

UNIVERZITET U BEOGRADU
MATEMATIČKI FAKULTET



Miloš P. Miković

ALGORITMI ZA REŠAVANJE PROBLEMA
NAJKRAĆE ZAJEDNIČKE NADNISKE

master rad

Beograd, 2021.

Mentor:

dr Aleksandar KARTELJ, docent
Univerzitet u Beogradu, Matematički fakultet

Članovi komisije:

dr Vladimir FILIPOVIĆ, redovni profesor
Univerzitet u Beogradu, Matematički fakultet

dr Stefan MIŠKOVIĆ, docent
Univerzitet u Beogradu, Matematički fakultet

Datum odbrane: _____

Hvala profesoru Aleksandru Kartelju.

Naslov master rada: Algoritmi za rešavanje problema najkraće zajedničke nadni-
ske

Rezime:

Ključne reči: optimizacija, pretraga bima, analiza bioloških sekvenci

Sadržaj

1	Uvod	1
1.1	Problem najkraće zajedničke nadniske	1
1.2	Pregled dosadašnjih istraživanja	2
2	Razrada	4
3	Zaključak	5
	Bibliografija	6

Glava 1

Uvod

Problem najkraće zajedničke nadniske (*eng.* Shortest Common Supersequence Problem) jedan je od dobro poznatih NP-teških problema optimizacije u oblasti analize reči [1]. Ukratko, PNZN¹ se može opisati kao problem pronalaženja najkraće reči ω sačinjene od simbola zadate konačne Azbuke Σ , tako da su sve sekvence iz unapred zadatog konačnog skupa \mathcal{L} sadržane u sekvenci ω . Kada se kaže da su sve reči iz skupa \mathcal{L} sadržane, misli se na to da se svaka reč iz skupa \mathcal{L} može dobiti uklanjanjem simbola iz reči ω ali u zadatom redosledu [3]. PNZN ima primene u mnogim oblastima informatike uključujući kompresiju podataka [4], optimizaciju upita [11], analizu i poređenje teksta i bioloških sekvenci [13] [2]. Kao rezultat velike primene u mnogim oblastima, postoji veliki broj istraživanja na temu ovog problema u pokušaju da se dođe do što boljeg i prihvatljivijeg rešenja.

1.1 Problem najkraće zajedničke nadniske

U ovom poglavlju formalno ćemo definisati PNZN, ali pre toga uvešćemo potrebnu notaciju koja će biti korišćena u nastavku teksta. Konačna azbuka sastoji se od konačnog broja slova i označavaćemo je sa Σ . Svaka konačna reč $\omega = \omega(1)\omega(2)\dots\omega(n)$ sastoji se od konačnog broja slova azbuke gde $\omega(j) \in \Sigma$ predstavlja j-to slovo reči $\omega \in \Sigma^*$. Duzinu reči ω označavaćemo sa $|\omega|$, praznu reč sa ε i važi da $|\varepsilon| = 0$. U skladu sa uvedenom notacijom $|\Sigma|$ predstavlja kardinalnost azbuke. Sa $\omega \supseteq \alpha$ označavaćemo broj pojavljivanja slova α u reči ω ($\omega(1)\omega(2)\dots\omega(n) \supseteq \alpha = \sum_{1 \leq i \leq n, \omega(i)=\alpha} 1$). Reč koja se dobija dodavanjem slova α na početak reči ω označavaćemo sa $\alpha\omega$ (takođe

¹U nastavku teksta PNZN ćemo koristiti kao skraćenicu za problem najkraće zajedničke nadniske

ćemo pisati $\omega = \alpha\omega'$), slično reč koja se dobija skidanjem slova α sa početka reči ω sa $\omega|_\alpha$. Brisanje slova α sa početka svake reči u zadatom skupu, u skladu sa uvedenom notacijom definišemo kao $\{\omega_1, \omega_2, \dots, \omega_n\}|_\alpha = \{\omega_1|_\alpha, \omega_2|_\alpha, \dots, \omega_n|_\alpha\}$.

