview process knowing that you will not have any chance of getting selected? It is reasonable to assume that this is not something anyone would want, and as such, these earlier applicants have strong incentive to not show up to their scheduled interview slots in the first place. Thus, this is disaster from the employer's point of view, as she now may not be able to observe and learn from early applicants as the optimal policy was designed to do.

Similar arguments could also be used for those e-commerce applications that we introduced at the beginning of this dissertation. Assuming the number of potential bidders are known, someone who has recently arrived at an online auction listing may not put down an offer or bid, as the seller will only use those early offers to gauge later potential bids. The setting is the same as before, where the seller's potential for choosing the best offer hinges on having these early applicants showing up in their intended time slots. Although we analyze the problem in a hiring framework throughout this chapter, it should be noted that similar logic can be applied to other settings.

When slot i has a higher probability of getting selected than slot j, an applicant is said to prefer slot i over slot j. We follow Buchbinder, Jain, and Singh [2] (henceforth will be referred to as BJS) and say a hiring mechanism is *incentive compatible* (IC) when each applicant does not prefer other interviewing slots over his own. Thus two important questions arise: first, does there exist an IC hiring mechanism? And second, if existence is guaranteed, what is the optimal IC hiring mechanism? Obviously, selecting applicants randomly with equal probability constitutes an IC hiring mechanism, and hence the first question has an affirmative answer. The key question then, is how much better than random selection can an employer do?

BJS considered incentives in this setting of the secretary problem. They gave an answer to the second question using a linear programming approach. There, they find that an optimal IC hiring mechanism selects the best applicant with probability $1 - \frac{1}{\sqrt{2}} \approx 0.29$ as $n \to \infty$ (where the symbol $\approx$ denotes an approximation of the true value), as compared to the more well known number $\frac{1}{e} \approx 0.368$ (also in the limit $n \to \infty$). Moreover, observe that random selection hires the best applicant with probability $\frac{1}{n} \to 0$. As such, their IC hiring mechanism does relatively well against the traditional optimal threshold hiring mechanism, and is a significant improvement over the trivial random selection mechanism. Let p_i^π denoted the unconditional probability a hiring mechanism π selects the applicant in the ith slot. BJS's approach consists of formulating the secretary problem as a linear program, and adding constraints $p_i^\pi = p_j^\pi$ for all $i \neq j$, which stipulates the mechanism must select all slots with the same probability.

In this chapter, we will show how to obtain linear programs derived in [2] as duals of some appropriately transformed linear program for a Markov Decision Process. Furthermore, we will explore other versions of the secretary problem in the same manner as [2], and show many conclusions which are vastly different from what BJS obtained in the traditional setting. One such result is that IC hiring mechanisms can have optimal value an order of magnitude different from the non-IC case. For example, it will be shown that by introducing IC to the rank-based problem, the employer will increase her expected rank from 3.8695, a constant, to one in the order of $\Theta(\log(n))$, which goes to infinity as $n \to \infty$.

3.2 An Incentive Compatible Hiring Mechanism Can Be Costly!

We say an incentive compatible hiring mechanism (for a minimization problem) is *costly* if its resulting optimal value $z_{IC}^*(n)$ has the property that $\lim_{n\to\infty} \frac{z^*(n)}{z_{IC}^*(n)} = 0$, where $z^*(n)$ is the optimal value for the traditional, non-incentive compatible case. Equivalently, for a maximization problem, we say the incentive compatible hiring mechanism is *costly* if $\lim_{n\to\infty} \frac{z_{IC}^*(n)}{z^*(n)} = 0$. Observe that the classical secretary problem has been shown to be not *costly* in [2]. This section shows that the rank-based secretary problem behaves differently, and that it is *costly* when one attempts to make it incentive compatible.

Per our discussion from the last chapter, the IC rank-based secretary problem can be posed as the following linear program:

$$
(P_{IC}) \quad
\begin{array}{lll}
\max & \sum_{i=1}^{n}\sum_{j=1}^{i} \frac{(n+1)(i-j+1)}{i(i+1)} q_i^j & \\
\text{s.t.} & q_i^j \leq 1 - \sum_{l<i} p_l & 1 \leq j \leq i \leq n \\
& p_i = \frac{1}{i} \sum_{k=1}^{i} q_i^k & 1 \leq i \leq n \\
& p_i = p & \\
& p, q_i^j \geq 0 &
\end{array}
$$

where p_i can be interpreted as the probability a policy selects the ith applicant, and q_i^j the probability a policy selects the ith applicant given he is the jth best so far. The condition $p_i = p$ makes certain that all slots are given the same probability of getting chosen. This is the incentive compatible approach proposed in [2].

Since each feasible solution to the linear program corresponds to a feasible policy, the problem at hand now reduces to optimizing the incentive compatible utility-based LP. The next *Proposition* shows how to construct the optimal solution for the IC utility-based problem, and exhibits its objective value.

Proposition 4. *Define $P_{IC}(p)$ as the linear program (P_{IC}) with a fixed $p \in [0, \frac{1}{n}]$. The following feasible solution is optimal for $(P_{IC}(p))$:*

- $1 \leq i \leq \lfloor \frac{1}{2p} + \frac{1}{2} \rfloor$*: then* $q_i^1 = ip$, $q_i^j = 0$ *for* $j \neq 1$.
- $\lfloor \frac{1}{2p} + \frac{1}{2} \rfloor + 1 \leq i \leq n$*: then* $q_i^1 = \ldots = q_i^k = 1 - (i-1)p$, $q_i^{k+1} = ip - k\left(1 - (i-1)p\right)$, *and* $q_i^j = 0$ *for* $k+1 < j \leq n$. *Here,* $k = \lfloor \frac{ip}{1-(i-1)p} \rfloor$.

And the corresponding objective value for this optimal incentive compatible mechanism is

$$
\begin{aligned}
u_n^*(p) \quad = \quad & \sum_{i=1}^{\lfloor \frac{1}{2p}+\frac{1}{2} \rfloor} \frac{(n+1)i}{i(i+1)} ip + \sum_{i=\lfloor \frac{1}{2p}+\frac{1}{2} \rfloor+1}^{n} \left(\sum_{j=1}^{\lfloor \frac{ip}{1-(i-1)p} \rfloor} \frac{n+1}{i(i+1)} (i-j+1)\left(1-(i-1)p\right) \right) \\
& + \sum_{i=\lfloor \frac{1}{2p}+\frac{1}{2} \rfloor+1}^{n} \frac{(n+1)}{i(i+1)} \left(i - \left\lfloor \frac{ip}{1-(i-1)p} \right\rfloor \right) \left(ip - k(1-(i-1)p)\right)
\end{aligned}
$$

Proof. First, for a given i, observe that $\frac{(n+1)(i-j+1)}{i(i+1)}$ is decreasing in j. As $\sum_{j=1}^{i} q_i^j = ip$, we should shift as much as possible into smallest j's. Since $q_i^j \leq 1 - (i-1) \cdot p$, the right hand side (RHS) serves as an upper ceiling for each q_i^j.

We consider two scenarios:

- $ip \leq 1 - (i-1)p$: then because of the observation above, we should shift everything into q_i^1, so that $q_i^1 = ip$. Note that $ip \leq 1 - (i-1)p \iff i \leq \frac{1}{2p} + \frac{1}{2}$.

- $ip > 1-(i-1)p$: this means we can only shift a maximum of $1-(i-1)p$ into each q_i^j. The maximum number of j's that we can shift $(1-(i-1)p)$ into is $k = \lfloor \frac{ip}{1-(i-1)p} \rfloor$. Whatever that is left over, i.e. $ip - k(1-(i-1)p)$, should be shifted to q_i^{k+1}.

The optimal objective value is derived as a consequence of the above constructed solution. □

We now obtain an upper bound to the optimal value of the utility-based problem, and equivalently, a lower bound to that of the rank-based problem. These results are presented in the next two propositions.

Proposition 5. *Let $u_n^*(p)$ denote the optimal objective value to the problem $P_{IC}(p)$. Then $u_n^*(p) \leq n+1-\frac{n+1}{n}\sum_{i=1}^{n}\frac{1}{i+1}$ for all $p \in [0, \frac{1}{n}]$.*

Proof. Consider $p \in [0, \frac{1}{n}]$, then:

$$\begin{aligned}
u_n^*(p) &\leq \sum_{i=1}^{n} \frac{(n+1)(i-1+1)}{i(i+1)} i \cdot p \\
&= \sum_{i=1}^{n} \frac{(n+1)i}{i+1} p \\
&= (n+1)p \sum_{i=1}^{n} \left(1 - \frac{1}{i+1}\right) \\
&= n(n+1)p - (n+1)p \sum_{i=1}^{n} \frac{1}{i+1} \\
&\leq n+1 - \frac{n+1}{n} \sum_{i=1}^{n} \frac{1}{i+1}
\end{aligned}$$

Justification for the two inequalities are as follows:

- Note that for $1 \leq i \leq \lfloor \frac{1}{2p} + \frac{1}{2} \rfloor$, we have $q_i^1 = ip$ and $q_i^j = 0$ for $j \neq 1$, per *Proposition 4*. When i is outside this range, we can shift all of the weight ip to q_i^1, and let $q_i^j = 0$ for other j's. Clearly this new solution is infeasible (it violates the condition $q_i^j \leq 1-(i-1)p$), but it forms an upper bound to the objective function value. Hence the first inequality follows.

- Observe that $n(n+1)p - (n+1)p\sum_{i=1}^{n}\frac{1}{i+1}$ is increasing in p. Since $p \in [0, \frac{1}{n}]$, the above is maximized at $p = \frac{1}{n}$. Hence the second inequality follows.

□

With the above proposition, we are now in position to show the optimal expected rank of a hired applicant is $\Omega(\log(n))$.

Proposition 6. *The optimal expected rank in the envy-free rank-based secretary problem is* $\Omega(\log(n))$.

Proof. Per our observation earlier, $r_n^* = n + 1 - u_n^*$. By *Proposition 5*, we then have:

$$r_n^* \geq n+1-\left(n+1-\frac{n+1}{n}\sum_{i=1}^{n}\frac{1}{i+1}\right) = \frac{n+1}{n}\sum_{i=1}^{n}\frac{1}{i+1} = \Omega(\log(n))$$

□

Recall that in the classical setting with objective of maximizing the probability of selecting the best applicant, introducing incentive compatibility decreases the optimal value from $\frac{1}{e} \approx 0.368$ to $(1 - 1/\sqrt{2}) \approx 0.293$. When we change the objective to minimizing the expected rank of the hiree, introducing incentive compatibility increases the optimal value from ≈ 3.870 to $\Omega(\log(n))$. As such, incentive compatibility can be costly to the employer depending on her hiring objective.

We next try to form a $\log(n)$ upper bound for the rank-based incentive compatible secretary problem. Similar to the previous case, we start out with a proposition for the utility-based problem. Its equivalence in the rank-based setting is also exhibited.

Proposition 7. *Let* u_n^* *be the optimal value for the linear program* (P_{IC})*. Then we must have* $u_n^* \geq n+1-\frac{n+1}{n}\log\frac{n}{2}-\frac{n+1}{n}\cdot\frac{n+1}{2(n+2)}\log n = \Omega(n-\log(n))$*. Equivalently,* $r_n^* \leq \frac{n+1}{n}\log\frac{n}{2}+\frac{n+1}{n}\cdot\frac{n+1}{2(n+2)}\log n = \mathcal{O}(\log(n))$*.*

Proof. Consider the optimal solution in *Proposition 4*, and modify q_i^{k+1} to be equal to 0, keeping all other values to be the same. Furthermore, let $p = \frac{1}{n}$. It follows that u_n^* is at least as large as the objective value evaluated at this (very likely to be infeasible) solution:

$$\begin{aligned}
u_n^* &\geq \sum_{i=1}^{\lfloor\frac{n}{2}+\frac{1}{2}\rfloor} \tfrac{(n+1)i}{i(i+1)}\cdot\tfrac{i}{n} + \sum_{i=\lfloor\frac{n}{2}+\frac{1}{2}\rfloor+1}^{n}\ \sum_{j=1}^{\lfloor\frac{i}{n-i+1}\rfloor} \tfrac{(n+1)(i-j+1)}{i(i+1)}\left(1-\tfrac{i-1}{n}\right)\\
&= \tfrac{n+1}{n}\sum_{i=1}^{\lfloor\frac{n}{2}+\frac{1}{2}\rfloor}\left(1-\tfrac{1}{i+1}\right) + \sum_{i=\lfloor\frac{n}{2}+\frac{1}{2}\rfloor+1}^{n}\ \sum_{j=1}^{\lfloor\frac{i}{n-i+1}\rfloor}\left(\tfrac{n+1}{i}\tfrac{n-i+1}{n} - \tfrac{n+1}{n}\tfrac{n-i+1}{i(i+1)}\cdot j\right)\\
&\geq \tfrac{n+1}{n}\sum_{i=1}^{\lfloor\frac{n}{2}+\frac{1}{2}\rfloor}\left(1-\tfrac{1}{i+1}\right) + \sum_{i=\lfloor\frac{n}{2}+\frac{1}{2}\rfloor+1}^{n}\left(\sum_{j=1}^{\lfloor\frac{i}{n-i+1}\rfloor}\tfrac{n+1}{n}\tfrac{n-i+1}{i} - \tfrac{n+1}{n}\sum_{j=1}^{\frac{i}{n-i+1}}\tfrac{n-i+1}{i(i+1)}\cdot j\right)\\
&\geq \tfrac{n+1}{n}\sum_{i=1}^{\lfloor\frac{n}{2}+\frac{1}{2}\rfloor}\left(1-\tfrac{1}{i+1}\right) + \sum_{i=\lfloor\frac{n}{2}+\frac{1}{2}\rfloor+1}^{n}\tfrac{n+1}{n}\left(\left(\tfrac{i}{n-i+1}-1\right)\cdot\tfrac{n-i+1}{i} - \tfrac{n+1}{2(i+1)(n-i+1)}\right)\\
&\geq n+1-\tfrac{n+1}{n}\log\tfrac{n}{2} - \tfrac{n+1}{n}\cdot\tfrac{n+1}{2(n+2)}\log n
\end{aligned}$$

Here, the second inequality holds due to subtraction of a larger term. The third inequality results from the fact that $\lfloor \frac{i}{n-i+1} \rfloor \geq \frac{i}{n-i+1} - 1$, and the last line ignores a negative term in the sum involving $\frac{n+1}{n}\frac{n-i+1}{i}$. Nevertheless, this shows that u_n^* is in $\Omega(n - \log(n))$.

