Closest String Problem

Aleksa Kojadinović
130/2017

Uvod

- Postavka problema:

Dato je n niski $s_1, s_2, \ldots s_n$ dužine m.

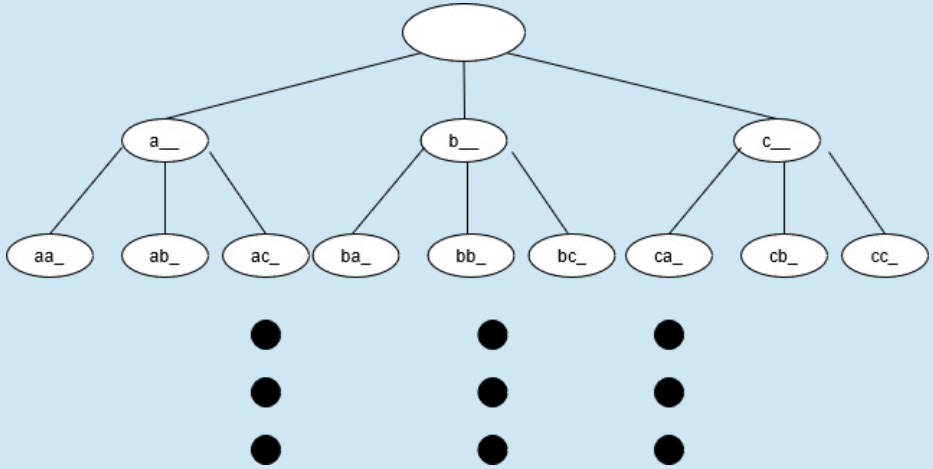
Naći nisku s dužine m koja minimizuje d gde je

$$d = \max\{d_H(s, s_i) | i = 1, \ldots n\}$$

- primer:
  - Azbuka {A, C, T, G}, m = 4, n = 3
  - Niske ‘ACCT‘, ‘AAGT‘, ‘CAGT‘
  - Optimalno rešenje:
    - s = ‘GCGT‘,
    - d_opt = 2
  - Provera:
  - d(GCGT, ACCT) = 2, d(GCGT, AAGT) = 2, d(GCGT, CAGT) = 2
  - max{2, 2, 2} = 2

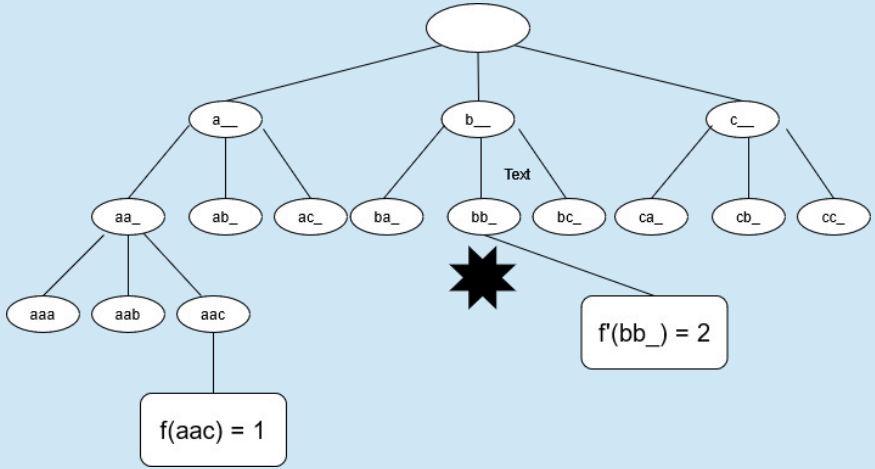
Rešenja > Brute Force > DFS

- pretragom u dubinu naći sve moguće niske dužine **m** nad datom azbukom i izabrati najbolju



Rešenja > Brute Force > DFS sa odsecanjem

- vrši se odsecanje kada string određenog prefiksa već premašuje dosadašnji najbolji rezultat

