On a Competitive Secretary Problem

Anna Karlin and **Eric Lei**
karlin@cs.washington.edu
elei@andrew.cmu.edu

Abstract

Consider a scenario in which there are multiple employers competing to hire the best possible employee. How does the competition between the employers affect their hiring strategies or their ability to hire one of the best possible candidates? In this paper, we address this question by studying a generalization of the classical secretary problem from optimal stopping theory: a set of ranked employers compete to hire from the same random stream of employees, and each employer wishes to hire the best candidate in the bunch. We show how to derive subgame-perfect Nash equilibrium strategies in this game and analyze the impact the competition has on the quality of the hires as a function of the rank of the employer. We present numerical results from simulations of these strategies.

1 Introduction

The *secretary problem* (Ferguson 2012; Freeman 1983) is a classical problem in optimal stopping theory, and can be formulated as follows: An employer wishes to hire a single employee out of a pool of n totally ordered candidates. The applicants are interviewed in a random order. Immediately after each interview, an irrevocable hiring decision is made. The decision is made knowing only how that candidate compares to all candidates that have been interviewed up to that point, but not how he compares to future candidates. Moreover, an applicant that has been rejected cannot be later hired. It is well known that if the goal of the employer is to maximize the probability of hiring the very best of the n candidates, then her optimal strategy is to wait until she has seen roughly an e^{-1} fraction of the applicants, and then hire the first one better than the best she has seen so far.

In this paper, we consider a version of this problem in a *competitive setting*. Consider, for example, multiple top computer science departments all considering the same set of faculty candidates, and competing to hire them. This changes the problem from one of decision theory to one of game theory, and raises the following questions: What are the optimal strategies of the employers in equilibrium? How well do the employers and applicants do in the presence of this competition?

We explore these questions in the context of the following stylized model: There are k totally ordered employers and n totally ordered applicants. The relative ranks of the applicants are initially unknown to the employers. The applicants arrive one by one in a random order and at the moment they arrive all the employers learn their rank relative to all applicants that have arrived earlier. As each applicant arrives, any number of the employers may choose to make her an offer. Again these decisions are irrevocable; there is no possibility of making her an offer later. Since the employers are totally ranked, an applicant that receives multiple offers will accept the offer from the highest ranked employer among those making her an offer. We assume that each employer wants to hire only one person, so once he has successfully hired, he will make no further offers.

We consider two possible objectives for the employers. The first is when each employer is trying to maximize the probability of hiring the very best applicant. In Section 4, we use dynamic programming to find a *subgame-perfect Nash equilibrium* for this game. A set of strategies is a subgame-perfect Nash equilibrium if (a) it is a Nash equilibrium—that is, each employer's strategy is a best-response to the strategies used by the other employers (note that the applicants' strategies are trivial: they simply accept the best offer they receive)—and (b) it remains a Nash equilibrium with respect to each subgame, the game that remains after m applicants have arrived, for each m.

Consider, for example, the case where $k = 2$, in which exactly two employers, say I and II, are competing for the best candidate. The strategy of employer I, the top-ranked employer, is simple: he just runs the standard secretary algorithm, since offers made by employer II would always be rejected in favor of offers from employer I. Employer II, on the other hand, must take employer I's strategy into account. It turns out that the best (subgame-perfect) response for employer II in the limit is to wait until an $e^{-3/2}$ fraction of the candidates have been interviewed, and then make offers to any candidate better than the best seen so far, until employer II successfully hires someone. (Employer II's first offer might be to the same person employer I makes an offer to, so II might end up making 2 offers.) Also, just as the top employer has probability e^{-1} of success (the same as the fraction of candidates he interviews before considering any offer), the probability that II successfully hires the best applicant is in fact $e^{-3/2}$. More generally, the lower the rank of the employer, the earlier they will start making offers and, of course, the lower their probability of success.