The statement regarding r_n^* can be obtained by observing the relationship between it and u_n^*.

□

With these results in hand, we can now conclude the optimal expected rank grows in the order of $\log n$.

Theorem 2. *The optimal expected rank $r_n^* = \Theta(\log n)$.*

Proof. By the above proposition, we know $r_n^* = \mathcal{O}(\log n)$. Due to *Proposition 6*, we also know that $r_n^* = \Omega(\log n)$. As such, we conclude that $r_n^* = \Theta(\log n)$.

□

From this result, we observe that incentive compatibility as BJS [2] defined it can be a costly attribute for rank-based hiring mechanisms.

3.3 Always-Hire Is Near-Optimal In The Asymptotic

In the classical setting of the secretary problem, the optimal policy may not select anyone at all. Furthermore, it is a nonobvious fact that the best IC policy may also not pick anyone. Moreover, even if it does select someone, that person may not be the best so far at the time of selection (see [2]). Contrast these findings to those for the rank-based (equivalently utility-based) problem. In the traditional and IC rank-based settings, the employer is guaranteed to always select someone, and selecting someone is near-optimal, respectively. This first claim of policies always making a hire was given as an assumption in [3], and will be shown by us as a simple consequence in our model. With regard to the second issue raised, it is not relevant in the rank-based setting, as our objective is no longer hiring the best overall, but to achieve the minimum expected rank. We state and prove these properties in the following lemmas.

Lemma 3. *In the traditional utility-based problem (non-incentive compatible), the optimal hiring policy always select someone.*

Proof. At first glance, this statement seems self-evident, as hiring someone always result in positive utility, and not hiring anyone would result in 0 utility. As such, it is to the employer's advantage to always select someone. We prove our claim below.

The proof is by contradiction. Suppose (p_i, q_i^j) is an optimal solution to the linear program presented in *Proposition 1* with $\sum_{i=1}^{n} p_i = 1 - \epsilon$ for some $\epsilon > 0$. In other words, the optimal policy corresponding to this optimal LP solution does not select anyone with positive probability. Observe that this solution must satisfy $q_n^j = 1 - \sum_{l<n} p_l$ for all $1 \le j \le n$, otherwise we can increase p_n, then increase the q_n^j which does not satisfy the equality relationship, which in turn increases the objective function.

With $q_n^j = 1 - \sum_{l=1}^{n-1} p_l$ for all $1 \le j \le n$, and $np_n = \sum_{l=1}^{n} q_n^l$, we then have the equality $np_n = \sum_{l=1}^{n} q_n^l = n - n \cdot \sum_{l=1}^{n-1} p_l$, so that $\sum_{i=1}^{n} p_i = 1$, contradicting our earlier assumption that this sum is strictly less than 1.

□

We have just shown the employer will always select someone in the utility-based (equivalently, rank-based) secretary problem. The next lemma examines the same setting, but with extra constraints for incentive compatibility.

Lemma 4. *For sufficiently large n, the best hiring policy among those that must select someone is near-optimal for the utility-based incentive compatible secretary problem.*

Proof. From *Propositions 5* and *7*, we obtain the following bounds for $u_n^*(p)$:

$$\begin{aligned}
n + 1 - 3\log(n) &\le n + 1 - \tfrac{2n}{n}\log\tfrac{n}{2} - \tfrac{2n(n+2)}{2n(n+2)}\log(n) \\
&\le n + 1 - \tfrac{n+1}{n}\log\tfrac{n}{2} - \tfrac{n+1}{n}\cdot\tfrac{n+1}{2(n+2)}\log n \\
&\le u_n^*(p) \\
&\le (n+1)n \cdot p - (n+1)p \cdot \sum_{i=1}^{n}\tfrac{1}{i+1} \\
&\le ((n+1)n - (n+1)\log(n+1))\, p
\end{aligned}$$

Observe that for $p \in \left(0, \frac{(n+1)-3\log(n)}{(n+1)n-(n+1)\log(n+1)}\right)$, these inequality bounds for $u_n^*(p)$ are being violated. As such, for a fixed n, the optimal p^* must lie in the interval $\left[\frac{(n+1)-3\log(n)}{(n+1)n-(n+1)\log(n+1)}, \frac{1}{n}\right]$. As n increases, this interval shrinks, and approaches $\frac{1}{n}$ in the limit from the left hand side. Moreover, the objective value obtained by letting $p = \frac{1}{n}$ differs from $u_n^* = \max_p u_n^*(p)$ by no more than $(n+1) - \log(n+1) - (n+1) + 3\log(n+1) = 2\log(n+1)$. This gives us a solution to within $\left(1 - \frac{2\log(n+1)}{n}\right)$ of the optimal solution simply by letting $p = \frac{1}{n}$.

□

We believe the statement in the previous *Lemma* can be made stronger, but are not able to provide a rigourous proof. The difficulty arises when we deal with several floor functions appearing in the optimal objective value. As such, we leave the statement here as a conjecture.

Conjecture: For all $n \ge 1$, the optimal incentive compatible mechanism for the utility-based secretary problem will always hire someone.

3.4 Incentive Compatibility In Generalized Utility-Based Problems

In a generalized utility-based secretary problem, the employer will derive a utility of $f(s)$ units for hiring the sth best overall applicant. Modeling this problem as a dynamic program or a linear program is similar to what we have shown previously. If we modify the classical secretary problem, and assign $f(1) = n$, and $f(s) = 0$ for all other $s \neq 1$, [2] showed that the optimal incentive compatible policy selects a candidate with expected utility of $\left(1 - \frac{1}{\sqrt{2}}\right) n$. In this section, we would like to know whether other variations of the utility-based secretary problem will improve upon this, perhaps up to a constant difference from n, the number of applicants. From the previous section, we know the incentive compatible rank-based secretary problem optimally selects an applicant with expected utility in $\Theta(n - \log(n))$. What if the employer has a utility function somewhere in between? In other words, what if $f(s) = n + 1 - s$ for $s = 1, \ldots, K$, and $f(s) = 0$ for $s \geq K + 1$, with K dependent or independent of n? This section focuses on the asymptotic behavior of such a group of utility function.

Asymptotic Behavior For A Special Class of Secretary Problems

We now focus on a special class of the utility-based secretary problem, namely that of $f(s) = n + 1 - s$ for $s = 1, \ldots, K$, and $f(s) = 0$ for $s = K + 1, \ldots, n$. We do not require the employer to hire an applicant by the end of the process. Here, K is assumed to be a fixed number, independent of n.

Proposition 8. *For fixed n, $1 \leq i \leq n$, and $1 \leq K \leq n$, then*

$$f_{n,i}^{K}(j) = \sum_{s=j}^{K} (n + 1 - s) \frac{\binom{i-1}{j-1}\binom{n-i}{s-j}}{\binom{n-1}{s-1}}$$

is a decreasing function in j.

Proof. We wish to show the difference $f_{n,i}^{K}(j) - f_{n,i}^{K}(j+1) > 0$. To do this, simply compare the term $(n + 1 - j')\frac{\binom{i-1}{j-1}\binom{n-i}{j'-j}}{\binom{n-1}{j'-1}}$ in $f_{n,i}^{K}(j)$, against the term $(n + 1 - j')\frac{\binom{i-1}{(j+1)-1}\binom{n-i}{j'-(j+1)}}{\binom{n-1}{j'-1}}$ in $f_{n,i}^{K}(j+1)$ for all $j' \geq j + 1$. The comparison simplifies to:

$$(n + 1 - j')\binom{i-1}{j-1}\binom{n-i}{j'-j} > (n + 1 - j')\binom{i-1}{(j+1)-1}\binom{n-i}{j'-(j+1)} \iff \frac{n-i-(j'-j)+1}{j'-j} > \frac{i-j}{j}$$

The left hand side of the inequality is a decreasing function in j' for $2 \leq j' \leq n$. As such, the partial sum (in terms of K) of $f_{n,i}^{K}(j) - f_{n,i}^{K}(j+1)$ is unimodal as K increases (i.e. this difference may first increase with K up to some appropriate K^*, then starts decreasing as K

goes beyond this K^*). When $K = j$, clearly this difference is positive. Furthermore, when $K = n$, this difference also stays positive due to the closed-form formula of our utility-based objective function. As such, we can conclude the difference must remain positive for all K in between, i.e. $f_{n,i}^K(j)$ is decreasing in j.

□

Corollary 1. *Assign utility $(n+1-i)$ to the ith best overall applicant for $1 \le i \le K$, and utility 0 for all others. For any fixed K, the utility secretary problem has optimal value at most a constant factor of n (depending on K), where n is the number of applicants in the problem.*

Proof. First, observe that the corresponding linear program for this version of the incentive compatible secretary problem is:

$$\begin{array}{lll}
\max & \frac{1}{n}\sum\limits_{i=1}^{n}\sum\limits_{j=1}^{i}\sum\limits_{s=j}^{K}(n+1-s)\frac{\binom{i-1}{j-1}\binom{n-i}{s-j}}{\binom{n-1}{s-1}}q_i^j & \\
\text{s.t.} & q_i^j \le 1 - \sum\limits_{l<i} p_l & 1 \le j \le i \le n \\
 & p_i = \frac{1}{i}\sum\limits_{j=1}^{i} q_i^j & 1 \le i \le n \\
 & p_i = p_j = p & 1 \le i \ne j \le n \\
 & p, p_i, q_i^j \ge 0 & 1 \le j \le i \le n
\end{array}$$

Because for a fixed i we have $\sum\limits_{s=j}^{K}(n+1-s)\frac{\binom{i-1}{j-1}\binom{n-i}{s-j}}{\binom{n-1}{s-1}}$ is decreasing in j (per the preceding proposition), it follows that we should shift as much into q_i^j as possible, before moving on to q_i^{j+1}. The proof then becomes similar to that of *Propositions 4* and *5*: that for $1 \le i \le n$, we can shift all of ip into q_i^1, i.e. $q_i^1 = ip$ and $q_i^j = 0$ for $j \ne 1$ to form an *infeasible* solution. The optimal objective value of this incentive compatible secretary problem then has the following upper bound:

$$\frac{1}{n}\sum_{i=1}^{n}\sum_{s=1}^{K}(n+1-s)\frac{\binom{i-1}{1-1}\binom{n-i}{s-1}}{\binom{n-1}{s-1}}\cdot ip$$

Next, when n is large, and k is much smaller than n, we have the following approximation asymptotics: $\binom{n}{k} \approx \frac{\left(\frac{2n}{k}-1\right)^k}{\sqrt{2\pi k}}$. Apply this to the upper bound of the optimal objective value above and we obtain:

$$\begin{aligned}
\frac{1}{n}\sum_{i=1}^{n}\sum_{s=1}^{K}(n+1-s)\frac{\binom{n-i}{s-1}}{\binom{n-1}{s-1}}ip &\approx \frac{1}{n}\sum_{i=1}^{n}\sum_{s=1}^{K}(n+1-s)\left(\frac{2(n-i)-(s-1)}{2(n-1)-(s-1)}\right)^{s-1}\cdot ip\\
&\leq \frac{1}{n}\sum_{i=1}^{n}\sum_{s=1}^{K}(n+1-s)\frac{[2(n-1)-(s-1)]^{s+1}}{[2(n-1)-(s-1)]^{s-1}}\cdot\frac{1}{4s(s+1)}\cdot ip+\text{l.o.t.}\\
&= \frac{1}{n}\sum_{s=1}^{K}(n+1-s)\left(2(n-1)-(s-1)\right)^2\cdot\frac{1}{4s(s+1)}\cdot p+\text{l.o.t.}\\
&\leq \frac{1}{4n}\sum_{s=1}^{K}(n+1-s)\left(2(n-1)\right)^2\cdot\frac{1}{s(s+1)}\cdot p+\text{l.o.t.}\\
&= \left((n-1)^2\cdot\left(1-\frac{1}{K+1}\right)\right)\cdot p+\text{l.o.t.}
\end{aligned}$$

where the first inequality is obtained by noting that $\sum_{i=1}^{n} i(2(n-i))^{s-1}$ can be approximated by the integral $\int_1^n x(2(n-x))^{s-1}\,dx$, which results in $\frac{1}{4s(s+1)}[2(n-1)]^{s-1}$ and lower order terms (l.o.t.). The above upperbound is maximized at $p=\frac{1}{n}$, so that we cannot do better than a constant factor of n.

$\square$

We see that to get to the $n-\log(n)$ threshold (achieved in the incentive compatible utility-based secretary problem) requires a K larger than a constant. It is an open problem to improve upon this basic bound, i.e. for what other classes of K will this remain true? And for what other classes of K will we get to the $(n-\log(n))$ threshold?

