

On a Competitive Secretary Problem

ERIC LEI, University of Washington

ANNA R. KARLIN, University of Washington

The secretary problem is a famous model for matching an employer with an applicant, in which an employer sees a stream of applicants and tries to hire the best. Here we consider an extension in which there are multiple employers seeing the same stream of applicants. Different employers are favored by the applicants, leading to competition between employers to hire higher-ranked applicants. We derive optimal strategies and related results for this extension, as well as give numerical results from simulations.

1. INTRODUCTION

The classical secretary problem [Ferguson 2012] is one of the most basic models for matching an employer with an employee: An employer wishes to hire a single employee out of a pool of n , and wants this employee to be the best in the pool. The employees arrive in a random order, and a decision as to whether to make an offer is irrevocable. If the goal of the employer is to maximize the probability of hiring the best in the bunch, he should wait until he has seen $1/e$ of the applicants, and then hire the first one better than the best he has seen so far.

Now consider a scenario in which there are multiple employers competing for the potential employees. We explore two questions in this paper: (1) how does the competition between the employers affect their hiring strategies? (2) How well do the employers and applicants do in the presence of competition? We explore this question in the context of a game defined as follows:

There are k totally ranked employers and n totally ranked 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; there is no possibility of making her an offer later. Since the employers are totally ranked (imagine for example that the employers are computer science departments and the applicants are faculty candidates), an applicant that receives multiple offers will accept the offer from the highest ranked employer among those making her an offer. Once an employer has hired an applicant, he can make no further offers. For most of this paper, we assume that the payoff to the j -th ranked employer is 1 if he successfully hires one of the top j applicants.

In Section 4, we use dynamic programming to find a subgame-perfect Nash equilibrium in this game. The computation of these equilibrium strategies is aided by the fact that they can be computed inductively in order of decreasing rank of the employer. (The applicants have no strategy – they simply take the best offer they receive.) Indeed, the strategy for the top ranked employer is the optimal strategy in the secretary problem: he is unaffected by the competition, since any offer he makes will be accepted. Thus, since his goal is to maximize the probability of hiring the best secretary, will run the usual secretary algorithm. The second-ranked employer, must, however, take the first-ranked employer's optimal strategy into account. However, he does not need to worry about competition from lower ranked employers. And so on.

The structure of the strategies used, not surprisingly, has the following general structure: Consider, say the i -th ranked employer. Then there is a set of numbers $1 > \alpha_1 > \alpha_2 > \dots > \alpha_i \geq 0$ such that employer i will attempt to make an offer to the j -th best so far, after $\alpha_j n$ 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.

Unfortunately, there is no closed form description of these strategies and the complexity of solving for the j -th highest ranked employer strategy is $\Omega(2^j n)$, 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.

In Section 5, we consider the setting in which each employer receives a payoff of 1 if he hires the top-ranked applicant, and 0 otherwise. We simplify our dynamic program to compute the optimal strategies and probabilities in polynomial time.

In Section 6, we describe the results of simulations with the optimal strategies.

2. RELATED WORK

The classical secretary problem and its solution are well-known and may be found in literature reviews such as [Freeman 1983]. The optimal strategy and proof of its optimality can be derived in many ways. In [Ferguson 2012], it is derived by a direct probabilistic proof. In [Bruss 2000], it is derived through a new algorithm called the Odds algorithm that more generally computes optimal strategies for a variety of optimal stopping problems.

There is also an abundance of work on this problem and its extensions. [Frank 1980] considers a problem very close to ours in which a single person wishes to hire one of the top few applicants. It uses a dynamic program to compute the optimal strategies. [Dietz 2011] considers this setting as well, but it limits the possible strategies to those that are simple in that they use at most two thresholds. It finds that this class of strategies is an extremely accurate approximation to the optimal strategies. In [Immorlica 2006], a competitive environment is introduced in which multiple identical employers try to hire the best applicant, and a applicant with multiple offers chooses randomly. It finds that the optimal strategy shifts the timing of the earliest offer further back as the number of employers grows.

3. THE GAME

We analyze the following game: There are k employers and n employees (secretaries). The n employees arrive one at a time. As each employee arrives, any employer who has not yet employed anyone, can decide to make that employee an offer. These offers are made simultaneously. If she receives one or more offers, the employee will choose her favorite. The strategic decision for the employers is when to make an offer.

