Think Eternally: Improved Algorithms for the Temp Secretary Problem and Extensions

Thomas Kesselheim[1] Andreas Tönnis[2]

[1]Department of Computer Science - TU Dortmund, Germany

[2]Department of Computer Science - University of Bonn, Germany

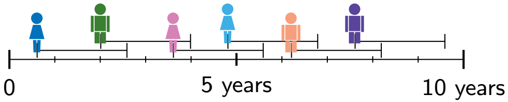
June 10, 2017

Motivation: Hiring with Fixed-Term Contracts

- Classical secretary problem often motivated with a hiring process
- Now, limited time horizon and fixed-term contracts

Motivation: Hiring with Fixed-Term Contracts

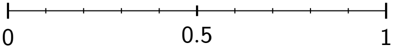
- Classical secretary problem often motivated with a hiring process
- Now, limited time horizon and fixed-term contracts
- E.g. 10 years project, 1 position and 2 year contracts



The Temp Secretary Problem

Fiat et al. [ESA'15]

Example: $\gamma = 0.2$ and $B = 1$

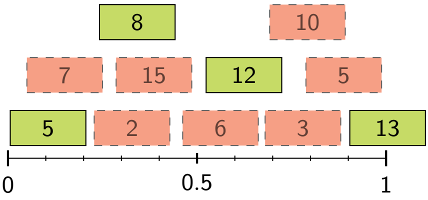


- Weight w_j for each candidate j
- Arrival date $\tau_j \in [0, 1]$ uniformly at random
- Contract durations γ
- Temporal packing constraints, e.g. $\leq B$ candidates at a time
- Objective: $\max \sum_{j \in S} w_j$

The Temp Secretary Problem

Fiat et al. [ESA'15]

Example: $\gamma = 0.2$ and $B = 1$



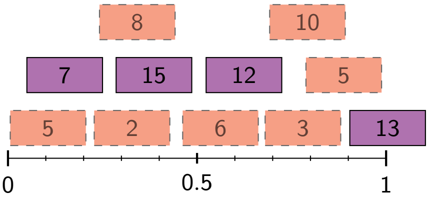
Choice: $5 + 8 + 12 + 13 = 38$

- Weight w_j for each candidate j
- Arrival date $\tau_j \in [0, 1]$ uniformly at random
- Contract durations γ
- Temporal packing constraints, e.g. $\leq B$ candidates at a time
- Objective: $\max \sum_{j \in S} w_j$

The Temp Secretary Problem

Fiat et al. [ESA'15]

Example: $\gamma = 0.2$ and $B = 1$



Choice: $5 + 8 + 12 + 13 = 38$

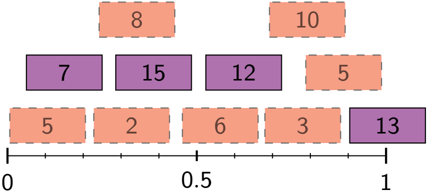
Opt.: $7 + 15 + 12 + 13 = 45$

- Weight w_j for each candidate j
- Arrival date $\tau_j \in [0, 1]$ uniformly at random
- Contract durations γ
- Temporal packing constraints, e.g. $\leq B$ candidates at a time
- Objective: $\max \sum_{j \in S} w_j$

The Temp Secretary Problem

Fiat et al. [ESA'15]

Example: $\gamma = 0.2$ and $B = 1$



Choice: $5 + 8 + 12 + 13 = 38$

Opt.: $7 + 15 + 12 + 13 = 45$

- Weight w_j for each candidate j
- Arrival date $\tau_j \in [0, 1]$ uniformly at random
- Contract durations γ
- Temporal packing constraints, e.g. $\leq B$ candidates at a time
- Objective: $\max \sum_{j \in S} w_j$

- Here $OPT(\mathcal{I})$ is a random variable
- c-competitive if $\mathbf{E}[\mathrm{ALG}(\mathcal{I})] \geq c \cdot \mathbf{E}[\mathrm{OPT}(\mathcal{I})]$

Our Results

We give a **simple algorithm** for the problem with $\gamma \ll 1$ that is

- $\frac{1}{2} - O(\sqrt{\gamma})$-competitive for all B
- $1 - O(\frac{1}{\sqrt{B}}) - O(\sqrt{\gamma})$-competitive for large B

Generalizations

- linear packing constraints
- $\frac{1}{4} - O(\sqrt{\gamma})$-competitive for different lengths $\lambda_j \leq \gamma$ and $B = 1$

A Non-Temporal Relaxation

For every feasible selection of candidates holds:

- at most B candidates selected within last γ time interval
- $\Rightarrow$ at most $B \left\lceil \frac{1}{\gamma} \right\rceil$ candidates selected in total

A Non-Temporal Relaxation

For every feasible selection of candidates holds:

- at most B candidates selected within last γ time interval
- $\Rightarrow$ at most $B \left\lceil \frac{1}{\gamma} \right\rceil$ candidates selected in total

Idea:

- spread selections evenly over arrival interval

Linear Scaling Approach

Attempt selection of candidate j if the candidate is within the $\left\lfloor \tau_j \frac{B}{\gamma} \right\rfloor$ best candidates seen so far.

Details...

Details...

I am happy to discuss details at the poster!