3.5 Online Auction Incentive Compatibility

In certain settings, we need not be overly restrictive with incentive compatibility constraints $p_i = p_j$ for all $i, j \in \{1, \ldots, n\}$. Consider an online auction where potential bidders arrive in a random sequential order (think eBay), and we wish to give incentive to the bidder arriving at time slot i to bid right away, rather than delaying until a later time slot. It has been shown in [1] that for the classical setting, the optimal probability of selecting the best applicant is $\frac{1}{2\sqrt{e}} \approx 0.303265$. For the rank-based setting, we shall show that incentive compatibility in this online auction setting does not yield any asymptotic improvement over the traditional one.

Lemma 5. *Consider the rank-based secretary problem with incentive compatibility constraints $p_i \geq p_{i+1}$, for all i. The optimal expected rank for the rank-based version is also of order $\Theta(\log(n))$.*

Proof. Let $f(i,j) = \frac{(n+1)(i-j+1)}{i(i+1)}$, and observe that for a fixed i, $f(i,j)$ is decreasing in j. From the constraint $i \cdot p_i = \sum_{k=1}^{i} q_i^k$, and $f(i,j)$ monotonically decreasing in j, it follows that we want to shift as much weight into q_i^1 as possible.

Now consider a solution where all possible weights are shifted into q_i^1, leaving $q_i^2 = q_i^3 = \ldots = 0$, so that it has value $q_i^1 = i \cdot p_i$. The objective value for this particular solution is then $\sum_{i=1}^{n} \frac{(n+1)i}{i(i+1)} \cdot i \cdot p_i = \sum_{i=1}^{n} \frac{(n+1)i}{i+1} \cdot p_i$. Observe that $\frac{(n+1)i}{i+1}$ is increasing in i, so that we want to allocate more weight to p_j than to p_i whenever $j > i$. This, together with our incentive compatibile constraints $p_i \geq p_j$ whenever $j > i$ implies we must assign equal weights to all p_i's. Since $\sum_{i=1}^{n} p_i \leq 1$, we can only assign a maximum of $\frac{1}{n}$ to each p_i. The objective value for the **infeasible** solution $p_i = \frac{1}{n}$ and $q_i^1 = ip_i, q_i^2 = q_i^3 = \ldots = 0 \; \forall i$ is $n + 1 - \frac{n+1}{n} \sum_{i=1}^{n} \frac{1}{i+1}$. Since this is an upper bound to the expected utility problem, it follows that the optimal expected rank is of order $\Omega(\log(n))$.

Since $p_i \geq p_{i+1}$ for all i is a relaxation of the more restrictive $p_i = p_j$ for all i, j, and the optimal expected rank for the latter problem is in $\Theta(\log(n))$, our claim follows by having proved the optimal expected rank (for the version $p_i \geq p_{i+1}$) also belongs to $\Omega(\log(n))$.

$\square$

3.6 Implementation Of Incentive Compatible Hiring Mechanisms

One may cast doubt on our original motivation for coming up with new hiring policies. It is true that *if* applicants know they will get interviewed but never offered a job, then they have incentive to not show up. But does the *if* ever happen? From an applicant's point of view, how is he to decide whether the employer chooses to use the classical threshold policy or some other hiring methods? If one comes in the 3rd interview slot out of 10 and does not get selected, is it because the employer uses a threshold hiring policy, or is it due to the fact that one is simply not the best candidate so far? There is an implicit trust in the employer, and her objective of selecting the best applicant does not align with actions to honor it.

To overcome aforementioned drawbacks, we propose executing the hiring policy through a 3rd party. The employer can submit her hiring algorithm, and applicants can submit their inquiries regarding the hiring policy and get them answered through this central location. For example, an applicant can ask of the probability this policy will select someone in the 3 position, or the value of n in this process, and get answers for these. We believe this check-and-balance approach will deliver desired incentives to applicants, and at the same time, ensure the hiring algorithm's integrity from being interfered with by the employer.

We recognize the fact that collusions between this 3rd party and the employer is also a potential problem. For this, we refer our readers to literatures on the design of collusion-free protocols.

3.7 A Concluding Note On Incentive Compatibility

In the last and current chapters, we showed the incentive compatibility property can be costly to the decision maker in an optimal stopping setting, with this greatly depending on her objective function. This observation holds for both types of incentive compatibility proposed in [2] and [1]. From a high-level perspective, we also showed Buchbinder, Jain, and Singh's linear programming based approach for modeling secretary problems can be obtained from known methods in the theory of Markov Decision Processes. Our insight allows for an alternative approach in cases where BJS's technique cannot be easily applied.

Chapter 4

How To Win Jobs And Influence Employers

4.1 Introduction

In the classic secretary problem, n (commonly known) applicants present themselves one by one to an employer. If the employer could wait before making a decision, she would be able to rank them from best to worst, and select the best candidate. The order for which applicants present themselves to the employer is random, i.e. it is equally likely from among the $n!$ permutations. After an applicant is interviewed, the employer must make her hiring decision on the spot: either to accept this applicant, or to reject him and moves on to the next applicant without the possibility of going back. The employer's objective is to design an algorithm that would maximize her probability of selecting the overall best applicant.

As shown in the first chapter, the employer's optimal hiring policy is deceptively simple: to skip over all applicants from slots 1 to $r^* - 1$, and selects the applicant in slot $r^* \leq j \leq n$ if he happens to be the best among those observed so far. Here, r^* is the smallest integer between 1 and n which satisfies $\frac{r^*}{n} \geq \frac{r^*}{n} \sum\limits_{k=r^*+1}^{n} \frac{1}{k-1}$. This threshold policy is typical in optimal stopping problems, where the decision maker spends some time *learning* about the population, and starts working on the choosing process after a certain time point. A moment of reflection, however, indicates that the approach may not work with all subjects. When we are dealing with items, this policy is perfectly fine, as items do not think strategically. When we are dealing with living, rational human beings, this policy begs for much improvement. One approach that tries to mitigate this concern was proposed by Buchbinder et al. [2], and was extended and generalized to the rank-based case in the previous chapter. There, the employer constrained herself to only hiring policies for which applicants have incentives to remain in their interview slots. Here, we shall allow applicants more freedom in the hiring process, where they can make decisions which will affect everyone involved, from other applicants to the employer herself.

We find it useful to pause here and make clear an assumption of applicants' attitude toward different hiring mechanisms. Although it seems trivial, the statement helps resolve a thorny subject later in the section.

Assumption: An applicant finds it unacceptable if he is invited to the ith interview slot, and this slot has probability 0 of getting hired.

We next focus on models where applicants can choose from a set of strategies which is dependent on one's position in the process. Working with the classical setting of maximizing the probability of hiring the best applicant, we now allow each applicant to submit a window of availability to the employer. She, in turn, must make her decision on applicants before their respective deadlines. The timing and nature of how applicants can submit their windows of availability play an important role in the analysis. The resulting mathematical problems become quite interesting, and could potentially give us insights into how players in such an optimal stopping process may, and should, behave.

4.2 An Extensive Secretary Game With Imperfect Information

During each of the n time units, the employer would invite an applicant to come in for an interview. Her objective is to maximize the probability of hiring the best applicant among these available n, while each of the applicants knows his respective interview slot and wants to maximize his probability of getting hired. After getting his interview, an applicant communicates to the employer a time window $t_i = j$, where his job status must be decided by time $t = i + t_i = i + j$. As examples, if the applicant is interviewed on the 4th day, and offers his time window $t_4 = 0$, then the employer must make her decision for this applicant right away. Whereas if he gives his time window as $t_4 = 3$, then the employer can make her decision on or before the 7th day. Note that the set of available strategies is different from applicant to applicant, and hence the game is not symmetric. Furthermore, a later applicant only has partial information about the game, i.e. the employer is still searching, and not actual time window values from earlier applicants. For example, the 4th applicant knows the employer is still looking to hire due to the fact that he is getting an interview now, but he does not know exact values for each of the earlier t_i's. In this way, the sequential game has imperfect information, and is classified as such.

- **Game:** Extensive Classical Secretary Game With Imperfect Information
- **Players:** $\mathcal{N} = \{0, 1, 2, \ldots, n\}$. Here, 0 denotes the employer, and $1 \leq i \leq n$ denotes the ith applicant, who occupies the ith interview slot.
- **Description:** There are n applicants, each wanting to maximize his chance of landing the job, and one employer who wishes to maximize her probability of selecting the best

applicant among these n. The value n is commonly known among all applicants and the employer. Time progresses from 1 to n. Before time 1, the employer assigns slots randomly to all participants. At any time i, the following events occur in this order.

1. If no one has been selected, the employer sends an invitation for, and interviews the ith applicant.
2. Applicant i tells the employer of his non-binding time window t_i.
3. Employer decides if she wants to accept a previously interviewed, and still available, applicant. If yes, the process ends. If not, the process continues to time $i+1$. The current ith applicant is taken to be available at this time i.

It may at first appear difficult to describe with precision certain components of this game. For example, what are *terminal histories* in such a game, and the utility derived by each applicant in one such terminal history? It turns out that an important class of players' strategies can be found even when the game tree is not explicitly enumerated. This set of dominant actions is presented in the theorem below.

Theorem 3. *Let r^* be the smallest integer between 1 and n such that $1 \geq \sum_{k=r^*+1}^{n} \frac{1}{k-1}$. When presenting their time windows, it is a dominant strategy for*

(a) those in slots $r^*+1 \leq i \leq n$ to force the employer to make her decision right away, i.e. $t_i = 0$ for $r^*+1 \leq i \leq n$.

(b) those in slots $1 \leq i \leq r^*$ to allow the employer until time r^* to make her decision, i.e. $t_i = r^* - i$ for $1 \leq i \leq r^*$.

Proof. The proof is by backward induction.

(a) Since the last applicant has no choice but to use $t_n = 0$, the statement trivially holds in this case. Assume the statement holds for applicants $i, i+1, \ldots, n$, where $r^*+2 \leq i \leq n$.

If the employer ever gets to the $(i-1)$st applicant, then this applicant has $n-i+2$ choices: either $t_{i-1} = 0, t_{i-1} = 1, \ldots, t_{i-1} = n-i+1$. We need only focus on the scenario that he is the best so far out of the first $n-1$ applicants, as he will not get selected otherwise. From the induction hypothesis, we know all later applicants will force the employer to make her decision immediately when she gets to them. As such, letting $t_{i-1} = 0$ will allow this $(i-1)$st applicant to get selected with probability 1, whereas letting $t_{i-1} > 0$ will get him selected with probability strictly smaller than 1 as future applicants may be better candidates than he is. Therefore, he will choose to play $t_{i-1} = 0$.

(b) Now consider the (r^*-i)th applicant (again assuming he is the best so far). When he comes in for the interview, he knows later applicants will all play $t_j = 0$ for $j \geq r^*$. As such, if he lets $t_{r^*-i} < i$, his chance of getting hired is 0, as the employer has a higher probability of selecting the best one by delaying her decision until time r^*. If he allows the time window to be $t_{r^*-i} = i$, then his chance of getting hired is $\frac{r^*-i}{r^*}$. If he lets $t_{r^*-i} = t > i$, his probability

of getting selected is at most $\frac{r^*-i}{r^*-i+t}$, which is strictly smaller than $\frac{r^*-i}{r^*}$. Therefore, he will play $t_{r^*-i} = i$. In this way, these applicants will wish to delay until time r^*.

□

Corollary 2. *Let r^* be the smallest integer between 1 and n such that $1 \geq \sum_{k=r^*+1}^{n} \frac{1}{k-1}$. Under the extensive classical secretary game with imperfect information setting, the employer will hire the best applicant with probability $\frac{r^*}{n}$. Furthermore, this probability is at least as large as the maximum probability of selecting the best applicant in the classical secretary problem.*

Proof. Because applicants $1 \leq i \leq r^*$ all allow the employer until time r^* to make her hiring decision, she will choose to do so at time r^*. This gives her a probability of $\frac{r^*}{n}$ for picking the best overall applicant. Furthermore, from the definition of r^*, we obtain the following:

$$
\begin{aligned}
& & \sum_{k=r^*+1}^{n} \frac{1}{k-1} &\leq 1 \\
&\iff & \sum_{k=r^*+1}^{n} \frac{1}{k-1} + \frac{1}{r^*-1} &\leq 1 + \frac{1}{r^*-1} \\
&\iff & \sum_{k=r^*}^{n} \frac{1}{k-1} &\leq \frac{r^*}{r^*-1} \\
&\iff & \frac{r^*-1}{n} \sum_{k=r^*}^{n} \frac{1}{k-1} &\leq \frac{r^*}{n}
\end{aligned}
$$

Observe that the left hand side denotes the probability of selecting the best applicant under the classical setting, whereas the right hand side is one for our discussed game-theoretic setting.

□

It is well-known that r^* is on the order of $\frac{n}{e}$ in the limit (e.g. [7]), and thus both probabilities approach $\frac{1}{e}$ as $n \to \infty$. This fact can be shown using integral approximation for the summation.

Discussion Of Secretary Game And Resulting Dominant Strategy

To facilitate our discussion, we shall call applicants in slots $i \leq r^* - 1$ *early*, and those in slots $i \geq r^*$ *late*, where r^* is as defined in the previous theorem. Recall that under the classical setting of the secretary problem, *early* applicants are interviewed but never get selected. Motivated by this flaw in the classical optimal policy, we sought to design hiring mechanisms where applicants have incentive to attend their interviews. One way to guarantee this is to devise mechanisms where every interview slot has the same probability of getting selected. Another is to consider mechanisms where everyone interviewed would get the same probability of getting selected. The secretary game introduced in this section falls into the second category.