The second objective we consider is when, for each j, the j-th ranked employer seeks to hire any one of the top j applicants. In Section 5, we show how to find a subgame-perfect Nash equilibrium for this objective. Specifically, we show that the structure of the unique subgame-perfect equilibrium strategies used is the following: Consider, say the j-th ranked employer. Then there is a set of numbers $\alpha_1 > \alpha_2 > \cdots > \alpha_j$ between 0 and 1 such that employer j will attempt to make an offer to the i-th best applicant so far or better after an α_i fraction of applicants have been seen. However, these thresholds vary over time, depending on whether higher ranked employers have already hired someone or not. After higher-ranked employers have hired, the thresholds increase, i.e., an employer's standards go up as applicants are snapped up by higher ranked employers and competition from these employers has ceased.

For each of the solutions we find, we present numerical results of simulations.

2 Related Work

The secretary problem and its solution first appeared roughly sixty years ago (Flood 1958; Gardner 1966). They are well-known and have been discussed in many literature reviews (Freeman 1983; Ferguson 1989). The optimal strategy and proof of its optimality can be derived in many ways. For instance, it may be derived through a direct probabilistic proof (Ferguson 2012). Additionally, one could use a new algorithm called the Odds algorithm, which more generally computes optimal strategies for a variety of optimal stopping problems (Bruss 2000).

There is an abundance of research on generalizations and extensions of the secretary problem. Optimal strategies have been found for the non-competitive case where a single person wishes to hire one of the top few applicants (Frank and Samuels 1980). Other authors have considered this latter setting as well while limiting the possible strategies to those that use at most two thresholds; these yield a very good approximation to optimal (Dietz, van der Laan, and Ridder 2011). Separately, a competitive environment has been introduced in which multiple identical employers try to hire the best applicant, but the applicants cannot distinguish between the employers (Immorlica, Kleinberg, and Mahdian 2006). So when multiple offers are received, the applicant accepts one at random. Thus, the equilibrium strategies for all employers are the same, but with timing shifted earlier and earlier as the number of employers grows. Another variant has been considered in which only the second-best applicant is desired (Vanderbei 1980). The author gives the exact optimal probability of success for even n and shows that it approaches $1/4$ for large n.

3 The Game

We summarize the game to be studied: There are k employers and n applicants. The n applicants arrive one at a time in a random order. We make two key assumptions:

1. The applicants are of unknown but strictly comparable quality.
2. All applicants rank the employers in the same way, and this ranking is publicly known.

We refer to the j-th ranked employer as employer j.

As each applicant arrives, any employer who has not yet hired anyone can decide to make that applicant an offer. These offers are made simultaneously. If the applicant receives one or more offers, she will choose the employer with the best ranking.

An employer strategy specifies, for each applicant i, given the relative ranking of applicants 1 through i, whether or not to make an offer to that applicant (assuming no previous offer made was accepted). The optimal strategy for an employer depends on the strategies employed by the other employers.

The two games we consider differ in the payoff structure. In both games, we consider only strategies that are in subgame-perfect Nash equilibrium. The key observation that enables us to compute these strategies is that they can be computed inductively. This is because employer j has competition only from employers 1 through $j-1$. Thus, the top-ranked employer does not need to consider competition at all, and simply runs the optimal strategy. Given this strategy, the optimal strategy for the second-ranked employer can be computed, and so on, yielding a set of strategies in equilibrium.

Remark: For brevity, we will sometimes refer to the optimal strategies in subgame-perfect Nash equilibrium simply as "optimal strategies".

4 Hiring Only the Best

This section derives the subgame-perfect Nash equilibrium strategies in the competitive setting in which every employer wants to hire *only the top-ranked applicant*. In other words, the value of the game to an employer is 1 if they hire the top-ranked applicant, and 0 otherwise. A particular set of strategies, one per employer, yields a certain probability of success to that employer, the probability of successfully hiring the best applicant.

It will be convenient, rather than talk about the probability of success, to talk about the probability of *failure* of a class of strategies. This is called the *risk* of a class of strategies.