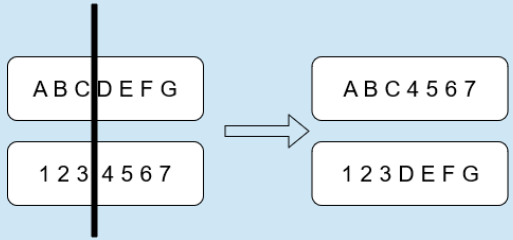


Brute Force vs Odsecanje

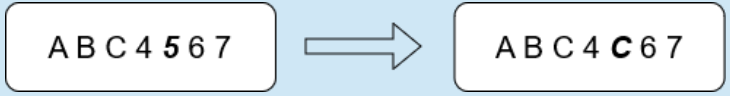
	Brute Force Solver		Pruning Solver	
m	RT	S	RT	S
2	0	1	0	1
3	0	2	0	2
4	0	2	0	2
5	0	2	0	2
6	0	3	0	3
7	0	3	0	3
8	0	4	0	4
9	0	4	0	4
10	0.01	4	0	4
11	0.02	5	0	5
12	0.04	5	0	5
13	0.08	5	0.01	5
14	0.16	6	0.02	6
15	0.34	5	0	5
16	0.69	6	0.02	6
17	1.45	7	0.05	7
18	3.39	6	0.03	6
19	6.62	7	0.16	7
20	15.11	8	0.35	8

Rešenja > Metaheuristike > Genetski algoritam

- **Inicijalna populacija** veličine P – nasumične niske iz date azbuke dužine m
- **Selekcija** – bira se P/2 niski iz populacije na osnovnu njihvih kvaliteta
- **Ukrštanje –** odabir nasumične pozicije i razmena odgovarajućih delova:



- **Mutacije –** promena nasumičnog slova



- **Elitistički pristup –** samo najbolje jedinke opstaju u sledećoj generaciji

Rešenja > Metaheuristike > Kolonija mrava

- **Matrica feromona**

pozicija/ slovo	***a***	***b***	***c***
1	0.25	0.25	0.5
2	0.33	0.25	0.42
3	0.11	0.8	0.09
4	0.2	0.5	0.3
5	0.4	0.2	0.4

a b c

(0.25, 0.25, **0.50**)

(**0.33**, 0.25, 0.42)

(0.11, **0.80**, 0.09)

(0.20, **0.50**, 0.30)

(**0.40**, 0.20, 0.40)

c a b b a

Rešenja > Metaheuristike > Kolonija mrava

- **Evaporacija**
  - sprečava zaglavljivanje u lokalnim optimumima

$$M_{j,\alpha}^{\text{new}} = M_{j,\alpha}^{\text{old}} \times (1 - \rho)$$

Rešenja > Metaheuristike > Kolonija mrava

- **Elitistički pristup**
  - samo najbolji mrav u iteraciji ima pravo da ažurira trag

$$M_{j,\alpha}^{\text{new}} = M_{j,\alpha}^{\text{old}} + (1 - \frac{lb}{m})$$

Rešenja > PTAS

- Osnovna ideja je svođenje nekog potproblema početnog problema na **problem celobrojnog programiranja.**

$$\min d$$

$$\sum_{\alpha \in \Sigma} x_{j,\alpha} = 1, \quad j = 1, \ldots, m$$

svaka pozicija mora imati tačno jedno slovo

$$\sum_{1 \leq j \leq m} \sum_{\alpha \in \Sigma} \chi(s_i[j], \alpha) \leq d, \quad i = 1, \ldots, n$$