If the employer ever gets to a late applicant, it is optimal for him to force the employer to make her decision right away. Furthermore, these late applicants play an important role in getting early ones to allow the employer until time r^* to make her decision. As such, in our model, the employer uses these late applicants as threats against early ones. It is the expectation of other players employing rational strategies in future periods that dictate strategies for these *early* applicants. Moreover, these late applicants have incentive to come to the interview if they are ever invited. Contrast this to the classical setting of the secretary problem, where early applicants are being used as guinea pigs to learn about the population, and have no incentive to participate in the interview even if they get invitations to come. It is in this way that our proposed game-theoretic model addresses the issue raised by BJS regarding incentives in the classical secretary setting.

Employer's Incentive

Implicit in our assumption is the fact that there are truly n applicants participating in the hiring process. Consider a scenario where it is the employer who announces n to everyone. It is to his interest to inflate the announced value n, as that would allow him to select the best applicant with probability 1. As an example, suppose the true value of n is 3, but the employer announces a value of $n = 5$ instead. In other words, the first three applicants exist, but the last two are imaginary. These auxiliary applicants were created by the employer with the sole purpose of enhancing her probability of hiring the best candidate. And it works beautifully if other applicants have no means to check on its validity. Everyone now plays under the scenario of $n = 5$, and the three real applicants all allow the employer until time 3 to make her hiring decision. At time $t = 3$, the employer can make an informed hire as all three applicants are available. Her probability of selecting the best is 1.

As such, the hiring mechanism should ensure the correct value of n is being distributed to all applicants. A possible implementation is to have a central station where an applicant can access the number n and his interview position. Further complications may arise due to possible collusion between this central station and the employer, but we again defer such implementation details to related literatures in these areas of collusion-free protocol design. For now, we are content with the assumption that all players will have a way to access accurate information regarding the hiring process.

Some Final Words On This Extensive Secretary Game

By allowing applicants more options in making their decisions, it is a pleasantly surprising fact that the employer can overcome shortcomings of the original optimal policy while improving her maximum probability of selecting the best candidate. Observe that the employer's objective and that of the best overall applicant are aligned, where both want to be found by the other. As such, when the *must-decide-now* restriction is lifted, applicants have more room to strategize, and the result ends up benefiting the employer.

4.3 A Strategic Secretary Game: Binding Contracts And Equilibria

Continuing with our discussion from the last section, we next focus on the setting where all applicants are required to make known of their time windows before the first interview begins. These commitments are binding once submited, so that they cannot be changed later on, and each applicant must adhere to what he proposes. From these time windows, the employer can now use a corresponding optimal policy to maximize her probability of selecting the best applicant. We provide further details for this simultaneous game below.

Computing The Payoff Matrix

When an applicant is assigned an interview slot by the employer, he must also make known his window of availability t_i. So that an applicant in slot i will be available for hire during times $i+0, i+1, \ldots, i+t_i$. Given the strategy profile $\vec{t} = (t_1, t_2, \ldots, t_n)$ played by the applicants, the employer can devise an optimal hiring mechanism $\pi(\vec{t})$, which when used, will give the probability $P_i^{\pi(\vec{t})}$ for selecting the applicant at position i.

Computing the optimal hiring mechanism given $\vec{t}$ can be done through a linear program for the secretary problem with backward solicitation, which was presented in the previous chapter:

$$
\begin{array}{lccl}
\max & \frac{1}{n}\sum\limits_{s=1}^{n} s \cdot p_s & & \\
\text{s.t.} & s \cdot p_{s,i} & \leq & \alpha(s,i)\left(1 - \sum\limits_{l<i} p_l - \sum\limits_{l \geq i}^{s-1} l \cdot p_{l,i}\right) \\
 & p_s & = & \sum\limits_{i=1}^{s} p_{s,i} \\
 & p_{s,i} \geq 0,\ p_s \text{ free} & &
\end{array}
$$

In the above, $\alpha(s,i)$ denotes the probability the ith applicant will accept an offer at time s (so that $\alpha(s,i)$ are data), and $p_{s,i}$ is the probability the employer chooses the ith applicant at time s. As such, given $\vec{t}$, we have the following: if $i + t_i \leq s-1$, then $\alpha(s,i) = 0$; otherwise, $\alpha(s,i) = 1$. An alternative way is by setting $\alpha(s,i) = 1$ for all $1 \leq i \leq s \leq n$. Given $\vec{t}$, if $i + t_i \leq s-1$, then set $p_{s,i} = 0$.

Given $\vec{t}$, the employer can compute her optimal policy using the above linear program. From the perspective of the ith applicant, he will get chosen with probability $P_i^{\pi(\vec{t})} = \sum\limits_{s=i}^{n} p_{s,i}^{\pi(\vec{t})}$.

In order for us to be able to interprete the p's as probabilities, we need to use a proof similar to that presented in [2]. Below, we will show an optimal solution to the LP above corresponds to an optimal policy for the secretary problem with backward solicitation, and vice versa.

Lemma 6. *Take any mechanism π for selecting applicants. Let $\alpha(s,i)$ be the probability for which the ith applicant will accept an offer at the sth round. Let $p_{s,i}$ denote the probability π selects the ith applicant in the sth round. Let p_s be the probability π selects someone in the sth round. Then the linear program below gives an upper bound to the probability that π selects the best applicant:*

$$
\begin{array}{lll}
max & \frac{1}{n}\sum\limits_{s=1}^{n} s \cdot p_s & \\
s.t & s \cdot p_{s,i} \leq \alpha(s,i) \cdot \left(1 - \sum\limits_{l<i} p_l - \sum\limits_{l \geq i}^{s-1} l \cdot p_{l,i}\right) & \forall\ 1 \leq i \leq s \leq n \\
& p_s = \sum\limits_{i=1}^{s} p_{s,i} & \forall\ 1 \leq s \leq n \\
& p_{s,i} \geq 0,\ \ p_s \ \textit{free} &
\end{array}
$$

Proof. Define the event $A_{s,i}$ to be such that π *offers* the ith applicant in the sth round. Define $B_{s,i}$ to be such that the ith applicant is available in the sth round. We denote $C_{s,i}$ as the intersection of events $A_{s,i}$ and $B_{s,i}$, and say this is the event where π *selects* the ith applicant in the sth round (i.e. π *offers* the ith applicant in the sth round *and* this ith applicant is available in the sth round). Let $D_{s,i}$ be the event the ith applicant is the relative best in round s (i.e. he is the best among the first s applicants).

Take $q_{s,i} = \Pr[A_{s,i}, B_{s,i} \mid D_{s,i}] = \Pr[C_{s,i} \mid D_{s,i}]$, i.e. the probability a policy π *selects* the ith applicant in sth round *given* that this ith applicant is the relative best out of first s.

Take $p_{s,i} = \Pr[A_{s,i}, B_{s,i}] = \Pr[C_{s,i}]$, i.e. the probability a policy *selects* the ith applicant in sth round. Observe that $s \cdot p_{s,i} = q_{s,i}$ by a simple conditioning (here we focus on policies that only select the best so far, due to our objective function).

Take $p_s = \sum\limits_{i=1}^{s} p_{s,i}$, i.e. the probability the policy π *selects* an applicant in the sth round.

Let us obtain an upperbound for $q_{s,i}$. We have:

$$
\begin{aligned}
q_{s,i} &= \Pr[\pi \text{ selects the } i\text{th applicant in } s\text{th round} \mid i\text{th applicant is best out of } s] \\
&= \Pr[\pi \text{ offers the } i\text{th applicant in } s\text{th round } \textit{and} \\
&\qquad\qquad i\text{th applicant is available at round } s \mid i\text{th applicant is best out of first } s] \\
&= \Pr[\pi \text{ offers the } i\text{th applicant in } s\text{th round} \mid i\text{th applicant is best out of first } s] \\
&\qquad\qquad \cdot \alpha(s,i) \\
&\leq \Pr[\pi \text{ did not select anyone in rounds } 1,2,\ldots,i-1 \ \textit{and} \\
&\qquad\quad \text{did not select } i\text{th applicant in rounds } i, i+1, \ldots, s-1 \\
&\qquad\qquad \mid i\text{th applicant is best out of first } s] \cdot \alpha(s,i) \\
&= \left(1 - \sum_{l=1}^{i-1} p_l - \sum_{l=i}^{s-1} l \cdot p_{l,i}\right) \cdot \alpha(s,i)
\end{aligned}
$$

Here, the third equality follows the fact that being available is independent of all other events. The inequality can be obtained by observing that in order to *offer* the ith applicant in round s, the policy must not have *selected* anyone in rounds $1, 2, \ldots, i-1$, and must not have *selected* the ith applicant in rounds $i, i+1, \ldots, s-1$ (due to the conditional that this ith applicant is best out of the first s).

The last equality requires a bit of justification. First, observe that the probability of a feasible policy selecting someone in the sth round and selecting someone in the tth round is 0 (we can only select one person in the entire process). As such we can decompose the right hand side into sums of probabilities of selecting someone at different rounds. Second, we must have $\Pr[\pi \text{ selects } i\text{th in round } j \mid i\text{th is best out of } j] = q_{j,i} = j \cdot p_{j,i}\ \forall j \geq i$.

Replacing $q_{s,i}$ with $s \cdot p_{s,i}$ and we obtain the desired inequality.

Consider the same policy π, we must have the following as the objective value:

$$
\begin{aligned}
\Pr[\pi \text{ selects best overall}] &= \sum_{i=1}^{n} \Pr[\pi \text{ selects } i\text{th applicant} \mid i\text{th applicant is best overall}] \\
&\qquad \cdot \Pr[i\text{th applicant is best overall}] \\
&= \frac{1}{n} \sum_{i=1}^{n} \sum_{s=i}^{n} \Pr[\pi \text{ selects } i\text{th applicant in round } s \\
&\qquad\qquad \mid i\text{th applicant is best overall}] \\
&= \frac{1}{n} \sum_{s=1}^{n} \sum_{i=1}^{s} \Pr[\pi \text{ selects } i\text{th applicant in round } s \\
&\qquad\qquad \mid \text{he is best among first } s] \\
&= \frac{1}{n} \sum_{s=1}^{n} \sum_{i=1}^{s} q_{s,i} \\
&= \frac{1}{n} \sum_{s=1}^{n} s \cdot p_s
\end{aligned}
$$

The third equality follows by observing that at round s, π cannot distinguish between whether this ith applicant is best overall or best among s. As such, these conditional probabilities are the same.

We have shown all selection policies must satisfy these constraints, and have probability of selecting the best applicant as given by the objective function. As such, the linear program's optimal value is an upperbound to the probability of selecting the best applicant in the secretary problem with backward solicitation. This also finishes the proof.

□

Lemma 7. *From any feasible solution to the linear program presented earlier, we can construct a hiring mechanism which will allow us to select the best applicant with probability matching that of the objective function.*

Proof. Suppose we have p_s and $p_{s,i}$ satisfying constraints of the linear program. Let us construct the policy of hiring applicants from these values of p_s and $p_{s,i}$. Define this mechanism π so that it *offers* the ith applicant in round s *given* this ith applicant is best among the first s, it did not select anyone in rounds $1, 2, \ldots, i-1$ and did not select ith applicant in rounds $i, i+1, \ldots, s-1$ with probability $\frac{s \cdot p_{s,i}}{\alpha(s,i) \cdot \left(1 - \sum_{l<i} p_l - \sum_{l \geq i}^{s-1} l \cdot p_{l,i}\right)}$ (and 0 if $\alpha(s,i) = 0$). In other words, conditional on the process still going at round s and the ith applicant is the best at this stage, we should extend offer to this ith applicant with the above defined probability.

With the above constructed policy π, we claim π *selects* the ith applicant in round s with probability $p_{s,i}$, and (which follows directly from $p_{s,i}$) *selects* someone in round s with probability p_s. We shall show this by induction on s (the rounds).

- $s = 1$: then π offers the 1st applicant in round 1 with probability $\frac{1 \cdot p_{1,1}}{\alpha(1,1)}$. As such, π selects the 1st applicant in round 1 with probability $\frac{1 \cdot p_{1,1}}{\alpha(1,1)} \cdot \alpha(1,1) = p_{1,1}$.
- $s \leq \hat{s}$: assume the constructed policy π selects the ith applicant in round $\hat{s}$ with probability $p_{\hat{s},i}$ for all $1 \leq i \leq \hat{s}$, and selects someone in the sth round with probability p_s.
- $s = \hat{s} + 1$: from the constructed policy π, observe that it *selects* the ith applicant in round $\hat{s} + 1$ given that it did not select anyone in rounds $1, 2, \ldots, i - 1$ and did not select ith applicant in rounds $i, i + 1, \ldots, \hat{s}$, and the ith applicant is best out of $\hat{s} + 1$, is $\frac{(\hat{s}+1) \cdot p_{\hat{s}+1,i}}{1 - \sum_{l<i} p_l - \sum_{l=i}^{\hat{s}} l \cdot p_{l,i}}$. Next, we can find the probability π selects the ith applicant in round $\hat{s}+1$ by conditioning on whether ith applicant is best so far at round $\hat{s}+1$, and whether the process can reach the stage $\hat{s}+1$. Multiplying out the respective probabilities gives us what we desire.

Now that we have shown p_s and $p_{s,i}$ are indeed probabilities of the constructed policy π selecting applicants, the probability π selecting the best overall applicant can be computed as in the proof of the previous *Lemma*. As such, we have shown (constructed) a policy from the LP solution which gives matching objective value.