Solution via Dynamic Programming

Let $R_j(i)$ be the optimal risk, or minimum probability of failure, for employer j among rules that reject the first i applicants, assuming that (a) the $j-1$ higher ranked employers are using their optimal strategies, and, (b) none of these j employers have hired from the first i applicants. This function depends implicitly on n. Given an integer T between 0 and n, define threshold strategy T to be the following: reject the first T applicants and then make an offer to any applicant better than all applicants seen so far. Note that an employer's offer could be rejected in favor of a higher-ranked employer, in which case the employer continues to make offers to "best so far" applicants.

Theorem 1. *For each n and each j between 1 and k, there is a unique number T_j, the* optimal threshold*, such that em-*

ployer j's optimal strategy in equilibrium is to play threshold strategy T_j. Furthermore, T_j is given by

$$T_j = \min\{i : R_j(i) \geq 1 - \frac{i}{n}\} - 1.$$

Additionally, $T_{j-1} \geq T_j$ for all j. (Define $T_0 = n$.)

Proof. We induct on k. The proof of the base case ($k = 1$) is a special case of the proof of the induction step below. Hence, for brevity, we omit it.

Suppose the theorem holds for $1, 2, \ldots, k-1$, and that employers 1 through $k-1$ are playing their optimal strategies. (For $k = 1$, this assumption is vacuous.) We derive player k's best response by backwards induction. Suppose for now that none of the employers has hired any of the first i applicants. By hypothesis $T_{k-1} \leq T_{k-2} \leq \cdots \leq T_1$. For employer k, there are two cases. When $i \leq T_{k-1}$, the employer has the option of making an offer to the i-th applicant if the applicant is ranked best so far. (Obviously, no employer would ever make an offer to an applicant that is not best so far.) In this case we have

$$R_k(i-1) = \frac{1}{i}\min\{R_k(i), 1 - \frac{i}{n}\} + (1 - \frac{1}{i})R_k(i) \quad (1)$$

This is because there is a $1/i$ chance that the i-th applicant is the best so far. If it is, the chance that it is the best overall is i/n, the probability that the best overall is in the first i. So the risk of making an offer is $1 - i/n$. Thus the optimal risk is the minimum of $1 - i/n$ and $R_k(i)$, the risk of rejecting the i-th applicant. If the i-th is not the best so far, the optimal risk is simply $R_k(i)$.

When $i > T_{k-1}$, employer k cannot hire an applicant ranked best so far because a higher ranked employer will also make an offer, and their offer will be successful. In this case we have

$$R_k(i-1) = \frac{1}{i}R_{k-1}(i) + (1 - \frac{1}{i})R_k(i). \quad (2)$$

This is because in the $1/i$ chance that i is the best so far, some higher-ranked employer will hire. Then the optimal risk becomes $R_{k-1}(i)$ because there are only $k-1$ employers looking to hire among employers 1 through k, so any strategy used by employer k after this hire would have an identical outcome to the same strategy used by employer $k-1$ when no one has hired. If the i-th applicant is not best so far, the risk is simply $R_k(i)$.

Let T be given by

$$T = \min\{i : R_k(i) \geq 1 - \frac{i}{n}\} - 1. \quad (3)$$

Note that T exists since $R_k(n) = 1$.

For all $k > 1$, employers k and $k-1$ have access to the same set of strategies and receive identical outcomes for the same strategy, except for situations in which they both make an offer to the best overall applicant and only employer $k-1$'s is accepted. Therefore, $R_k(i) \geq R_{k-1}(i)$, so if $R_{k-1}(i) \geq 1 - i/n$, then $R_k(i) \geq 1 - i/n$. It follows that $T \leq T_{k-1}$. (This fact is trivial for $k = 1$.)

It can be seen from (1, 2) that R_k is nondecreasing, while $1 - i/n$ is decreasing. Therefore, $R_k(i) \geq 1 - i/n$ for all $i > T$, and $R_k(i) < 1 - i/n$ for all $i \leq T$. When the i-th applicant is best so far, the chance of success of hiring is higher than the best chance by rejecting if and only if $i > T$. It follows that there is a unique optimal strategy, namely the threshold strategy $T_k = T$.