Neka važi da $\omega_1, \omega_2 \in \Sigma^*$, za reč ω_1 kažemo da je supersekvenca reči ω_2 u oznaci $\omega_1 \succ \omega_2$ ako važi sledeća rekurzivna definicija [1]:

$$\begin{aligned} \omega_1 \succ \varepsilon &\triangleq \text{Tačno} \\ \varepsilon \succ \omega_2 &\triangleq \text{Netačno, Ako } \omega_2 \neq \varepsilon \\ \alpha\omega_1 \succ \alpha\omega_2 &\triangleq \omega_1 \succ \omega_2 \\ \alpha\omega_1 \succ \beta\omega_2 &\triangleq \omega_1 \succ \beta\omega_2, \text{ Ako } \alpha \neq \beta \end{aligned} \tag{1.1}$$

Zapravo, $\omega_1 \succ \omega_2$ označava da se svi simboli iz ω_2 nalaze u ω_1 u datom redosledu, ali ne nužno uzastopno. Na primer, za datu azbuku $\Sigma = \{a, c, t, g\}$, važi $agcatg \succ act$. Sada možemo formalno definisati PNZN. Instanca PNZN može se definisati kao $\mathcal{I} = (\Sigma, \mathcal{L})$, gde Σ predstavlja konačnu azbuku, a \mathcal{L} predstavlja skup od m reči $\{\omega_1, \omega_2, \dots, \omega_m\}$, $\omega_i \in \Sigma^*$. Potrebno je pronaći reč ω najmanje dužine tako da važi da je ω supersekvenca svake reči iz skupa \mathcal{L} ($\omega \succ \omega_i, \forall \omega_i \in \mathcal{L}$ i $|\omega|$ je minimalna). Na primer za instancu PNZN $\mathcal{I} = (\{a, c, t, g\}, \{act, cta, aca\})$, najmanja zajednička supersekvenca instance \mathcal{I} je $acta$.

1.2 Pregled dosadašnjih istraživanja

Problem najkraće zajedničke nadniske prvi je uveo Dejvid Mejer (*eng.* David Maier) 1978. godine u svom radu „The Complexity of Some Problems on Subsequences and Supersequences” [8]. Dokazano je da je PNZN NP-kompletan problem nad svakom azbukom Σ za koju važi da $|\Sigma| \geq 2$ [7]. Korišćenjem dinamičkog programiranja (*eng.* dynamic programming) PNZN nad dve reči dužine n rešen je algoritmom vremenske složenosti $\mathcal{O}(n^2)$ i prostorne složenosti $\mathcal{O}(n^2)$. Algoritam zasnovan na dinamičkom programiranju može biti unapređen, pa tako za k reči dužine maksimalno n , PNZN može biti rešen u $\mathcal{O}(n^k)$ prostornoj i vremenskoj složenosti [12]. Jasno je da ovakav algoritam nije praktičan za velike vrednosti k . S obzirom na to da ne postoji algoritam polinomijalne složenosti koji rešava PNZN, pribegava se optimizacionim metodama u rešavanju ovog problema. Ono što je karakteristično za optimizacioni

pristup rešavanju problema jeste to da se formira algoritam koji rešava postojeći problem tako što daje rešenje koje je prihvatljivo pod određenim uslovima. Takvo rešenje ne mora nužno biti optimalno rešenje problem. Na ovaj način, korišćenjem određene optimizacione tehnike, dobija se algoritam koji se izvršava brzo u realnim uslovima i daje prihvatljivo dobra rešenja.