□

These two lemmas allow us to use the presented linear program for solving the secretary problem with backward solicitation. And from the LP, we can compute payoffs for applicants playing the game below.

A Strategic Game Between Job Applicants

We present here a model for strategic interaction between applicants. Here, there are n players (applicants), and the game is modeled as simultaneous-move.

Model: Strategic Game For The Classical Secretary Problem
Players: $\mathcal{N} = \{1, 2, \ldots, n\}$.
Action Sets: $\mathcal{A}_i = \{0, 1, \ldots, n - i\}$ is taken to be the set of available actions for the ith applicant.

Utility Function: $u_i : \mathcal{A}_1 \times \ldots \times \mathcal{A}_n \to \mathbb{R}$ is the ith applicant's utility function. Per our discussion earlier, we take $u_i(\vec{t}) \equiv P_i^{\pi(\vec{t})}$, where the latter can be computed from a linear program.

Contrast this simultaneous-move game model to the extensive game presented in the last section. In the extensive game, one is able to obtain the set of dominant strategies that each participant will use, and such strong result may not carry to here. Recall that in the previous game, the dominant strategy for each player is obtained by starting from the last applicant and work backward. Due to the sequential nature of time window submission, we were able to obtain the best action for each applicant at his time of interview. When all participants are making simultaneous decisions, that approach no longer works, and one must settle on some other solution concept for the problem. One such potential solution concept is that of equilibrium, a state where no participants wish to individually deviate from. There are no guarantees that applicants will play strategies that would result in any these equilibria, they are helpful nevertheless in gaining a better understanding of what people *might* do.

With the game formally defined, we next focus on finding a particular type of equilibria, Nash equilibria, for these strategic interaction settings. Recall that a Nash equilibrium of a strategic game is a strategy profile $\vec{t} \in \mathcal{A}_1 \times \ldots \times \mathcal{A}_n$ such that for every applicant $i \in \mathcal{N}$ we have $u_i(\vec{t}_{-i}, t_i) \geq u_i(\vec{t}_{-i}, \tilde{t})$ for all $\tilde{t} \in A_i$ (e.g. [10] [11]). Nash's theorem showed the existence of a mixed strategy in all strategic games, and that result necessarily carries over to this version of the secretary game as well. What is more interesting, and less obvious, is whether there exists an equilibrium in pure strategy for these games. For our game, a pure strategy for applicant i is a window of availability that is submitted with probability 1. As an example, in a secretary game of 3 applicants, if the first applicant submits $t_1 = 2$ with probability 1, then this would be considered a pure strategy. Whereas if the first applicant submits $t_1 = 0$ with probability $\frac{1}{2}$, and $t_1 = 1$ with probability $\frac{1}{2}$, then this would be considered a mixed strategy. We will answer this question of existence of pure strategy equilibrium in the upcoming sections.

Pure-Strategy Equilibrium For The Case of 3 Applicants

Let us illustrate this for the case $n = 3$, i.e. there are 3 applicants for the position. Consider the following feasible strategies $\vec{t} = (t_1, t_2, t_3)$.

1. $\vec{t} = (0, 0, 0)$: This is the classical setting. Solving yields the employer's probability of selecting the best applicant is $\frac{1}{2}$, and the applicants' probabilities for getting selected are:

$p_{s,i}$	1	2	3
1	0	$\cdot$	$\cdot$
2	0	$\frac{1}{2}$	$\cdot$
3	0	0	$\frac{1}{6}$
$P_i^{\pi(\vec{t})}$	0	$\frac{1}{2}$	$\frac{1}{6}$

2. $\vec{t} = (1, 0, 0)$: Solving yields the employer's probability of selecting the best applicant is $\frac{2}{3}$, and the applicants' probabilities for getting selected are:

$p_{s,i}$	1	2	3
1	0	$\cdot$	$\cdot$
2	$\frac{1}{2}$	$\frac{1}{2}$	$\cdot$
3	0	0	0
$P_i^{\pi(\vec{t})}$	$\frac{1}{2}$	$\frac{1}{2}$	0

3. $\vec{t} = (2, 0, 0)$: Solving yields the employer's probability of selecting the best applicant is $\frac{5}{6}$, and the applicants' probabilities for getting selected are:

$p_{s,i}$	1	2	3
1	0	$\cdot$	$\cdot$
2	0	$\frac{1}{2}$	$\cdot$
3	$\frac{1}{3}$	0	$\frac{1}{6}$
$P_i^{\pi(\vec{t})}$	$\frac{1}{3}$	$\frac{1}{2}$	$\frac{1}{6}$

4. $\vec{t} = (0, 1, 0)$: Solving yields the employer's probability of selecting the best applicant is $\frac{2}{3}$, and the applicants' probabilities for getting selected are:

$p_{s,i}$	1	2	3
1	0	$\cdot$	$\cdot$
2	0	0	$\cdot$
3	0	$\frac{1}{3}$	$\frac{1}{3}$
$P_i^{\pi(\vec{t})}$	0	$\frac{1}{3}$	$\frac{1}{3}$

5. $\vec{t} = (1, 1, 0)$: Solving yields the employer's probability of selecting the best applicant is $\frac{5}{6}$, and the applicants' probabilities for getting selected are:

$p_{s,i}$	1	2	3
1	0	$\cdot$	$\cdot$
2	$\frac{1}{2}$	0	$\cdot$
3	0	$\frac{1}{3}$	$\frac{1}{6}$
$P_i^{\pi(\vec{t})}$	$\frac{1}{2}$	$\frac{1}{3}$	$\frac{1}{6}$

6. $\vec{t} = (2, 1, 0)$: Solving yields the employer's probability of selecting the best applicant is 1, and the applicants' probabilities for getting selected are:

$p_{s,i}$	1	2	3
1	0	$\cdot$	$\cdot$
2	0	0	$\cdot$
3	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$
$P_i^{\pi(\vec{t})}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$

From the above, it follows that $(1, 0, 0)$ is the only pure-strategy equilibrium, and the employer's probability for selecting the best applicant has improved to $\frac{2}{3}$.

Non-Existence of Pure-Strategy Equilibria

Having seen how there exists a pure-strategy equilibrium for the case $n = 3$ applicants, we naturally seek to know whether there exists a pure-strategy equilibrium for every secretary game with $n \geq 4$ applicants. The answer turns out to be negative, and is stated in the next theorem.

Theorem 4. *When there are $n = 4$ applicants, dependent on the employer's optimal hiring policy, the secretary hiring game may not have any pure-strategy equilibrium.*

Proof. The proof is by brute force inspection. The reader is invited to check all 24 cases of playable strategies for applicants. Observe that when there are $n = 4$ applicants, these are the possible corresponding probabilities for getting selected. In the case applicants play $(1, 0, 1, 0)$, the employer selects to optimally hire at time 4. Stepping through all strategies show that none of the below 24 constitute an equilibrium. Note that had the employer optimally selected at time 2 (for the same $(1, 0, 1, 0)$), then there would have been a pure Nash equilibrium, namely $(1, 0, 0, 0)$.

As an example, the first table conveys the following information:

- When applicants use the strategies $(t_1 = 0, t_2 = 0, t_3 = 0, t_4 = 0)$, i.e. the classical secretary problem considered in chapter 1, the employer would select the best applicant with probability $\frac{11}{24}$.
- The 2nd applicant would get selected with probability $\frac{1}{2}$.

- The 3rd applicant would get selected with probability $\frac{1}{6}$.
- The 4th applicant would get selected with probability $\frac{1}{12}$.

$\vec{t}=(0,0,0,0)$				$\frac{11}{24}$
$p_{s,i}$	1	2	3	4
1	0	$\cdot$	$\cdot$	$\cdot$
2	0	$\frac{1}{2}$	$\cdot$	$\cdot$
3	0	0	$\frac{1}{6}$	$\cdot$
4	0	0	0	$\frac{1}{12}$
$P_i^{\pi(\vec{t})}$	0	$\frac{1}{2}$	$\frac{1}{6}$	$\frac{1}{12}$

$\vec{t}=(0,0,1,0)$				$\frac{1}{2}$
$p_{s,i}$	1	2	3	4
1	0	$\cdot$	$\cdot$	$\cdot$
2	0	0	$\cdot$	$\cdot$
3	0	0	0	$\cdot$
4	0	0	$\frac{1}{4}$	$\frac{1}{4}$
$P_i^{\pi(\vec{t})}$	0	0	$\frac{1}{4}$	$\frac{1}{4}$

$\vec{t}=(0,1,0,0)$				$\frac{7}{12}$
$p_{s,i}$	1	2	3	4
1	0	$\cdot$	$\cdot$	$\cdot$
2	0	0	$\cdot$	$\cdot$
3	0	$\frac{1}{3}$	$\frac{1}{3}$	$\cdot$
4	0	0	0	$\frac{1}{12}$
$P_i^{\pi(\vec{t})}$	0	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{12}$

$\vec{t}=(0,1,1,0)$				$\frac{2}{3}$
$p_{s,i}$	1	2	3	4
1	0	$\cdot$	$\cdot$	$\cdot$
2	0	0	$\cdot$	$\cdot$
3	0	$\frac{1}{3}$	0	$\cdot$
4	0	0	$\frac{1}{4}$	$\frac{1}{6}$
$P_i^{\pi(\vec{t})}$	0	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{6}$

$\vec{t}=(0,2,0,0)$				$\frac{2}{3}$
$p_{s,i}$	1	2	3	4
1	0	$\cdot$	$\cdot$	$\cdot$
2	0	0	$\cdot$	$\cdot$
3	0	0	$\frac{1}{3}$	$\cdot$
4	0	$\frac{1}{4}$	0	$\frac{1}{6}$
$P_i^{\pi(\vec{t})}$	0	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{6}$

$\vec{t}=(0,2,1,0)$				$\frac{3}{4}$
$p_{s,i}$	1	2	3	4
1	0	$\cdot$	$\cdot$	$\cdot$
2	0	0	$\cdot$	$\cdot$
3	0	0	0	$\cdot$
4	0	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
$P_i^{\pi(\vec{t})}$	0	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$

$\vec{t}=(1,0,0,0)$				$\frac{1}{2}$
$p_{s,i}$	1	2	3	4
1	0	$\cdot$	$\cdot$	$\cdot$
2	$\frac{1}{2}$	$\frac{1}{2}$	$\cdot$	$\cdot$
3	0	0	0	$\cdot$
4	0	0	0	0
$P_i^{\pi(\vec{t})}$	$\frac{1}{2}$	$\frac{1}{2}$	0	0

$\vec{t}=(1,0,1,0)$				$\frac{1}{2}$
$p_{s,i}$	1	2	3	4
1	0	$\cdot$	$\cdot$	$\cdot$
2	0	0	$\cdot$	$\cdot$
3	0	0	0	$\cdot$
4	0	0	$\frac{1}{4}$	$\frac{1}{4}$
$P_i^{\pi(\vec{t})}$	0	0	$\frac{1}{4}$	$\frac{1}{4}$

$\vec{t}=(1,1,0,0)$				$\frac{1}{2}$
$p_{s,i}$	1	2	3	4
1	0	$\cdot$	$\cdot$	$\cdot$
2	$\frac{1}{2}$	0	$\cdot$	$\cdot$
3	0	$\frac{1}{3}$	$\frac{1}{6}$	$\cdot$
4	0	0	0	0
$P_i^{\pi(\vec{t})}$	$\frac{1}{2}$	$\frac{1}{3}$	$\frac{1}{6}$	0

$\vec{t}=(1,1,1,0)$				$\frac{2}{3}$
$p_{s,i}$	1	2	3	4
1	0	$\cdot$	$\cdot$	$\cdot$
2	0	0	$\cdot$	$\cdot$
3	0	$\frac{1}{3}$	0	$\cdot$
4	0	0	$\frac{1}{4}$	$\frac{1}{6}$
$P_i^{\pi(\vec{t})}$	0	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{6}$

$\vec{t}=(1,2,0,0)$				$\frac{17}{24}$
$p_{s,i}$	1	2	3	4
1	0	$\cdot$	$\cdot$	$\cdot$
2	$\frac{1}{2}$	0	$\cdot$	$\cdot$
3	0	0	$\frac{1}{6}$	$\cdot$
4	0	$\frac{1}{4}$	0	$\frac{1}{12}$
$P_i^{\pi(\vec{t})}$	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{6}$	$\frac{1}{12}$

$\vec{t}=(1,2,1,0)$				$\frac{3}{4}$
$p_{s,i}$	1	2	3	4
1	0	$\cdot$	$\cdot$	$\cdot$
2	0	0	$\cdot$	$\cdot$
3	0	0	0	$\cdot$
4	0	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
$P_i^{\pi(\vec{t})}$	0	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$

$\vec{t} = (2,0,0,0)$				$\frac{5}{8}$
$p_{s,i}$	1	2	3	4
1	0	·	·	·
2	0	$\frac{1}{2}$	·	·
3	$\frac{1}{3}$	0	$\frac{1}{6}$	·
4	0	0	0	0
$P_i^{\pi(\vec{t})}$	$\frac{1}{3}$	$\frac{1}{2}$	$\frac{1}{6}$	0

$\vec{t} = (2,0,1,0)$				$\frac{2}{3}$
$p_{s,i}$	1	2	3	4
1	0	·	·	·
2	0	0	·	·
3	$\frac{1}{3}$	0	0	·
4	0	0	$\frac{1}{4}$	$\frac{1}{6}$
$P_i^{\pi(\vec{t})}$	$\frac{1}{3}$	0	$\frac{1}{4}$	$\frac{1}{6}$