We have shown that threshold strategy T is optimal at any index i such that none of employers 1 through k has hired any of the first i applicants. It remains to be shown that this holds when ℓ of these employers have hired, for every $1 \leq \ell < k$. When ℓ employers have hired, there are exactly $k - \ell$ employers looking to hire among employers 1 through k. Then any strategy used by employer k after these hires would have an identical outcome to the same strategy used by employer $k-\ell$ when no one has hired. So it is optimal for employer k to play threshold strategy $T_{k-\ell}$. The most recent hire was made at an index $i \geq T_{k-\ell} \geq T$ since $T_{k-\ell}$ is the first index at which ℓ employers potentially make offers. If employer k was already playing T before these hires, then there is no difference in switching to $T_{k-\ell}$. Therefore, T is optimal.

□

The dynamic program and thresholds can be numerically computed by the following algorithm.

$T_0 \leftarrow n$
for $j = 1, \ldots, k$ **do**
    $R_j(n) \leftarrow 1$
    for $i = n-1, \ldots, 0$ **do**
        if $i \leq T_{j-1}$ **then**
            $R_j(i) \leftarrow$ value from (1)
        else
            $R_j(i) \leftarrow$ value from (2)
        end if
    end for
    $T_j \leftarrow$ value from (3)
end for

For each j, the algorithm takes $\Theta(n)$ time to compute R_j and $O(n)$ time to compute T_j. Therefore, the algorithm takes $\Theta(kn)$ time.

Probability of Success Equals Threshold

Next, we show that, as n approaches infinity, the probability that an employer hires the best applicant approaches the optimal threshold as a fraction of n. The following theorem and proof is analogous to Theorem 1 in a related paper (Immorlica, Kleinberg, and Mahdian 2006).

Theorem 2. *Let T_j be the optimal threshold for the j-th employer in equilibrium, as determined by Theorem 1. Then*

$$\lim_{n\to\infty} \textit{Pr (j hires best)} = t_j := \lim_{n\to\infty} \frac{T_j}{n}.$$

Proof. Fix j. Assume employers 1 through $j-1$ use their optimal thresholds $t_1, \ldots, t_{j-1}$. Let t be the threshold used by employer j, not necessarily t_j. Let $f(t)$ denote the probability that employer j hires the best applicant using threshold t. We show below that there is a constant C such that $f(t) = t\log(1/t) + Ct$ when $t \in (0, t_{j-1}]$. Since $t_j = \arg\max f(t)$ and $t_j \in (0, t_{j-1}]$, we have

$$0 = f'(t_j) = \log(1/t_j) - 1 + C = f(t_j)/t_j - 1$$

so $f(t_j) = t_j$.

We now prove that $f(t) = t\log(1/t) + Ct$ when $t \in (0, t_{j-1}]$. In the discrete setting, the probability that employer j hires the best by the T_{j-1}-th applicant using strategy T is

$$\sum_{i=T+1}^{T_{j-1}} \frac{1}{n}\frac{T}{i-1} = \frac{T}{n}\sum_{i=T+1}^{T_{j-1}} \frac{1}{i-1}$$

which corresponds to the integral

$$\frac{T}{n}\int_{T}^{T_{j-1}} \frac{dx}{x-1}.$$

Using a change of variable $x = nu$, we get

$$\frac{T}{n}\int_{T/n}^{T_{j-1}/n} \frac{dx}{x-1/n}$$

Let $n \to \infty$, $T/n \to t$, and $T_{j-1}/n \to t_{j-1}$. The probability that employer j hires the best before time t_{j-1} using strategy t is