Vremenom je predloženo mnogo heurističkih i metaheurističkih algoritama za rešavanje PNZN. Neke od poznatijih heurističkih funkcija koje su korišćene u rešavanju PNZN su Alphabet [9], Majority Merge i Weighted Majority Merge [1], Tournament i Greedy [14], Reduce-Expand [9]. Pored navedenih funkcija, korišćeni su i metaheuristički algoritmi, genetski algoritam (*eng.* genetic algorithm) [5] i optimizacija kolonijom (*eng.* colony optimization) [10], koji predstavljaju složenije optimizacione tehnike i imaju tendenciju ka dužem vremenu izvršavanja ne većim instancama problema [6].

Uvedi Beam search kao heuristiku koja će biti korišćena u ovom radu.

Glava 2

Razrada

Glava 3

Zaključak

Bibliografija

- [1] Antonio J. Fernandez Christian Blum, Carlos Cotta and Francisco Gallardo. A Probabilistic Beam Search Approach to the Shortest Common Supersequence Problem. In *Lecture Notes in Computer Science*, 2007.
- [2] Joseph Kruskal David Sankoff. *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*. Center for the Study of Language and Inf, 1983.
- [3] Garey and Johnson. Shortest common supersequence. on-line at: <https://www.csc.kth.se/~viggo/wwwcompendium/node165.html>.
- [4] Storer JA. *Data compression: methods and theory*. Computer Science Press, 1988.
- [5] Frerk Schneider Jurgen Branke, Martin Middendorf. Improved heuristics and a genetic algorithm for finding short supersequences, 1998.
- [6] Hon Wai Leong Kang Ning. Towards a better solution to the shortest common supersequence problem: the deposition and reduction algorithm, 2006.
- [7] Esko Ukkonen Kari-Jouko Raiha. The shortest common supersequence problem over binary alphabet is NP-complete. In *Theoretical Computer Science*, 1981. Volume 16, Issue 2, Pages 187-198.
- [8] David Maier. The Complexity of Some Problems on Subsequences and Supersequences, 1978.
- [9] Gianluca Della Vedova Giancarlo Mauri Paolo Barone, Paola Bonizzoni. An Approximation Algorithm for the Shortest Common Supersequence Problem: An Experimental Analysis, 2001.

- [10] Martin Middendorf Rene Michel. An island model based ant system with lookahead for the shortest supersequence problem. 2006.
- [11] Timos K. Sellis. Multiple-query optimization, 1988. ACM Transactions on Database Systems (TODS), 13(1):23-52.
- [12] Ming Li Tao Jiang. On the Approximation of Shortest Common Supersequences and Longest Common Subsequences, 1995.
- [13] Ronald L. Rivest Clifford Stein Thomas H. Cormen, Charles E. Leiserson. *Introduction to Algorithms, Second edition*. MIT Press and McGraw-Hill, 2001.
- [14] Vadim G. Timkovsky. Some Approximations for Shortest Common Nonsubsequences and Supersequences. In *String Processing and Information Retrieval*, 2006.

Biografija autora

Vuk Stefanović Karadžić (*Tršić, 26. oktobar/6. novembar 1787. — Beč, 7. februar 1864.*) bio je srpski filolog, reformator srpskog jezika, sakupljač narodnih umotvorina i pisac prvog rečnika srpskog jezika. Vuk je najznačajnija ličnost srpske književnosti prve polovine XIX veka. Stekao je i nekoliko počasnih mastera. Učestvovao je u Prvom srpskom ustanku kao pisar i činovnik u Negotinskoj krajini, a nakon sloma ustanka preselio se u Beč, 1813. godine. Tu je upoznao Jerneja Kopitara, cenzora slovenskih knjiga, na čiji je podsticaj krenuo u prikupljanje srpskih narodnih pesama, reformu ćirilice i borbu za uvođenje narodnog jezika u srpsku književnost. Vukovim reformama u srpski jezik je uveden fonetski pravopis, a srpski jezik je potisnuo slavenosrpski jezik koji je u to vreme bio jezik obrazovanih ljudi. Tako se kao najvažnije godine Vukove reforme ističu 1818., 1836., 1839., 1847. i 1852.