We make two key assumptions:

- (1) The employees are of unknown but strictly comparable quality, and arrive in a random order.
- (2) All employees have the same ranking on employers and this ranking is publicly known.

4. THE OPTIMAL STRATEGY

This section derives the optimal strategy in the competitive setting in which the j -th ranked employer wants to hire from the top j ranked applicants.

4.1. 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 $x = (x_1, \dots, x_k)$. Now x describes who has hired and who has not. The optimal strategy for each em-

ployer depends on x , meaning it may change with time if another employer makes a new hire.

Let $R_j(i, x; n)$ be the minimum probability that Employer j fails to get one of the top j applicants among all strategies that do not attempt to hire until more than i applicants have arrived and that are valid under x . A strategy for Employer j is valid under x if, assuming all other employers act optimally, Employer j never has the potential to make an offer to a candidate to which a better-ranked employer who has not hired according to x would also make an offer. Also, we say R_j is defined only when $x_j = 0$. Now $R_j(i, x; n)$ represents the risk of rejecting the i -th applicant.

Let r_i rank of the i -th applicant, and let rr_i be the relative rank of the i -th applicant, meaning its rank among the first i applicants that appear.

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

4.2. Backwards Induction

Our aim is to compute $R_j(0, 0; n)$, the optimal initial probability of failure 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,

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

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

The next lemma describes a dynamic program to compute R_j . An interpretation of the program is that it computes the optimal risk R_j at applicant i in terms of the optimal risk at $i + 1$. When the i -th applicant would accept Employer j 's offer, then the program computes whether there is a higher probability of success in making an offer or rejecting the applicant. When the i -th applicant would receive and accept a different offer, the program uses the risk associated with the new hiring status. 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, R_j 's must be computed with increasing j .

LEMMA 4.1. *We have*

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

with initial condition

$$R_j(n, x; n) = 1.$$

PROOF. The optimal risk when no applicants are accepted is 1, so $R_j(n, x; n) = 1$.

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 $t \leq HRR(i, j - 1, x; n)$ for the following reason. By definition $t = HRR(i, j - 1, x; n)$ 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 $R_j(i, NX(i, t, j - 1, x; n); n)$. Clearly we must move from $i - 1$ to i . We also move to status $NX(i, t, x; n)$ from x because x has gained a 1 somewhere to represent another employer making a hire.

Next we explain the second sum. For the remaining t , 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 constraint is still x . If he accepts, 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. \square

Note that x is a binary vector of length k and there are n applicants, so computing R_j takes time $\Omega(2^k n)$.

The next lemma is important to proving the form of the optimal strategy.

LEMMA 4.2. R_j is nondecreasing in i .

PROOF. Fix x , n , and j , and i . For any $rr_i = t > HRR(i, j - 1, x; n)$, we have $\min(R_j(i, x; n), \Pr(r_i > j | rr_i = t; n)) < R_j(i, x; n)$. For any $rr_i = t \leq HRR(i, j - 1, x; n)$, we claim that

$$R_j(i, NX(i, t, j - 1, x; n); n) \leq R_j(i, x; n).$$

The hiring status for the left term is $x' = NX(i, t, j - 1, x; n)$. 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. The available strategies for x are a subset of those for x' . 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 $R_j(i, x'; n) \leq R_j(i, x; n)$.

Then every term in the expression from Lemma 4.1 for $R_j(i - 1, x; n)$ is at most $R_j(i, x; n)$, which implies that R_j is nondecreasing in i . \square

These lemmas imply the optimal strategy.

THEOREM 4.3. Given n applicants, a hiring status x , an employer rank j , and an applicant relative rank $t \leq j$, there is a number $T_{jt}(x)$ such that it is optimal for Employer j to accept an applicant of relative rank t if and only if the applicant arrives at position $T_{jt}(x)n$ or later.