$\vec{t} = (2,1,0,0)$				$\frac{3}{4}$
$p_{s,i}$	1	2	3	4
1	0	·	·	·
2	0	0	·	·
3	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	·
4	0	0	0	0
$P_i^{\pi(\vec{t})}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	0

$\vec{t} = (2,1,1,0)$				$\frac{5}{6}$
$p_{s,i}$	1	2	3	4
1	0	·	·	·
2	0	0	·	·
3	$\frac{1}{3}$	$\frac{1}{3}$	0	·
4	0	0	$\frac{1}{4}$	$\frac{1}{12}$
$P_i^{\pi(\vec{t})}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{12}$

$\vec{t} = (2,2,0,0)$				$\frac{5}{6}$
$p_{s,i}$	1	2	3	4
1	0	·	·	·
2	0	0	·	·
3	$\frac{1}{3}$	0	$\frac{1}{3}$	·
4	0	$\frac{1}{4}$	0	$\frac{1}{12}$
$P_i^{\pi(\vec{t})}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{12}$

$\vec{t} = (2,2,1,0)$				$\frac{11}{12}$
$p_{s,i}$	1	2	3	4
1	0	·	·	·
2	0	0	·	·
3	$\frac{1}{3}$	0	0	·
4	0	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{6}$
$P_i^{\pi(\vec{t})}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{6}$

$\vec{t} = (3,0,0,0)$				$\frac{17}{24}$
$p_{s,i}$	1	2	3	4
1	0	·	·	·
2	0	$\frac{1}{2}$	·	·
3	0	0	$\frac{1}{6}$	·
4	$\frac{1}{4}$	0	0	$\frac{1}{12}$
$P_i^{\pi(\vec{t})}$	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{1}{6}$	$\frac{1}{12}$

$\vec{t} = (3,0,1,0)$				$\frac{3}{4}$
$p_{s,i}$	1	2	3	4
1	0	·	·	·
2	0	0	·	·
3	0	0	0	·
4	$\frac{1}{4}$	0	$\frac{1}{4}$	$\frac{1}{4}$
$P_i^{\pi(\vec{t})}$	$\frac{1}{4}$	0	$\frac{1}{4}$	$\frac{1}{4}$

$\vec{t} = (3,1,0,0)$				$\frac{5}{6}$
$p_{s,i}$	1	2	3	4
1	0	·	·	·
2	0	0	·	·
3	0	$\frac{1}{3}$	$\frac{1}{3}$	·
4	$\frac{1}{4}$	0	0	$\frac{1}{12}$
$P_i^{\pi(\vec{t})}$	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{12}$

$\vec{t} = (3,1,1,0)$				$\frac{11}{12}$
$p_{s,i}$	1	2	3	4
1	0	·	·	·
2	0	0	·	·
3	0	$\frac{1}{3}$	0	·
4	$\frac{1}{4}$	0	$\frac{1}{4}$	$\frac{1}{6}$
$P_i^{\pi(\vec{t})}$	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{6}$

$\vec{t} = (3,2,0,0)$				$\frac{11}{12}$
$p_{s,i}$	1	2	3	4
1	0	·	·	·
2	0	0	·	·
3	0	0	$\frac{1}{3}$	·
4	$\frac{1}{4}$	$\frac{1}{4}$	0	$\frac{1}{6}$
$P_i^{\pi(\vec{t})}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{6}$

$\vec{t} = (3,2,1,0)$				$\frac{12}{12}$
$p_{s,i}$	1	2	3	4
1	0	·	·	·
2	0	0	·	·
3	0	0	0	·
4	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
$P_i^{\pi(\vec{t})}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$

□

We re-emphasize here the importance of the above theorem. Existence of a pure-strategy equilibrium in the case where $n = 4$ is dependent on how the employer optimally selects applicants. If her optimal policy is as presented above, then no pure-strategy equilibria exist. There, observe that when applicants play the strategies $(1, 0, 1, 0)$, the employer waits until time 4 to make a decision. But her probability of selecting the best applicant is also maximized had she chosen to make a decision at time 2, where she could choose between applicants 1 and 2. Had she followed this optimal hiring policy, then it can easily be checked that $(1, 0, 0, 0)$ is a pure-strategy equilibrium.

Having made this observation, a natural question is whether there exists a pure-strategy equilibrium for cases of $n \geq 5$, **independent** of how the employer optimally makes her hiring decision. Our next objective is to present such pure-strategy equilibria. It turns out that the structure of a pure-strategy equilibrium in the case of n odd is different from that in the case of n even. Since n odd is easier to state and prove, we shall start there.

Existence of Pure-Strategy Equilibria For $n \geq 3$, Odd

For this section, it is useful to focus on the strategy $\vec{t}$, where

$$t_i = \begin{cases} \left\lceil \frac{n}{2} \right\rceil - i & \text{for } 1 \leq i \leq \left\lceil \frac{n}{2} \right\rceil \\ n - i & \text{for } \left\lceil \frac{n}{2} \right\rceil + 1 \leq i \leq n \end{cases}$$

In other words, those in slots up to $\left\lceil \frac{n}{2} \right\rceil$ will allow the employer up to time $\left\lceil \frac{n}{2} \right\rceil$ to make her decision. Whereas those applicants in later slots will allow the employer until the end to make her decision. We shall show when n is odd, $\vec{t}$ constitutes a pure-strategy equilibrium.

Proposition 9. *Assuming n is odd. When all applicants adhere to playing $\vec{t}$, the employer selects the best applicant with probability $\frac{\lceil n/2 \rceil}{n} > \frac{1}{2}$. Each applicant in slots up to $\left\lceil \frac{n}{2} \right\rceil$ will get selected with probability $\frac{1}{\lceil n/2 \rceil}$. The rest of the applicants will not get selected.*

Proof. The employer will only make her hiring decision at one of two epochs: time $\left\lceil \frac{n}{2} \right\rceil$, or time n. In the former, her probability of selecting the best applicant is $\frac{\lceil n/2 \rceil}{n} > \frac{1}{2}$ (because n is assumed to be odd). In the latter, her probability of selecting the best applicant is $1 - \frac{\lceil n/2 \rceil}{n} < \frac{1}{2}$. As such, she will make her decision at time $\left\lceil \frac{n}{2} \right\rceil$, and only selects from among these $\left\lceil \frac{n}{2} \right\rceil$ applicants. It is clear that each of these applicants gets selected with probability $\frac{1}{\lceil n/2 \rceil}$.

□

The next lemma shows that if all other applicants adhere to playing the strategy $\vec{t}_{-i}$, then applicant i, where $1 \leq i \leq \left\lceil \frac{n}{2} \right\rceil$, does not want his window of availability to end before time $\left\lceil \frac{n}{2} \right\rceil$. Note that if $\left\lceil \frac{n}{2} \right\rceil < i \leq n$, then however he changes his strategy will not matter, as the employer will always choose at time point $\left\lceil \frac{n}{2} \right\rceil$, and therefore this applicant who comes after time $\left\lceil \frac{n}{2} \right\rceil$ will not get selected.

Lemma 8. *Consider an arbitrary applicant i, where $1 \leq i \leq \left\lceil \frac{n}{2} \right\rceil - 1$, and suppose he changes his window of availability to end before time $\left\lceil \frac{n}{2} \right\rceil$. In other words, his new t'_i is such that $i + t'_i \leq \left\lceil \frac{n}{2} \right\rceil - 1$. If all other applicants adhere to the strategy $\vec{t}_{-i}$, then applicant i will get selected with probability 0.*

Proof. Let us denote the last time where applicant i is available as $\tilde{t}$; that is, $i + t'_i = \tilde{t}$. Due to the employer's objective, she must use a policy of hiring the best so far starting at one of three possible time epochs: $\tilde{t}$, $\lceil \frac{n}{2} \rceil$, or n. Let Π_α (Π_β) denote her policy of hiring the best so far starting at $\tilde{t}$ ($\lceil \frac{n}{2} \rceil$). Consider the following three possible scenarios.

- ith applicant is best out of $\tilde{t}$, but not best out of $\lceil \frac{n}{2} \rceil$.
  - This event happens with probabilty $\frac{1}{\tilde{t}} - \frac{1}{\lceil n/2 \rceil}$.
  - $\Pr[\Pi_\alpha$ selects best applicant$] = 0$.
  - $\Pr[\Pi_\beta$ selects best applicant$] = \frac{\lceil n/2 \rceil}{n}$.
- ith applicant is best out of $\tilde{t}$, and is best out of $\lceil \frac{n}{2} \rceil$.
  - This event happens with probability $\frac{1}{\lceil n/2 \rceil}$.
  - $\Pr[\Pi_\alpha$ selects best applicant$] = \frac{\lceil n/2 \rceil}{n}$.
  - $\Pr[\Pi_\beta$ selects best applicant$] = 1 - \frac{\lceil n/2 \rceil}{n}$.
- ith applicant is not the best out of $\tilde{t}$.
  - This event happens with probability $1 - \frac{1}{\tilde{t}}$.
  - $\Pr[\Pi_\alpha$ selects best applicant$] = \frac{\lceil n/2 \rceil}{n}$.
  - $\Pr[\Pi_\beta$ selects best applicant$] = \frac{\lceil n/2 \rceil}{n}$.

From these, it follows that:

$$\begin{aligned}\Pr[\Pi_\alpha \text{ selects best applicant}] &= \frac{\lceil n/2 \rceil}{n} \cdot \frac{1}{\lceil n/2 \rceil} + \frac{\lceil n/2 \rceil}{n} \cdot \left(1 - \frac{1}{\tilde{t}}\right) \\ &= \frac{1}{n} + \frac{\lceil n/2 \rceil}{n} - \frac{\lceil n/2 \rceil}{\tilde{t}n}\end{aligned}$$

and

$$\begin{aligned}\Pr[\Pi_\beta \text{ selects best applicant}] &= \frac{\lceil n/2 \rceil}{n} \cdot \frac{1}{\tilde{t}} - \frac{1}{n} + \frac{1}{\lceil n/2 \rceil} - \frac{1}{n} + \frac{\lceil n/2 \rceil}{n} - \frac{\lceil n/2 \rceil}{\tilde{t}n} \\ &= \frac{1}{\lceil n/2 \rceil} + \frac{\lceil n/2 \rceil}{n} - \frac{2}{n}\end{aligned}$$

Observe that $\Pr[\Pi_\alpha$ selects best applicant$] < \Pr[\Pi_\beta$ selects best applicant$]$ if and only if $\frac{3}{n} < \frac{1}{\lceil n/2 \rceil} + \frac{\lceil n/2 \rceil}{\tilde{t}n}$, which holds for all odd $n \geq 3$, and $\tilde{t} \leq \lceil \frac{n}{2} \rceil - 1$.

As such, the employer will never use policy Π_α, and hence applicant i will never get selected. This proves the lemma.

□

Although not needed for the proof above, we note that the policy of starting the selection at time n is also dominated by Π_β. It is not needed because that policy also does not select the ith applicant, so that he does not have incentive to change his window of availability.

The next lemma shows that it also does not pay for an applicant to hold off until a time beyond $\lceil \frac{n}{2} \rceil$.

Lemma 9. *Consider an arbitrary applicant i, where $1 \leq i \leq \lceil \frac{n}{2} \rceil$, and suppose he changes his window of availability to end after time $\lceil \frac{n}{2} \rceil$. In other words, his new t_i' is such that $i + t_i' \geq \lceil \frac{n}{2} \rceil + 1$. If all other applicants adhere to the strategy $\vec{t}_{-i}$, then applicant i will get selected with probability strictly less than $\frac{1}{\lceil n/2 \rceil}$.*

Proof. Let us denote the last time where applicant i is available as $\tilde{t}$; that is, $i + t_i' = \tilde{t}$. Due to the employer's objective, she must use a policy of hiring the best so far starting at one of three possible time epochs: $\lceil \frac{n}{2} \rceil$, $\tilde{t}$, or n. Let Π_α (Π_β) denote her policy of hiring the best so far starting at $\tilde{t}$ ($\lceil \frac{n}{2} \rceil$). Consider the following three possible scenarios: (1) the ith applicant is best out of $\lceil \frac{n}{2} \rceil$, but not best out of $\tilde{t}$; (2) the ith applicant is best out of $\lceil \frac{n}{2} \rceil$, and also is best out of $\tilde{t}$; and (3) the ith applicant is not the best out of $\lceil \frac{n}{2} \rceil$. Observe that both Π_α and Π_β select the best applicant with probability 1 in case (1), and that they both select the best applicant with probability $\frac{\tilde{t}}{n}$ in case (2). In case (3), the probability is $1 - \frac{\lceil n/2 \rceil}{n}$ for Π_α, which is strictly less than the probability of $\frac{\lceil n/2 \rceil}{n}$ for Π_β. As such, the employer will choose to use Π_β.

Using a similar argument, it can be shown that the policy of waiting until time n to make a decision is strongly dominated by Π_β as well. As such, policy Π_β will be used.

Given that the employer uses Π_β to optimally select the best candidate, the ith applicant will only get hired in scenario (2), which happens with probability $\frac{1}{\tilde{t}}$. Observe that this probability is strictly less than $\frac{1}{\lceil n/2 \rceil}$, which is the probability the ith applicant would have been selected had he chosen to play $\tilde{t} = \lceil n/2 \rceil$. Therefore, he will not delay to beyond time $\lceil n/2 \rceil$.

□

The previous two lemmas allow us to conclude with the following theorem.