distanca do svake ulazne niske je najviše *d*

$$x_{j,\alpha} \in \{0, 1\} \quad (\forall j)(\forall \alpha)$$

celobrojnost

Rešenja > PTAS

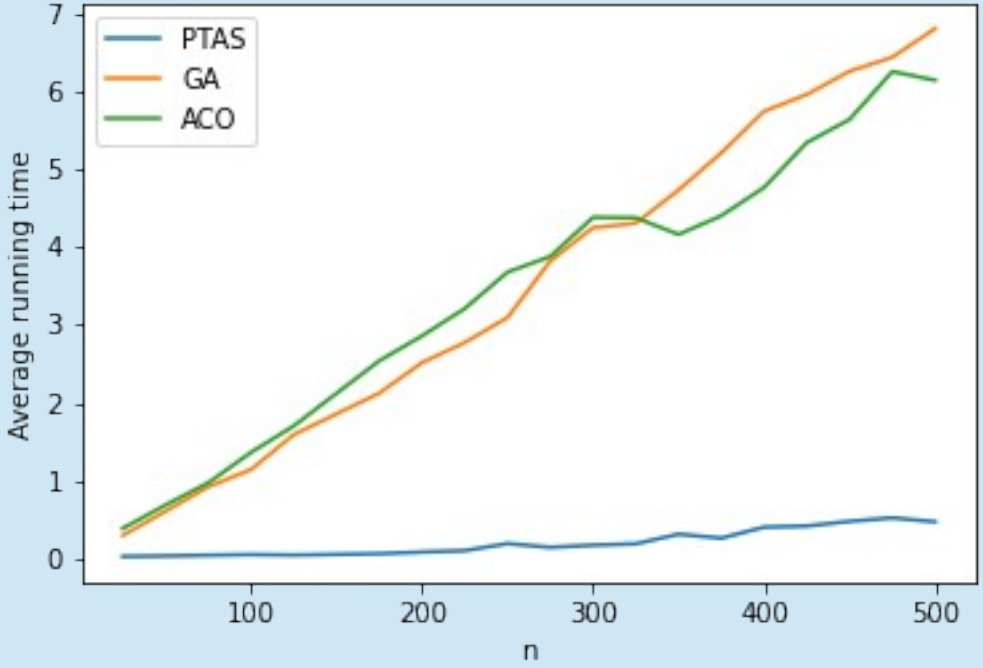
- Pravimo potproblem

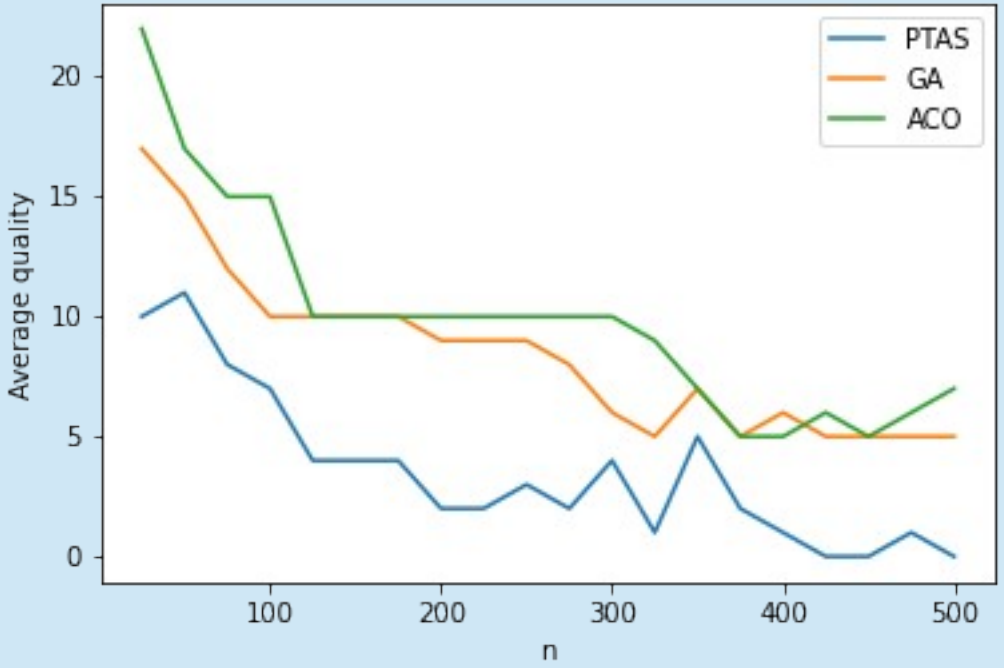
$$s1, s2 = \text{argmax}_{i \neq j} d_H(s_i, s_j)$$
$$d_H(s_1, s_2) = k$$

- Optimizaciju vršimo nad onim pozicijama gde se s1 i s2 ne slažu
- U zavisnosti od kompleksnosti dobijenog problema rešavamo ga:
  - LP relaksacijom
    - Odbacimo uslov celobrojnosti. Za svaku poziciju dobijamo raspodelu verovatnoća za svako slovo
  - grubom silom