PROOF. It is optimal for Employer j to offer to the i -th applicant with $rr_i = t$ if and only if $R_j(i, x; n) \geq \Pr(r_i > j | rr_i = t; n)$, meaning when the risk $R_j(i, x; n)$ of rejecting the applicant is at greater than or equal to the risk $\Pr(r_i > j | rr_i = t; n)$ of accepting. Now R_j is nondecreasing in i and $\Pr(r_i > j | rr_i = t; n)$ 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 $T_{jt}(x)$ correspond to the first such i :

$$T_{jt}(x) = \min_i i/n \quad \text{s.t.} \quad R_j(i, x; n) \geq \Pr(r_i > j | rr_i = t).$$

□

4.3. Computing the Auxiliary Functions

LEMMA 4.4. *We can compute HRR as*

$$HRR(i, j, x; n) = \max_{t, \ell} t$$

$$\text{s.t. } \tau_{\ell t}(x)n \leq i, \quad 1 \leq t \leq \ell \leq j, \quad x_\ell = 0.$$

PROOF. Interpret t as the relative rank of applicant i and ℓ as the index of the employer. Constraint 1 says that i must be after threshold $\tau_{\ell t}(x)$, the threshold for employer ℓ to take relative rank t . Employer ℓ must be seeking to hire people of relative rank t . Constraint 2 says that the relative rank t is at most the employer index ℓ , which is at most j . Employer ℓ only ever wants relative ranks better than ℓ . Constraint 3 says that employer ℓ has not hired. □

LEMMA 4.5. *We can compute AE as*

$$AE(i, t, j, x; n) = \min_{\ell} \ell$$

$$\text{s.t. } \tau_{\ell t}(x)n \leq i, \quad t \leq \ell \leq j, \quad x_\ell = 0.$$

PROOF. Interpret ℓ as the index of the employer. Constraint 1 says that i must be after threshold $\tau_{\ell t}(x)$, the threshold for employer ℓ to take relative rank t . Employer ℓ must be seeking to hire people of relative rank t . Constraint 2 says that the employer index ℓ is at least t and at most j . Employer ℓ only ever wants relative ranks better than ℓ . Constraint 3 says that employer ℓ has not hired. □

4.4. Different Employer Objective

Instead of hiring one's own rank or better, the results of this section can be generalized to the setting where Employer j wants to hire one of the best c_j applicants for some constant c_j . It is not difficult to show that Theorem 4.3 holds in this case.

5. HIRING ONLY THE BEST

5.1. Optimal Strategy

The previous section can be adapted to the special case where each employer only wants the best applicant. Because this special case is much simpler, the dynamic program can be simplified to an $O(kn)$ dynamic program.

Let T_i be the optimal threshold of E_i . Let $R_j(i)$ be the optimal risk for E_j among rules that reject the first i applicants assuming that no employers have hired from the first i applicants. Then $R_j(n) = 1$ for all j . Also

$$R_1(i-1) = \frac{1}{i} \min\{R_1(i), 1 - \frac{i}{n}\} + (1 - \frac{1}{i})R_1(i).$$

There is a $1/i$ chance that the i th 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 accepting is $1 - 1/i$. Thus the optimal risk is the minimum of the risk of accepting and the risk of rejecting. If the i th is not the best so far, the optimal risk is simply $R_1(i)$.

The threshold T_1 is one before the first index where the risk of accepting is less than the risk of rejecting. We subtract one because this index may be hired, but everything should be rejected up to the threshold,

$$T_1 = \min\{i-1 : R_1(i) \geq 1 - \frac{i}{n}\}.$$

Now T_1 exists since at $i = n$, $R_1(n) \geq 0$.

For $j > 1$, we have two cases depending on whether $i \leq T_{j-1}$. If so, then E_j will be able to hire a best so far. Otherwise, E_j cannot hire because a higher ranked employer will hire instead. When $i \leq T_{j-1}$, we have

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

for the same reasons as when $j = 1$. When $i > T_{j-1}$, we have

$$R_j(i-1) = \frac{1}{i} R_{j-1}(i) + (1 - \frac{1}{i})R_j(i).$$

In the $1/i$ chance that i is the best so far, the top ranked applicant will hire. Then the optimal risk becomes $R_{j-1}(i)$ because there are only $j-1$ employers looking to hire, so any strategy used by E_j after this hire would have an identical outcome to the same strategy used by E_{j-1} when no one has hired. If i is not best so far, the risk is simply $R_j(i)$.