$$t\int_{t}^{t_{j-1}} \frac{dx}{x} = t\log(1/t) - t\log(1/t_{j-1}).$$

We now must compute the probability that employer j hires the best after time t_{j-1}, an event we denote A. Let B denote the event that employer j does not hire before t_{j-1}, which occurs if and only if employer j does not hire between t and t_{j-1}. Since A implies B, we have $\Pr(A) = \Pr(A, B) = \Pr(B)\Pr(A|B)$. Now B occurs if and only if the best applicant before t_{j-1} comes before t, so $\Pr(B) = t/t_{j-1}$. Next, assume B occurs. Then A is independent of t because the information that employer j does not hire between t and t_{j-1} is the same as the information that employer j does not hire before t_{j-1}. That is, assuming B occurs, employer j's strategy appears identical for any $t \leq t_{j-1}$. This implies that $\Pr(A|B)$ is independent of t. Therefore, $\Pr(A) = C't$ for some C' independent of t. Finally, $f(t) = t\log(1/t) + Ct$, where $C = C' - \log(1/t_{j-1})$.

□

Numerical Results – Best Only

Table 1 shows the optimal thresholds for ten employers in subgame-perfect Nash equilibrium with competition when the objective is to hire the best applicant. To obtain the results in the far right column, by adapting our model to a continuous setting, we were able to derive the thresholds for the top employers in the limit as n tends to infinity. For example, the thresholds for the top 4 employers derived in this way are $e^{-1} \approx 0.368$, $e^{-3/2} \approx 0.223$, $e^{-47/24} \approx 0.141$ and $e^{-2761/1152} \approx 0.091$. Note that, by Theorem 2, the numbers in the far right column are also equal to the limiting probability of success.

Employer Rank	Threshold ($n = 10$)	Threshold ($n = 50$)	Threshold ($n \to \infty$)
1	3	18	.368
2	2	11	.223
3	1	7	.141
4	1	4	.091
5	1	3	.059
6	1	2	.039
7	1	1	.026
8	1	1	.017
9	0	1	.012
10	0	1	.008

Table 1: Thresholds for ten employers when hiring best applicant in subgame-perfect Nash equilibrium with competition. The right column shows the values t_j (see Theorem 2).

5 Hiring At or Above Employer Rank

In this section we develop a dynamic program for finding the subgame-perfect Nash equilibrium strategies in the competitive setting in which every employer j wants to hire *one of the top j applicants*. In other words, the value of the game to employer j is 1 if they hire one of the top j-ranked applicants, and 0 otherwise.

As we shall see in detail in the next section, the optimal equilibrium strategies in this setting are quite a bit more complicated. In particular, the strategy for an employer at time t depends on precisely which of the higher ranked employers have already hired at that time. Given this information, the optimal strategy specifies a set of j threshold times at which this employer becomes less and less selective: after the first threshold time, he will only hire the best so far, after the second, he will only hire one of the top two so far and so on.

Numerical Results – Top j

We begin with some numerical results about the optimal strategies in equilibrium. In the next section, we explain the derivation of these strategies.

Hiring Status	Employer Rank		
	1	2	3
No hires	.368	.246 .559	.189 .413 .635
Emp 2 has hired	.368		.258 .507 .727
Emp 1 has hired		.347 .667	.239 .475 .677
Emps 1 and 2 have hired			.337 .587 .775

Table 2: Thresholds for three employers when hiring at worst own rank in subgame-perfect Nash equilibrium with competition.

Table 2 shows the optimal thresholds for three employers in the subgame-perfect Nash equilibrium with competition as fractions of n as n approaches infinity, when employer j tries to hire one of the best j applicants. Up to three values are listed in each box. The first corresponds to the threshold for hiring the relative best, the second to hiring the second relative best, and the third to hiring the third relative best.

For example, consider the thresholds for the rank 3 employer when no one has hired. After $.189n$ applicants have passed, the rank 3 employer should make an offer to any new applicant that is the best so far. Similarly, after $.413n$ applicants, he should also offer to the second best so far. After $.635n$, he should also offer to the third best so far. However, this strategy may need to change if a hire is made by another employer. For instance, if the rank 2 employer hires someone, then these thresholds should be abandoned, and instead the rank 3 employer should use $.258n$, $.507n$, and $.727n$.