Eksperimentalni rezultati > n

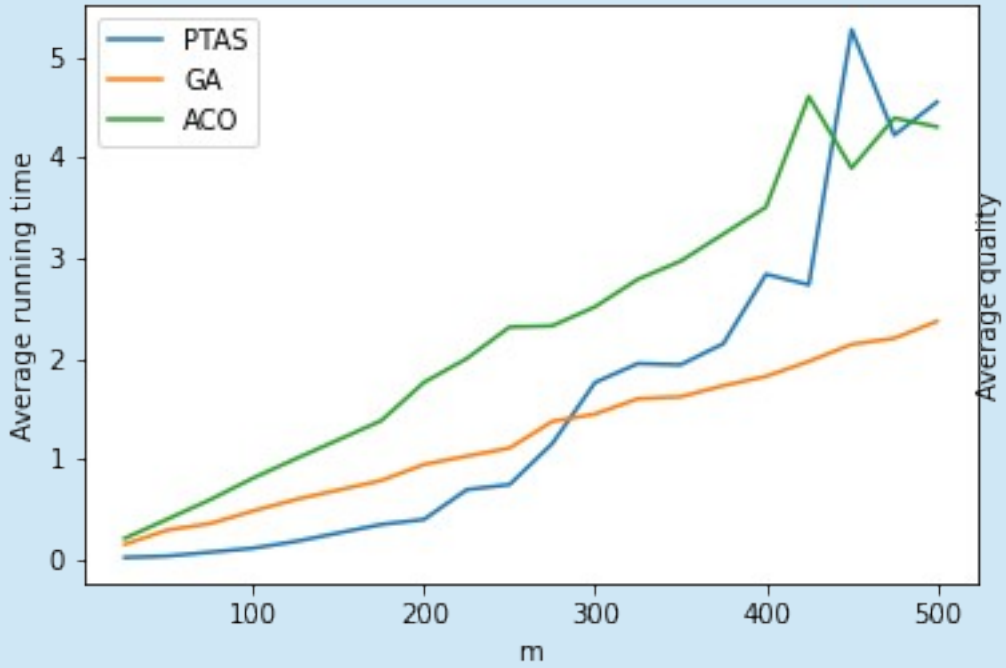
- Povećavanje broja ulaznih niski (*n*)