In general, the threshold T_j is one before the first index where the risk of accepting is less than the risk of rejecting and where the index is no farther than T_{j-1} ,

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

Now T_j exists since at $i = n$, $R_j(n) \geq 0$.

5.2. Probability of Success Equals Threshold

As n approaches infinity, we show that the probability that an employer hires the best applicant equals their threshold as a fraction of n . We model this limit as a continuous setting. The employers see applicants in a period of time $[0, 1]$. There is an applicant associated with each value in this set. At any time, the employers know the relative rankings of all previous applicants, as before. For $t \in [0, 1]$, (continuous) threshold rule t is the strategy of rejecting all applicants before time t and taking the next one better than those seen so far. This is analogous to the discrete setting.

The following theorem and proof is based on Theorem 1 in [Immorlica 2006].

THEOREM 5.1. *When each E_i uses optimal threshold t_i , E_i hires the best applicant with probability t_i .*

PROOF. Fix i . Assume E_1, \dots, E_{i-1} use optimal thresholds t_1, \dots, t_{i-1} . Let t be the threshold used by E_i , not necessarily the optimal t_i . Let $f(t)$ denote the probability that E_i hires the best applicant using threshold t . We show that there exists a constant C such that $f(t) = t \log(1/t) + Ct$ when $t \in (0, t_{i-1}]$. Since $t_i = \arg \max f(t)$ and $t_i \in (0, t_{i-1}]$, we have

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

so $f(t_i) = t_i$.

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

$$\sum_{j=t+1}^{t_{i-1}} \frac{1}{n} \frac{t}{j-1} = \frac{t}{n} \sum_{j=t+1}^{t_{i-1}} \frac{1}{j-1}$$

which corresponds to the integral

$$\frac{t}{n} \int_t^{t_{i-1}} \frac{dx}{x-1}.$$

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

$$\frac{t}{n} \int_{t/n}^{t_{i-1}/n} \frac{ndu}{nu - 1} = \frac{t}{n} \int_{t/n}^{t_{i-1}/n} \frac{dx}{x - 1/n}$$

Let $n \rightarrow \infty$, and reinterpret t and t_i as values in $[0, 1]$. Then in the continuous setting, the probability that E_i hires the best before time t_{i-1} using strategy t is

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

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

□

6. EXPERIMENTAL RESULTS

This section describes the numerical values of the optimal strategies as well as other relevant simulations from when each employer tries to hire an applicant of at least their own rank. Unless otherwise stated, all simulations were performed with $n = 3,000$ and 3,000 iterations where applicable. All error bars correspond to 95% confidence intervals. When there are no error bars, the values were computed precisely using a dynamic program.

Table I. Probability of Hiring an Applicant

Employer Rank	1	2	3	4	5	6	7
Probability of hiring: using sub-game perfect equilibrium strategies	.368	.432	.466	.490	.501	.520	.531
Probability of hiring: optimal non-competitive strategy in competitive setting	.359	.403	.381	.343	.326	.326	.317
Probability of hiring: optimal non-competitive strategy without competition	.368	.574	.708	.799	.860	.903	.932

Table I and Figure 1 show, for each employer with ranks 1-7, their expected payoff in the game, i.e. for the employer of rank j , the probability of successfully hiring one of the top j candidates. The first row is when all employers use subgame-perfect strategies. For comparison, the second and third rows show these probabilities when the j -th ranked employer uses the optimal standalone strategy for hiring one of the top j applicants, ignoring the fact that there is competition. The third row shows the performance of this strategy when there is no competition, and the second row shows the performance of this strategy when there is competition. For example, the rank 2 employer's optimal competitive strategy has a 43.2% chance of hiring the first or second best applicant in the competitive setting. Using the optimal non-competitive strategy, he only has a 39.6% chance. Using the optimal non-competitive strategy in the non-competitive setting, he has a 57.4% chance.