Theorem 5. *With $n \geq 3$ and n being odd, then $\vec{t}$ is a pure-strategy equilibrium.*

Proof. For $1 \leq i \leq \lceil \frac{n}{2} \rceil$, applicant i does not want to change his strategy since his probability of getting selected will be lowered. For $\lceil \frac{n}{2} \rceil < i \leq n$, applicant i changing his strategy does not matter as his probability of getting selected still remains at 0. As such, $\vec{t}$ is a pure-strategy equilibrium.

□

As was observed for the case $n = 4$, this constructed strategy is not always a pure-strategy equilibrium for even n. Whether it is an equilibrium or not depends on how the employer optimally makes her hiring decision. Given $\vec{t}$, she has equal probability of selecting the best applicant (at $\frac{1}{2}$) by either choosing at $\lceil \frac{n}{2} \rceil$, or at n. If she makes her hiring decision at $\lceil \frac{n}{2} \rceil$, then $\vec{t}$ is a pure-strategy equilibrium by arguments in the previous theorem. If, on the other hand, she makes her hiring decision at n, then $\vec{t}$ is not an equilibrium due to early applicants having a better chance of getting selected by changing their windows of availability to end at n (from 0 to $\frac{1}{n}$).

Thus, we now place our focus on finding a pure Nash equilibrium for $n \geq 6$, n being even, independent of the employer's optimal hiring policy.

Existence of Pure-Strategy Equilibria For $n \geq 6$, Even

In this subsection, we will show there exists a pure-strategy equilibrium for $n \geq 6$, and n being even, independent of the employer's optimal hiring policy. The proof is constructive, and as such, it is useful to define $\vec{t}$ as the strategy where

$$t_i = \begin{cases} \frac{n}{2} - i & \text{for } 1 \leq i \leq \frac{n}{2} \\ n - 1 - i & \text{for } \frac{n}{2} + 1 \leq i \leq n - 1 \\ 0 & \text{for } i = n \end{cases}$$

In other words, under the collective strategy $\vec{t}$, the first half of applicants will all allow the employer until time $\frac{n}{2}$ to make her hiring decision. The second half of applicants (except the last one) will allow the employer until time $n - 1$ to make her hiring decision. And the applicant in the nth slot has no other choice but to remain in that position.

Proposition 10. *When all applicants adhere to playing $\vec{t}$, the employer selects the best applicant with probability $\frac{1}{2}$. Each applicant in slots up to $\frac{n}{2}$ will get selected with probability $\frac{2}{n}$. The rest of the applicants will not get selected.*

Proof. When n is even, if the employer selects the best one at time $\frac{n}{2}$, then her probability of selecting the best applicant is $\frac{n/2}{n} = \frac{1}{2}$. If she waits until a later time, then her probability of selecting the best applicant is strictly less than $\frac{1}{2}$, as there are $\frac{n}{2}$ applicants left, and at time $n - 1$ she does not know the total order of all applicants (the last applicant has yet to appear). As such, she will make her hiring decision at time $\frac{n}{2}$. Under this optimal hiring policy, observe that each applicant $1 \leq i \leq \frac{n}{2}$ gets selected with probability $\frac{1}{n/2}$. The others are selected with probability 0. $\square$

The next lemma shows that if all other applicants adhere to playing the strategy $\vec{t}_{-i}$, then applicant i, where $1 \leq i \leq \frac{n}{2}$, does not want his window of availability to end before time $\frac{n}{2}$. Note that if $\frac{n}{2} < i \leq n$, then however he changes his strategy will not matter, as he will not get selected (holds for n even, and $n \geq 6$; in the case $n = 4$, the third applicant delaying to time 4 will allow the employer to be able to select the best applicant with probability $\frac{1}{2}$ by hiring at time 4).

Lemma 10. *Consider an arbitrary applicant i, where $1 \leq i \leq \frac{n}{2} - 1$, and suppose he changes his window of availability to end before time $\frac{n}{2}$. In other words, his new t'_i is such that $i + t'_i \leq \frac{n}{2} - 1$. If all other applicants adhere to the strategy $\vec{t}_{-i}$, then applicant i will get selected with probability 0.*

Proof. Denote $\tilde{t} = i + t'_i$, so that $\tilde{t}$ is the last time applicant i is available for hire. Although $\lceil \frac{n}{2} \rceil$ is equivalent to $\frac{n}{2}$ when n is even, we will use the former notation throughout this proof to illustrate a point: that this proposed strategy is not an equilibrium when n is odd. Now, observe that the employer will only make her hiring decisions starting at one of the time epochs $\tilde{t}$, $\lceil \frac{n}{2} \rceil$, $n-1$, or n. Let Π_α be the policy where the employer hires the best so far starting at time $\tilde{t}$. Also let Π_β be the policy where the employer hires the best so far starting at time $\lceil \frac{n}{2} \rceil$. We first show that the employer will choose applicants with the policy Π_β. Consider the following three scenarios:

- ith applicant is best at time $\tilde{t}$, but is not best at time $\lceil \frac{n}{2} \rceil$.
  - This event occurs with probability $\frac{1}{\tilde{t}} - \frac{1}{\lceil n/2 \rceil}$.
  - Pr[Π_α selects best | event] = 0. This is because the policy Π_α would have already picked the ith applicant at time $\tilde{t}$, which is not the best overall.
  - Pr[Π_β selects best | event] = $\frac{\lceil n/2 \rceil}{n}$. This is because the policy Π_β would have been able to pick the best so far at time $\lceil \frac{n}{2} \rceil$.
- ith applicant is best at time $\tilde{t}$, and is also best at time $\lceil \frac{n}{2} \rceil$.
  - This event occurs with probability $\frac{1}{\lceil n/2 \rceil}$.
  - Pr[Π_α selects best | event] = $\frac{\lceil n/2 \rceil}{n}$. This is because the policy Π_α would have picked the ith applicant at time $\tilde{t}$.
  - We need to decompose this event further into three smaller sub-events.
    1. ith applicant is also best so far at time $n-1$, and is also best so far at time n.
       * This event occurs with probability $\frac{1}{n}$.
       * Pr[Π_β selects best | smaller event] = 0. This is because the policy Π_β started hiring at time $\lceil \frac{n}{2} \rceil$, for which the ith applicant is already unavailable.
    2. ith applicant is also best so far at time $n-1$, but not best so far out of n.
       * This event occurs with probability $\frac{1}{n-1} - \frac{1}{n}$.
       * Pr[Π_β selects best | smaller event] = 1. This is because no applicants available for hire up until time $n-1$ is the best so far. As such, only at time n can Π_β hires the best so far, which turns out to be the best overall as well.
    3. ith applicant is not the best so far at time $n-1$.
       * This event occurs with probability $\frac{1}{\lceil n/2 \rceil} - \frac{1}{n-1}$.
       * Pr[Π_β selects best | smaller event] = $\frac{n-1}{n}$. This is because the best so far is not available at time $\lceil \frac{n}{2} \rceil$, but is available at time $n-1$.
- ith applicant is not best at time $\tilde{t}$.

- This event occurs with probability $1 - \frac{1}{\tilde{t}}$.
- $\Pr[\Pi_\alpha$ selects best | event$] = \frac{\lceil n/2 \rceil}{n}$. This is because the best so far will surely be available at time $\lceil \frac{n}{2} \rceil$.
- $\Pr[\Pi_\beta$ selects best | event$] = \frac{\lceil n/2 \rceil}{n}$. The reasoning is the same as above.

From these, we observe that:

$$\begin{aligned}
\Pr[\Pi_\alpha \text{ selects best}] < \Pr[\Pi_\beta \text{ selects best}] \quad &\iff \quad \tfrac{1}{n} < \tfrac{\lceil n/2 \rceil}{n\tilde{t}} - \tfrac{1}{n} + \tfrac{1}{n-1} - \tfrac{1}{n} + \tfrac{n-1}{n\lceil n/2 \rceil} - \tfrac{1}{n} \\
&\iff \quad \tfrac{4}{n} - \tfrac{1}{n-1} - \tfrac{n-1}{n\lceil n/2 \rceil} < \tfrac{\lceil n/2 \rceil}{n\tilde{t}}
\end{aligned}$$

For $i \leq \tilde{t} \leq \lceil \frac{n}{2} \rceil - 1$, the right hand side is smallest at $\tilde{t} = \lceil \frac{n}{2} \rceil - 1$. We shall check for the strict inequality for this value of $\tilde{t}$. This strict inequality will be shown to **hold** for the case where n is even, and **fails** in the case where n is odd. As such, this strategy is only a candidate for equilibrium when there are an even number of applicants.

1. $n = 2k$, for $k \geq 1$.

$$\begin{aligned}
\tfrac{4}{n} - \tfrac{1}{n-1} - \tfrac{n-1}{n\lceil n/2 \rceil} < \tfrac{\lceil n/2 \rceil}{n\tilde{t}} \quad &\iff \quad \tfrac{4}{2k} - \tfrac{1}{2k-1} - \tfrac{2k-1}{2k^2} < \tfrac{k}{2k\cdot(k-1)} \\
&\iff \quad \tfrac{2}{k} - \tfrac{1}{2k-1} - \tfrac{1}{k} + \tfrac{1}{2k^2} < \tfrac{1}{2(k-1)} \\
&\iff \quad \tfrac{2(k-1)}{k} - \tfrac{2k-2}{2k-1} + \tfrac{k-1}{k^2} < 1 \\
&\iff \quad \tfrac{1}{2k-1} < \tfrac{1}{k}\left(1 + \tfrac{1}{k}\right) \\
&\iff \quad 1 < \left(2 - \tfrac{1}{k}\right)\left(1 + \tfrac{1}{k}\right)
\end{aligned}$$

   But this strict inequality is obviously true for all $k \geq 1$. Hence the employer will use the policy Π_β, and as such the ith applicant gets selected with probability 0. Since his probability of getting selected had he stayed with the strategy t_i is $\frac{1}{n/2}$, he does not have incentive to play t_i'.

2. $n = 2k + 1$, for $k \geq 2$.

$$\begin{aligned}
\tfrac{4}{n} - \tfrac{1}{n-1} - \tfrac{n-1}{n\lceil n/2 \rceil} < \tfrac{\lceil n/2 \rceil}{n\tilde{t}} \quad &\iff \quad \tfrac{4}{2k+1} - \tfrac{1}{2k} - \tfrac{2k}{(2k+1)(k+1)} < \tfrac{k+1}{(2k+1)\cdot k} \\
&\iff \quad 4 - \tfrac{2k+1}{2k} - \tfrac{2k}{k+1} < \tfrac{k+1}{k} \\
&\iff \quad 2 < \tfrac{3/2}{k} + \tfrac{2k}{k+1} \\
&\iff \quad 2 < \tfrac{(3/2)(k+1)+2k^2}{k(k+1)}
\end{aligned}$$

   which fails for $k \geq 4$.

   And since $\Pr[\Pi_\alpha$ selects best$] > \Pr[\Pi_\beta$ selects best$]$ when an arbitrary applicant i shortens his window of availability to $\tilde{t} = \lceil \frac{n}{2} \rceil - 1$, the employer will use policy Π_α

in this instance. But that implies the applicant i with $\tilde{t}$ will get selected in the first two scenarios for a total probability of $\frac{1}{\tilde{t}} - \frac{1}{\lceil n/2 \rceil} + \frac{1}{\lceil n/2 \rceil} = \frac{1}{\tilde{t}} > \frac{1}{\lceil n/2 \rceil}$. Hence the ith applicant has incentive of playing a different strategy.

Although we did not explicitly compare Π_β and an optimal policy which starts at time $n-1$, it can be shown that the latter is dominated by the former in both cases of n being even or odd (its probability of selecting the best applicant is $\frac{n-1-\lceil n/2 \rceil}{n} + \frac{1}{n}\frac{\lceil n/2 \rceil}{n-1}$). We ignore this policy because it will choose the ith applicant with probability 0 for $1 \leq i \leq \lceil \frac{n}{2} \rceil$. A similar argument applies for an optimal policy that starts at time n.

□

We have shown how each applicant in slots between 1 and $\frac{n}{2}$ does not want to shorten his window of availability given all others play the strategy $\vec{t}_{-i}$. In the next lemma, we shall show the complement fact: that they also do not want to lengthen their windows of availability.

Lemma 11. *Consider an arbitrary applicant i, where $1 \leq i \leq \frac{n}{2}$, and suppose he changes his window of availability to end after time $\frac{n}{2}$. In other words, his new t'_i is such that $i + t'_i \geq \frac{n}{2} + 1$. If all other applicants adhere to the strategy $\vec{t}_{-i}$, then applicant i will get selected with probability strictly less than $\frac{2}{n}$.*

Proof. Denote $\tilde{t} = i + t'_i$, so that $\tilde{t}$ is the last time applicant i is available for hire. Observe that the employer will only make her hiring decisions starting at one of the time epochs $\frac{n}{2}$, $\tilde{t}$, $n-1$, or n. Let Π_α be the policy where the employer hires the best so far starting at time $\tilde{t}$. Also let Π_β be the policy where the employer hires the best so far starting at time $\frac{n}{2}$. We first show that the employer will choose applicants with the policy Π_β. Consider the following three scenarios:

- ith applicant is best at time $\frac{n}{2}$, but is not best at time $\tilde{t}$. When $\tilde{t} \leq n-1$, both Π_α and Π_β select the best applicant with probability $\frac{n-1}{n}$. When $\tilde{t} = n$ and the ith applicant is also best at time $n-1$, then both policies select the best applicant with probability 1. When $\tilde{t} = n$ and the ith applicant is not the best at time $n-1$, then Π_α selects the best with probability $\frac{1}{n}$, and Π_β selects the best with probability $\frac{n-1}{n}$. As such, Π_β strongly dominates Π_α in this scenario.