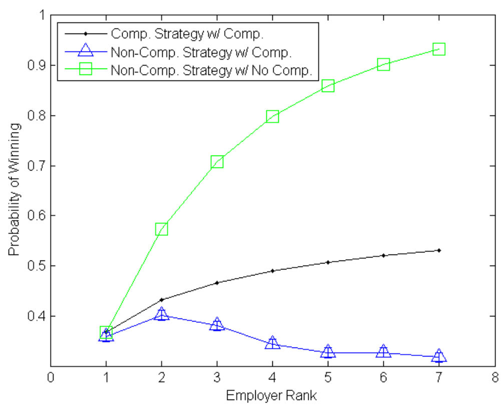

Figure 1: Probabilities of success for each employer using different strategies in various settings.

Figure 1 shows, for each employer in ranks 1 through 7, the probability of hiring an applicant of same or better rank for three different strategies: The highest curve is the probability of success using the optimal strategy in the absence of any competition. The second highest curve is the probability of success using the subgame-perfect Nash equilibrium with competition. The lowest curve shows the simulated probability of success using the optimal strategy that ignores competition, but in the presence of competition, for very large n. This figure shows very clearly how important it is to take the competition into account.

Table 3 shows the simulated distributions of the true ranks of hired applicants when all employers use the optimal equilibrium strategies, for very large n. For example, the rank 2 employer has a 19.1% chance to hire the best applicant, 22.6% chance to hire the second best applicant, 28.6% chance to hire the third best or worse applicant, and a 29.4% chance to hire no one. It is interesting to contrast this to employer 2's outcome when he attempts to hire only the best, that is, "move up in the world". As we saw earlier, in this latter case, his probability of hiring the best applicant is higher, in fact 22.3%, but this comes at the expense of being less likely to hire the second best, at 9.9%, and being more likely to hire nobody, at 44.2% (probabilities not shown in table).

Hired Rank	Employer Rank					
	1	2	3	4	5	6
1	38.1	19.1	11.8	7.4	4.1	2.6
2		22.9	16.8	12.1	8.0	5.6
3			18.3	14.9	10.9	7.8
4				14.5	14.4	10.3
5					13.5	12.7
6						12.4
$>$ own	25.5	28.6	28.1	28.5	29.7	29.8
none	36.4	29.4	25.1	22.7	19.3	18.6

Table 3: Simulated distribution of hired applicants when hiring at worst own rank in subgame-perfect Nash equilibrium with competition.

The Dynamic Program

Notation and Definitions: Given k employers and n applicants, at any point in time in the game, let x_j be the indicator variable for whether employer j has hired an applicant. Let $x = (x_1, \ldots, x_k)$ be known as the hiring status. The optimal strategy for each employer depends on x, meaning it may change with time if another employer makes a new hire.

We now define several functions that all depend implicitly on n. Let the optimal risk $R_j(i, x)$ be the minimum probability that employer j fails to get one of the top j applicants among all strategies that reject the first i applicants, given x when the i-th applicant arrives. We say R_j is defined only when $x_j = 0$. Let r_i be the rank of the i-th applicant, and let rr_i be the relative rank of the i-th applicant, meaning its rank relative to the first i applicants. As in the previous section, our derivation of the optimal strategy for j assumes that employers 1 through $j-1$ are playing their optimal equilibrium strategies.

There are three auxiliary functions. Let $HRR(i, j, x)$ be the highest relative rank of applicant i that would be hired by one of employers 1 through j who has not hired yet under x, according to optimal strategies. Let HRR equal 0 if there is no such applicant. Let $AE(i, t, j, x)$ (accepted employer) be the rank of the employer who would offer to applicant i with relative rank t and be accepted, who has not hired yet under x, and whose employer rank is j at most. Let $NX(i, t, j, x)$ (new x) be a copy of x but with a 1 at index $AE(i, t, j, x)$. Then $NX(i, t, j, x)$ is the new hiring status after applicant i is hired. Note that $t \leq HRR(i, j, x)$ implies that $AE(i, t, j, x)$ and $NX(i, t, j, x)$ exist.