Table II and Figure 2 show the mean position of hired applicants for each employer when Employer j tries to hire at worst the j -th ranked applicant, using different strategies to in competitive and non-competitive settings. The optimal strategy for the competitive setting is used in the competitive setting. The optimal strategy for the

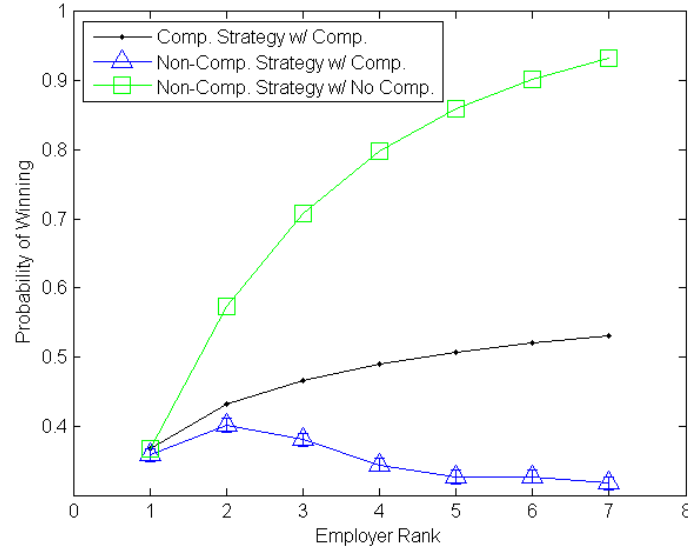


Fig. 1. Table I as a graph. Optimal probability of making a desired hire vs. employer rank.

Table II. Position of Hired Applicant

Employer Rank	1	2	3	4	5	6	7
Mean Position of Hired Applicant with Competitive Strategy in Competitive Setting	.582	.520	.515	.508	.513	.513	.516
Mean Position of Hired Applicant with Non-Competitive Strategy in Competitive Setting	.581	.705	.750	.765	.767	.770	.772
Mean Position of Hired Applicant with Non-Competitive Strategy in Non-Competitive Setting	.581	.599	.599	.593	.583	.575	.567

non-competitive setting is used in both settings. The mean is over instances in which an employer makes a hire, so it excludes instances in which the employer hires no one. The values given are positions as a fraction of n . For example, the rank 2 employer's optimal competitive strategy hires an applicant at position $.520n$ on average in the competitive setting, given that the employer makes a hire.

Table III. Hiring Thresholds

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

These numbers show the hiring thresholds in the optimal strategy for when the j -th ranked employer hires at worst the j -th ranked applicant in the competitive setting.

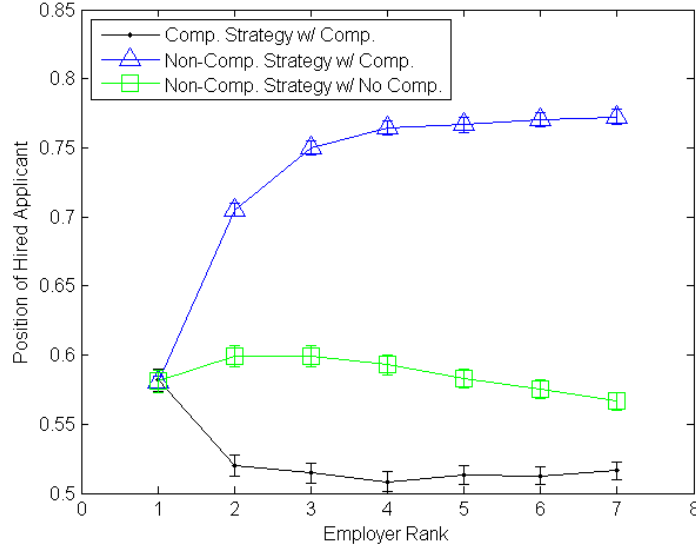


Fig. 2. Table II as a graph. Mean position of hired applicant vs. employer rank.

A set of thresholds is given for each possible state of hiring, assuming Employer 3 has not hired. The values given should be multiplied by n to determine the index to use as thresholds. 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.

Consider, for example, 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 $.727n$.