- ith applicant is best at time $\frac{n}{2}$, and is best at time $\tilde{t}$. The both policies Π_α and Π_β select the best applicant with probability $\frac{\tilde{t}}{n}$ for all $\frac{n}{2} + 1 \leq \tilde{t} \leq n$.

- ith applicant is not best at time $\frac{n}{2}$. Then Π_α selects the best applicant with probability **strictly less than** $\frac{1}{2}$. Π_β, on the other hand, selects the best applicant with probability $\frac{1}{2}$. So that Π_β again strongly dominates Π_α in this scenario.

It follows that the employer will strongly prefer policy Π_β over Π_α. We also observe that an optimal policy which begins at time $n-1$ is dominated by policy Π_β (in the second and third scenarios, it selects the best applicant with probability less than $\frac{1}{2}$). Furthermore, one which begins at time n is also dominated by Π_β. As such, the employer will use Π_β to select applicants.

Under the policy Π_β, the ith applicant only gets selected in scenario (2), and his probability of getting selected is $\frac{1}{t} < \frac{1}{n/2}$. As such, he does not have incentive to lengthen his window of availability to beyond time $\frac{n}{2}$.

□

These results allowed us to conclude with this theorem.

Theorem 6. *With $n \geq 6$ and n being even, then $\vec{t}$ is a pure-strategy equilibrium.*

Proof. For $1 \leq i \leq \frac{n}{2}$, applicant i does not want to change his strategy since his probability of getting selected will be lowered. For $\frac{n}{2} < i \leq n$, applicant i changing his strategy does not matter as his probability of getting selected still remains at 0. As such, $\vec{t}$ is a pure-strategy equilibrium.

□

Pareto Optimality And Other Conjectures

It turns out that our constructed equilibrium strategies have a very desirable property: they are Pareto optimal. This fact is proved below.

Proposition 11. *The constructed equilibria are Pareto-optimal (and hence, by definition, payoff-dominant).*

Proof. Let p_i^π denote the probability for which a policy π selects applicant i, observe that $\sum_{i=1}^{n} p_i^\pi \leq 1$. But in our constructed equilibrium, the optimal policy selects each of the first $\left\lceil \frac{n}{2} \right\rceil$ with probability $\frac{1}{\lceil n/2 \rceil}$, so that $\sum_{i=1}^{n} p_i^\pi = 1$. As such, if we increase the payoff of one applicant, some other applicant's payoff must decrease, so that our constructed equilibria are at the Pareto frontier.

□

There are other properties of the game that we observed, but could not prove. These are listed as conjectures here.

- All equilibria for even $n \geq 4$ have employer's payoff of at least $\frac{1}{2}$. As such, our constructed pure-strategy equilibria are employer's payoff minimum.
- All equilibria for odd $n \geq 3$ have employer's payoff of exactly $\frac{\lceil n/2 \rceil}{n}$.
- Taken together, the lower bound for the employer's payoff in an equilibrium is $\frac{1}{2}$.
- As $n \to \infty$, the maximum employer's payoff goes to $\frac{1}{2}$ in an equilibrium.

- For $n \geq 8$, there exists an equilibrium where all applicants are selected with positive probability.

Other Equilibria In The Strategic Secretary Game

Having shown existence of a pure strategy equilibrium in each case where $n \neq 4$, a natural question to ask is whether our constructed equilibria are unique in all instances of n. This statement turns out to be false, as we shall present all equilibria for $3 \leq n \leq 7$ in this section. It should be noted that these equilibria are constructed from payoff matrices as a result of solving linear programs using the GNU Linear Programming Kit (GLPK). Other solvers may give different sets of optimal solutions, and in turn change the payoff matrices, which give rise to other equilibria (see the case of $n = 4$ discussed earlier of why this is so). The GAMBIT Game Analyzer [9] software was used to obtain these results.

The information below can be read as follows (for $n = 3$): a (pure) equilibrium is obtained when applicants use the strategy profile $\vec{t} = (1, 0, 0)$, i.e. $t_1 = 1$, $t_2 = 0$, and $t_3 = 0$. In this equilibrium, the employer hires the best applicant with probability $\frac{2}{3}$. We also present two tables for the case $n = 4$ to illustrate the importance of the employer's optimal decision in affecting equilibria in this game. The sole difference between these two are in their payoff matrices for $\vec{t} = (1, 0, 1, 0)$, where the employer decides at time 2 for the former, and at time 4 for the latter. The sets of equilibria, as a result, are also different for these two cases. Here, $P^{\pi}(\vec{t})$ denotes the employer's probability of selecting the best applicant under this equilibrium played with the strategy profile $\vec{t}$, and i_j denotes the probability applicant i will play the strategy $t_i = j$.

- $n = 3$:

$P^{\pi}(\vec{t})$	1_0	1_1	1_2	2_0	2_1	3_0
$\frac{2}{3}$	0	1	0	1	0	1

- $n = 4$ (employer decides at time 2 when $\vec{t} = (1, 0, 1, 0)$):

$P^{\pi}(\vec{t})$	1_0	1_1	1_2	1_3	2_0	2_1	2_2	3_0	3_1	4_0
$\frac{1}{2}$	0	1	0	0	1	0	0	1	0	1
$\frac{1}{2}$	0	1	0	0	1	0	0	$\frac{1}{3}$	$\frac{2}{3}$	1
$\frac{31}{45}$	0	$\frac{1}{3}$	$\frac{2}{3}$	0	$\frac{1}{3}$	$\frac{2}{3}$	0	$\frac{3}{5}$	$\frac{2}{5}$	1

- $n = 4$ (employer waits until time 4 to decide when $\vec{t} = (1, 0, 1, 0)$):

$P^{\pi}(\vec{t})$	1_0	1_1	1_2	1_3	2_0	2_1	2_2	3_0	3_1	4_0
$\frac{17}{24}$	0	$\frac{1}{2}$	$\frac{1}{2}$	0	0	1	0	$\frac{2}{3}$	$\frac{1}{3}$	1
$\frac{17}{24}$	0	0	1	0	$\frac{1}{2}$	$\frac{1}{2}$	0	$\frac{2}{3}$	$\frac{1}{3}$	1

- $n = 5$:

$P^\pi(\vec{t})$	1_0	1_1	1_2	1_3	1_4	2_0	2_1	2_2	2_3	3_0	3_1	3_2	4_0	4_1	5_0
$\frac{3}{5}$	0	0	1	0	0	0	1	0	0	1	0	0	1	0	1
$\frac{3}{5}$	0	0	1	0	0	0	1	0	0	1	0	0	0	1	1

- $n = 6$:

$P^\pi(\vec{t})$	1_2	2_1	3_0	4_0	4_1	5_0	6_0
$\frac{1}{2}$	1	1	1	1	0	1	1
$\frac{1}{2}$	1	1	1	0	1	1	1

- $n = 7$:

$P^\pi(\vec{t})$	1_3	2_2	3_1	4_0	5_1	5_2	6_0	6_1	7_0
$\frac{4}{7}$	1	1	1	1	1	0	1	0	1
$\frac{4}{7}$	1	1	1	1	1	0	0	1	1
$\frac{4}{7}$	1	1	1	1	0	1	1	0	1
$\frac{4}{7}$	1	1	1	1	0	1	0	1	1

For $n = 8$, one version of the payoff matrix gives rise to 49 equilibria, while another gives 29. For $n = 9$, our Duo Core Intel Pentium 4 CPU 2.80GHZ and 1GB RAM never completed the equilibrium computation.

Discussion Of Constructed and Other Nash Equilibria

We have shown how to construct strategy profiles that give rise to pure equilibria in all games with $n \neq 4$. Furthermore, we exhibited all Nash equilibria for selected cases of $3 \leq n \leq 7$. In this section, we will take a look at our constructed and these other equilibria.

For $n = 3$, the strategy profile makes sense, as the constructed pure strategy Nash equilibrium is a result of dominant strategies employed by applicants. From the tables presented earlier for $n = 3$, observe that the 1st applicant's dominant strategy is to play $t_1 = 1$, and the 2nd applicant's dominant strategy is to play $t_2 = 0$. Since the third applicant has only one choice to make, his is trivially dominant. As such, these dominant strategies result in the Nash equilibrium presented.

Using Gambit and the *dominance* tool feature, we check for dominating strategies in a couple of games with $n > 3$. For the case of $n = 4$ where the employer waits until time 2 to make her decision, our constructed pure-strategy equilibrium is not dominating. Furthermore, there is not a strongly dominating strategy profile for all applicants; as such, it is difficult for us to determine how the game will be played out.

With $n = 5$, our constructed pure strategy is weakly dominated. Since $t_1 = 2, t_2 = 1, t_3 = 0$ are strongly dominating strategies for applicants 1, 2, and 3, respectively, it is sensible to assume that they will play these. Given this, no matter how the 4th applicant plays will result in his probability of getting selected being 0. Therefore, if he stays with using a pure strategy, then the game will end up in a Nash equilibrium.

In $n = 6$, there are no dominating strategies for all applicants. Again, as in the case of $n = 4$, we cannot be certain of how this game will play out.

For $n = 7$, again there are no dominating strategies for all applicants. It is not clear if participants will use strategies that would result in any of the presented equilibria constructed. In fact, we conjecture that this pattern will continue for games of higher value of n, namely the nonexistence of dominating strategies that would result in a Nash equilibrium.

4.4 Concluding Notes On Game-Theoretic Approach

In this chapter, we presented two game-theoretic approaches to the classical secretary problem. By allowing applicants the freedom to determine their windows of availability with non-binding commitment, the employer can improve her probability of selecting the best candidate and gain the desired property that everyone who is interviewed will get hired with the same probability. If the employer chooses to solicit every applicant's window of availability before the first interview begins, and makes this a binding commitment, then the resulting simultaneous game is guaranteed to almost always have a pure-strategy equilibrium. The lone exception to this fact is when $n = 4$, where its existence is dependent on how the employer optimally selects applicants. These equilibria are easy to characterize: applicants before time $\lceil \frac{n}{2} \rceil$ will wait for a decision until this time, and applicants after time $\lceil \frac{n}{2} \rceil$ will wait for a decision either at time $n - 1$, or at time n. For some values of n, our constructed equilibria are good indicators of how applicants will strategize. For many other values of n, we believe different equilibria, or none at all, might be agreed on instead. We used Gambit to enumerate all equilibria presented in this strategic game, and illustrated our arguments for particular cases of $3 \leq n \leq 7$. Lastly, we showed certain properties of our constructed equilibria, and left a few interesting conjectures concerning this secretary game.

Chapter 5

Concluding Notes

We come to the end of this dissertation hoping to have made the case for the need to consider participants' behavior in all optimization problems. Our setting throughout this dissertation has mostly been confined to a hiring process, and our argument was for new approaches to the secretary problem. Our main nitpick is the complete ignorance of human incentives in previous attempts at an optimal solution. We extended, compared, and constrasted an old approach used on another variant of the problem, and showed results which are markedly different from those found in prior works. We also proposed an alternative game-theoretic solution, which makes use of the inherent competition between players in the process, giving them more room to compete, and by doing so, improve the employer's chance of meeting her goal. Simple modifications, like those we have made, change the dynamic of the problem completely and provide new insights into how people should approach the process. We believe our framework for analysis can be extended to other sequential decision settings, where strategic choices of participants play an important role in the process. Whether it's job hunting, dating, or whatever else that you can think of, good luck searching!

Bibliography

[1] Niv Buchbinder, Kamal Jain, and Mohit Singh. "Incentives in Online Auctions via Linear Programming". In: *WINE'10 Proceedings of the 6th International Conference on Internet and Network Economics* (2010), pp. 106–117.

[2] Niv Buchbinder, Kamal Jain, and Mohit Singh. "Secretary Problems via Linear Programming". In: *IPCO'10: 14th International Conference On Integer Programming and Combinatorial Optimization* (2010), pp. 163–176.

[3] Y.S. Chow et al. "Optimal Selection Based On Relative Rank". In: *Israel Journal of Mathematics* 2.2 (1964), pp. 81–90.

[4] P.R. Freeman. "The Secretary Problem and its Extensions: A Review". In: *International Statistical Review* 51 (1983), pp. 189–206.

[5] P.R. Freeman. "Who Solved the Secretary Problem?" In: *Statistical Science* 4.3 (1989), pp. 282–296.

[6] M. Gardner. "Mathematical Games". In: *Scientific American* (1960), pp. 150–153.

[7] J.P. Gilbert and F. Mosteller. "Recognizing The Maximum of a Sequence". In: *Journal of American Statistical Association* 61 (1966), pp. 35–79.

[8] D.V. Lindley. "Dynamic Programming and Decision Theory". In: *Applied Statistics* 10.1 (1961), pp. 39–51.

[9] R.D. McKelvey, A.M. McLennan, and T.L. Turocy. "Gambit: Software Tools for Game Theory, Version 13.1.0". In: (2013).

[10] John Nash. "Equilibrium points in n-person games". In: *Proceedings of the National Academy of Sciences* 36.1 (1950), pp. 48–49.

[11] John Nash. "Non-Cooperative Games". In: *The Annals of Mathematics* 54.2 (1951), pp. 286–295.

[12] M.C.K. Yang. "Recognizing the Maximum of a Random Sequence Based On Relative Rank With Backward Solicitation". In: *Journal of Applied Probability* 11 (1974), pp. 504–512.