Solution: Our aim is to compute $R_j(0, \mathbf{0})$, the optimal initial probability of success for employer j. We start by computing the probability that the i-th applicant has worse rank than c conditioned on its relative rank. This probability follows the cumulative hypergeometric distribution,

$$\Pr(r_i > c | rr_i = t) = \sum_{m=0}^{t} \binom{c}{m} \binom{n-c}{i-m} \Big/ \binom{n}{i}$$

because we sum the probabilities of all ways in which the first i applicants contain $m \leq t$ of the top c applicants. This quantity is nonincreasing in i.

The next lemma describes a dynamic program to compute R_j. The program must be computed with i decreasing from n to 0 and with the number of 1's in the first j indices of x decreasing from j to 0. Additionally, the R_j's must be computed with increasing j.

Lemma 3. *Let $s = HRR(i, j-1, x)$. We have*

$$R_j(i-1, x) = \frac{1}{i}\left\{ \sum_{t=1}^{s} R_j(i, NX(i,t,j-1,x)) + \sum_{t=s+1}^{i} \min\{R_j(i,x), \Pr(r_i > j | rr_i = t)\} \right\}$$

with initial condition $R_j(n, x) = 1$.

Proof. The optimal risk when no applicants are given offers is 1, so $R_j(n, x) = 1$.

We condition on the relative rank t of applicant i. Now t is uniformly distributed between 1 and i, so we get the initial $1/i$ factor. Next we explain the first sum. We consider all t that would be hired by an employer ranked better than j. This is all $t \leq HRR(i, j-1, x)$ for the following reason. By definition $t = HRR(i, j-1, x)$ would be hired by a better employer than j. Then any lesser t would also be hired because if an employer is hiring relative rank t, it clearly is also hiring better relative ranks than t. For these t that would be hired by a better employer than j, the minimum risk is $R_j(i, NX(i, t, j-1, x))$. Clearly we must move from $i-1$ to i. We also move to status $NX(i, t, x)$ from x because x has gained a 1 somewhere to represent another employer making a hire.

For the remaining t (second sum), the relative ranks such that no better employers than j would make an offer, employer j has the choice of rejecting or hiring. If he rejects, his risk is clearly $R_j(i, x)$ because he moves from $i-1$ to i and the hiring status is still x. If he hires, his risk is $\Pr(r_i > j | rr_i = t)$, the probability that the true rank of i is worse than j given its relative rank of t. So his optimal risk is the minimum of these choices. □

Lemma 4. *R_j is nondecreasing in i.*

Proof. Fix x, n, and j, and i. When $rr_i = t > HRR(i, j-1, x)$, we have $\min\{R_j(i, x), \Pr(r_i > j | rr_i = t)\} \leq R_j(i, x)$. When $rr_i = t \leq HRR(i, j-1, x)$, we claim that $R_j(i, NX(i, t, j-1, x)) \leq R_j(i, x)$. The hiring status for the left term is $x' = NX(i, t, j-1, x)$. Its only difference from the hiring status x for the right term is that exactly one additional employer has made a hire in x'. In other words, the employer faces strictly less competition when the hiring status is x' than x, in the sense that fewer of its offers would be turned down in favor of offers from better-ranked employers. Therefore, the optimal strategy for x cannot do better than the optimal strategy for x'. Since R_j represents the chance that an optimal strategy loses, it must be that $R_j(i, x') \leq R_j(i, x)$. Then every term in the formula from Lemma 3 for $R_j(i-1, x)$ is at most $R_j(i, x)$. □

These lemmas imply the optimal strategy.