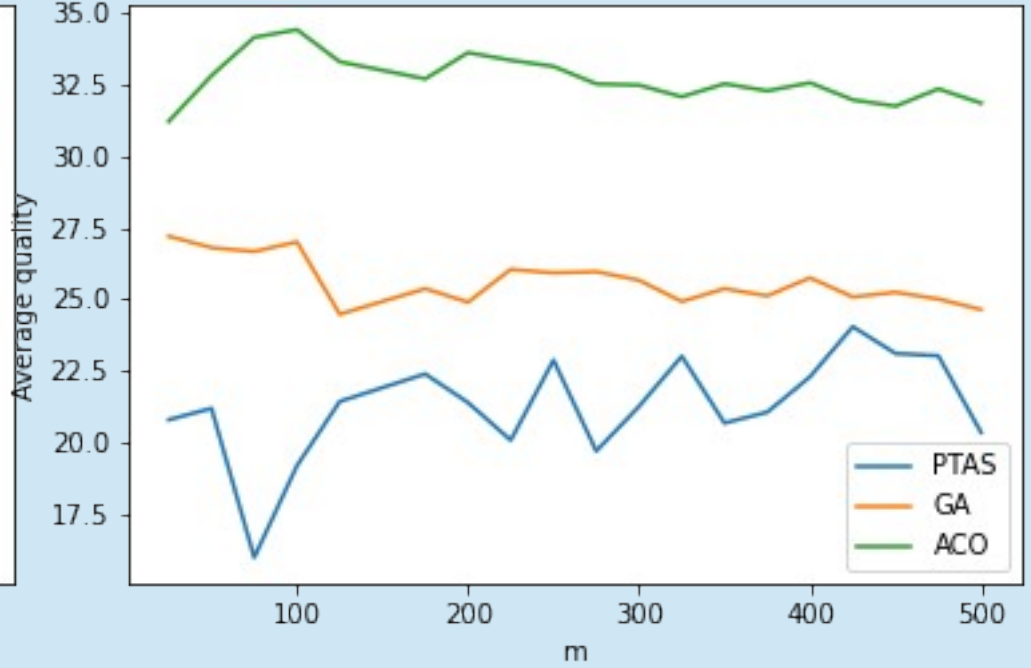




Eksperimentalni rezultati > m

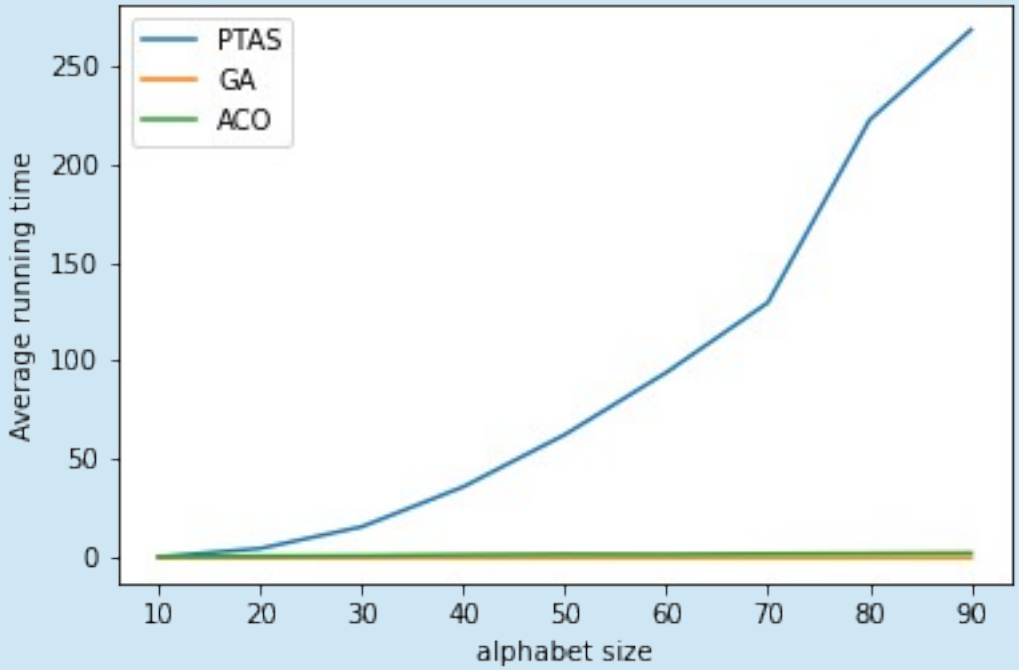
- Povećavanje dužina niski (*m*)

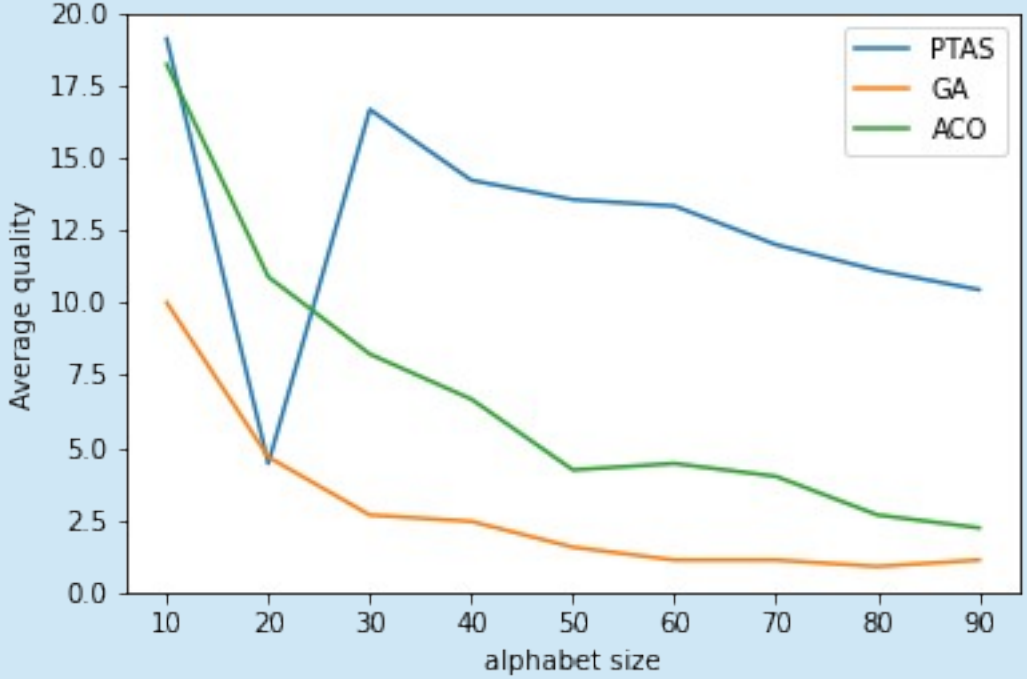




Eksperimentalni rezultati > veličina azbuke

- Povećavanje veličine azbuke $|\Sigma|$





Zaključak

- Najbolje sveukupno ponašanje pokazuje ACO
  - često prednjači i u kvalitetu i u vremenu izvršavanja
- Genetski algoritam
  - pokazao se vremenski bolji u slučaju povećanja *m* i azbuke
  - po kvalitetu blizu ACO
- PTAS ima najbolji kvalitet kod povećanja azbuke
  - ali vremena izvršavanja su neprihvatljiva