Table IV. Distribution of Hired Applicants Using Optimal Competitive Strategy in Competitive Setting

Hired Rank	Employer Rank						
	1	2	3	4	5	6	7
1	.381	.191	.118	.074	.041	.026	.026
2		.229	.168	.121	.080	.056	.031
3			.183	.149	.109	.078	.057
4				.145	.144	.103	.089
5					.135	.127	.093
6						.124	.117
7							.121
> own rank	.255	.286	.281	.285	.297	.298	.282
none	.364	.294	.251	.227	.193	.186	.183

Table IV and Figure 3 show the distributions of the true ranks of hired applicants when Employer j uses the optimal strategy to hire at worst the rank j applicant in the competitive setting. For example, the rank 2 employer has a 19.1% chance to hire the

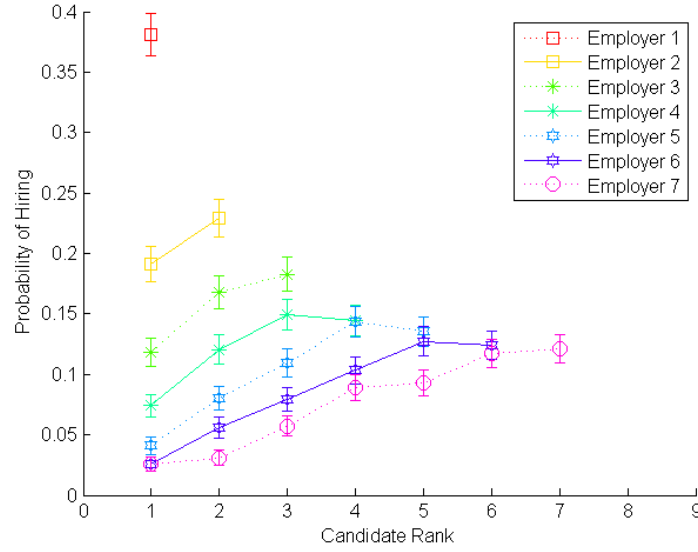


Fig. 3. Table IV as a graph. Probability of hiring vs. rank of hired applicant. The probabilities for each employer are plotted as a separate series.

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.

Table V. Distribution of Hired Applicants Using Optimal Non-Competitive Strategy in Competitive Setting

Hired Rank	Employer Rank						
	1	2	3	4	5	6	7
1	36.3	17.3	8.0	2.8	1.6	1.1	0.7
2		22.3	12.0	6.7	3.2	1.9	1.4
3			17.3	10.6	6.4	3.7	2.5
4				14.4	9.6	5.8	4.3
5					11.7	9.1	6.3
6						10.6	7.8
7							10.1
> own rank	26.9	13.8	10.5	9.1	8.0	7.0	5.3
none	36.8	46.6	52.3	56.5	59.6	61.0	61.6

Table V and Figure 4 show the distributions of the true ranks of hired applicants in the competitive setting when Employer j uses the optimal non-competitive strategy to hire at worst the rank j applicant. For example, the rank 2 employer has a 17.3% chance to hire the best applicant, 22.3% chance to hire the second best applicant, 13.8% chance to hire the third best or worse applicant, and 46.6% chance to hire no one.

Table VI and Figure 5 show the distributions of the true ranks of hired applicants in the non-competitive setting when Employer j uses the optimal non-competitive strategy to hire at worst the rank j applicant. For example, the rank 2 employer has a 32.1% chance to hire the best applicant, 23.2% chance to hire the second best applicant, 20.8% chance to hire the third best or worse applicant, and 23.9% chance to hire no one.

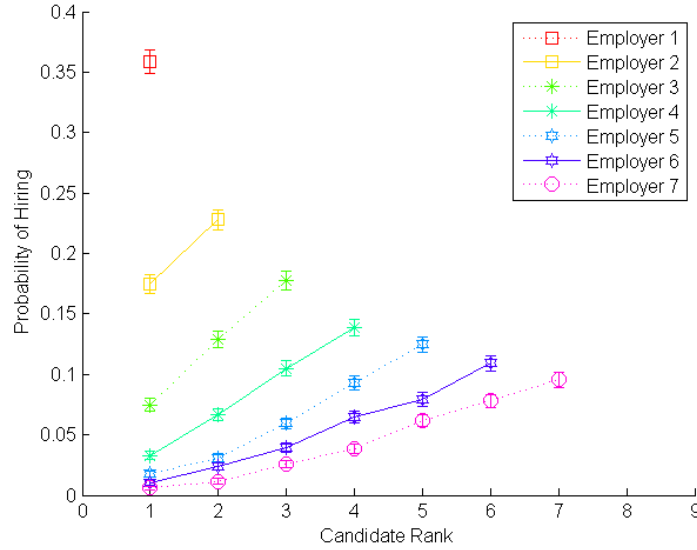


Fig. 4. Table V as a graph. Probability of hiring vs. rank of hired applicant. The probabilities for each employer are plotted as a separate series.