Theorem 5. *Given n applicants, a hiring status x, an employer rank j, and an applicant relative rank $t \leq j$, there is an integer $T_{jt}(x)$ such that it is optimal for employer j to make an offer to an applicant of relative rank t if and only if the applicant arrives after position $T_{jt}(x)$.*

Proof. It is optimal for employer j to offer to the i-th applicant with $rr_i = t$ if and only if $R_j(i, x) \geq \Pr(r_i > j | rr_i = t)$, meaning when the risk $R_j(i, x)$ of rejecting the applicant is greater than or equal to the risk $\Pr(r_i > j | rr_i = t)$ of hiring. Now R_j is nondecreasing in i and $\Pr(r_i > j | rr_i = t)$ is nonincreasing in i. Thus if this condition is satisfied for some i', then it is satisfied for all $i > i'$. Then we are done by letting $T_{jt}(x)$ be the position preceding the first such i,

$$T_{jt}(x) = \min\{i : R_j(i, x) \geq \Pr(r_i > j | rr_i = t)\} - 1.$$

□

Computing the Auxiliary Functions:

Lemma 6. *$HRR(i, j, x) = \max_{t,\ell} t$ such that $T_{\ell t}(x) < i$, $1 \leq t \leq \ell \leq j$ and $x_\ell = 0$.*

Proof. Interpret t as the relative rank of applicant i and ℓ as the rank of an employer. Constraint 1 says that i must be after threshold $T_{\ell t}(x)$ so that employer ℓ is seeking to hire people of relative rank t. Constraint 2 says that the relative rank t is at most ℓ because employer ℓ only ever wants relative ranks better than ℓ. Constraint 3 says that employer ℓ has not hired. □

Lemma 7. *$AE(i, t, j, x) = \min_\ell \ell$ such that $T_{\ell t}(x) < i$, $t \leq \ell \leq j$ and $x_\ell = 0$.*

Proof. The argument is similar to the previous lemma. □

Unfortunately, we have no closed-form description of these strategies, and the complexity of solving for the j-th highest ranked employer strategy is $O(2^j n)$, since to be a best response, he must change his strategy depending precisely on which subset of higher ranked employers have already hired someone. It is an interesting open question whether there are other equilibria that are simpler.

6 Conclusion

We examined two extensions of the secretary problem that added competition between employers to hire the best secretaries. Optimal strategies to both problems were found via dynamic programming. In the optimal strategies, lower-ranked employers started hiring earlier, while higher-ranked employers had more freedom to choose and waited longer.

Numerous interesting open questions are introduced by altering or extending the model considered here: For example, what if the applicants do not disappear forever, but rather are strategic in their decision-making, holding out for future better offers? What if employers can specify deadlines for the acceptance of an offer? What happens when salaries are introduced? What do the equilibria of these more realistic games look like?

References

Bruss, F. T. 2000. Sum the odds to one and stop. *Annals of Probability* 1384–1391.

Dietz, C.; van der Laan, D.; and Ridder, A. 2011. Approximate results for a generalized secretary problem. *Probability in the Engineering and Informational Sciences* 25(2):157–169.

Ferguson, T. S. 1989. Who solved the secretary problem? *Statistical Science* 4(3):282–296.

Ferguson, T. S. 2012. *Optimal stopping and applications*.

Flood, M. R. 1958. Untitled. Letter, a copy of which can be found in the Martin Gardner papers at Stanford University Archives, series 1, box 5, folder 19.

Frank, A. Q., and Samuels, S. M. 1980. On an optimal stopping problem of Gusein-Zade. *Stochastic Processes and their Applications* 10(3):299–311.

Freeman, P. R. 1983. The secretary problem and its extensions: A review. *International Statistical Review/Revue Internationale de Statistique* 189–206.

Gardner, M. 1966. *New Mathematical Diversions from Scientific American*. Simon and Schuster.

Immorlica, N.; Kleinberg, R.; and Mahdian, M. 2006. Secretary problems with competing employers. *Internet and Network Economics* 389–400.

Vanderbei, R. J. 1980. The optimal choice of a subset of a population. *Mathematics of Operations Research* 5(4):481–486.