Table VI. Distribution of Hired Applicants Using Optimal Non-Competitive Strategy in Non-Competitive Setting

Hired Rank	Employer Rank						
	1	2	3	4	5	6	7
1	33.5	32.1	30.6	29.0	27.7	26.3	25.3
2		23.2	23.8	23.2	22.4	21.8	20.9
3			15.8	16.5	17.0	17.2	17.2
4				11.0	11.7	12.1	12.4
5					6.7	7.4	7.9
6						4.7	5.1
7							3.5
> own rank	27.7	20.8	14.2	9.4	6.9	5.3	4.0
none	38.8	23.9	15.7	10.9	7.6	5.3	3.6

7. CONCLUSION

We described an extension of the secretary problem in which multiple employers compete to hire the top-ranked secretaries. We derived optimal strategies for when the employers wanted to hire someone of their own rank or better, and for when the employers all wanted to hire the best secretary. Our main tool was a dynamic program whose numerical values were computed and shown. Also, we gave statistics from simulations of this model.

Numerous open questions present themselves immediately:

- (1) What is the optimal strategy for other employer objectives?
- (2) What if the applicants do not disappear forever, but rather are strategic in their decision-making, holding out for future better offers?
- (3) Should the employers introduce deadlines for acceptance of offers (as opposed to requiring immediate decisions), and if so, what is the right strategy for an employer as a function of his rank? What are the equilibria in the resulting games? How

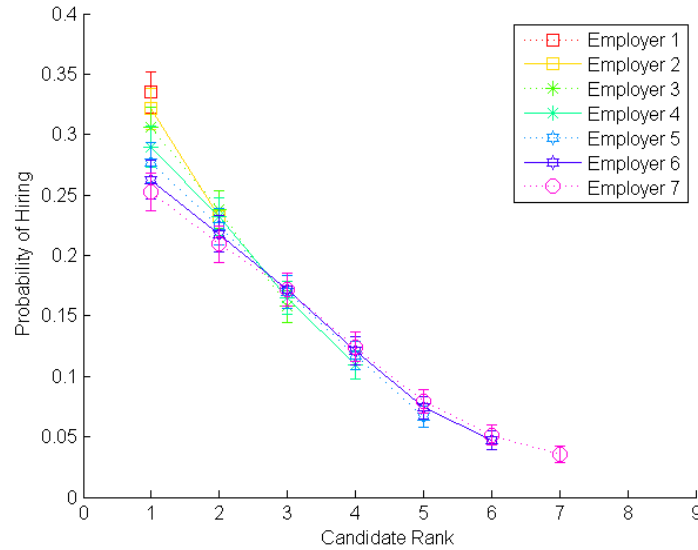


Fig. 5. Table VI as a graph. Probability of hiring vs. rank of hired applicant. The probabilities for each employer are plotted as a separate series.

well does the k -th ranked employer do in this equilibrium? How far is the resulting matching from being stable (in the sense of stable marriage)?

- (4) What happens when salaries are introduced?
- (5) Do typical hiring practices indeed represent an equilibrium, and how close to socially optimal they are (for an appropriate definition of social welfare)?

REFERENCES

- Bruss, F. Thomas. "Sum the odds to one and stop." *Annals of Probability* (2000): 1384-1391.
- Dietz, Chris, Dinard van der Laan, and Ad Ridder. "Approximate results for a generalized secretary problem." *Probability in the Engineering and Informational Sciences* 25.02 (2011): 157-169.
- Ferguson, Thomas S. "Optimal stopping and applications." (2012).
- Frank, Arthur Q., and Stephen M. Samuels. "On an optimal stopping problem of Gusein-Zade." *Stochastic Processes and their Applications* 10.3 (1980): 299-311.
- Freeman, P. R. "The secretary problem and its extensions: A review." *International Statistical Review/Revue Internationale de Statistique* (1983): 189-206.
- Immorlica, Nicole, Robert Kleinberg, and Mohammad Mahdian. "Secretary problems with competing employers." *Internet and Network Economics*. Springer Berlin Heidelberg, 2006. 389-400.