Igrajući linkovi

Seminarski rad u okviru kursa
Konstrukcija i Analiza algoritama 2
Matematički fakultet

Petar Đorđević

3. jun 2024.

Sažetak

Ovaj rad istražuje Algoritam igrajućih linkova (*Dancing Links Algorithm - DLX*), koji je efikasna metoda za rešavanje problema pokrivanja tačaka, specifično za rešavanje problema tačnog omotača. Algoritam, koji je osmislio Donald Knut [3], omogućava brzu i efikasnu manipulaciju matricama pokrivanja. Ova tehnika je naročito pogodna za rešavanje složenih kombinatornih problema kao što je raspoređivanje poslova, logičke rešetke, puzzle i slično. U okviru ovog rada, implementiraće se Sudoku rešavač kao konkretan primer primene Algoritma igrajućih linkova (DLX).

Sadržaj

1 Uvod

Problem tačnih pokrivača (*Exact Cover Problem*) je klasičan kombinatorni problem odabira podskupova iz date kolekcije tako da svaki element univerzalnog skupa bude tačno jednom pokriven [1]. Rešavanje ovog problema ima primenu u raznim oblastima kao što su teorija grafova, optimizacija, veštačka inteligencija i računarstvo. Problem tačnog pokrivača je NP-težak, što znači da je izuzetno složen za rešavanje, te zahteva najbolji mogući algoritam za efikasno pronalaženje rešenja.

Jedan od najpoznatijih algoritama za rešavanje ovih problema je Algoritam igrajućih linkova (*Dancing Links Algorithm - DLX*), koji je osmislio Donald Knuth. DLX koristi strukturu podataka poznatu kao igrajući linkovi koja optimizuje operacije dodavanja, uklanjanja i pretraživanja elemenata u matricama pokrivanja.

Sudoku je popularna logička slagalica koja zahteva popunjavanje 9x9 mreže brojevima od 1 do 9, uz poštovanje pravila da se svaki broj mora pojaviti tačno jednom u svakom redu, koloni i 3x3 podmreži. Problem rešavanja Sudokua može se preformulisati kao problem tačnih pokrivača, što ga čini idealnim za primenu DLX-a. Ovo preformulisanje omogućava da se koristi DLX za efikasno pronalaženje rešenja, čak i za najteže zagonetke.

Cilj ovog rada je da implementira Sudoku rešavač koristeći Algoritam igrajućih linkova i da demonstrira efikasnost ovog algoritma. Pored toga, rad će evaluirati performanse DLX-a na različitim primerima Sudoku zagonetki.

2 Problem tačnog omotača

Problem tačnog pokrivača (*Exact Cover Problem - ECP*) predstavlja ključnu podvrstu problema zadovoljstva ograničenja (*Constraint Satisfaction Problems - CSP*), gde se cilj sastoji u pronalaženju podskupa elemenata koji tačno pokrivaju dati skup. Svaki podskup se može posmatrati kao klauza, a tačno pokrivanje zahteva da svaka klauza bude zadovoljena tačno jednom.

ECP se može definisati na sledeći način: Dat je univerzalni skup U i kolekcija S podskupova U. Cilj je pronaći podskup $S' \subseteq S$ takav da su svi elementi iz U tačno jednom pokriveni, odnosno svaki element iz U pripada tačno jednom podskupu iz S'.

Pošto se ECP može reukovati na CSP probleme [2] znamo da je to NP-težak problem, što i ima smisla intuitivno: da bi se utvrdilo da li dati skup podskupova sadrži tačno pokrivanje, potrebno je proveriti sve moguće kombinacije, što može dovesti do eksponencijalnog rasta vremena izvršavanja.

Postoje različiti pristupi rešavanju ECP-a, među kojima su i algoritmi koji koriste tehniku pretraživanja unazad, kao što su algoritam tačne pretrage i algoritam podeli pa vladaj. Pored toga, neki od najefikasnijih algoritama za rešavanje problema tačnog pokrivača su bazirani na pretraživanju uz upotrebu algoritama pomoću heuristika, kao što su gramzivi algoritmi i algoritmi zasnovani na tačnom pokrivanju kontraprimera (*backtracking*). Ovi algoritmi kombinuju preciznost i efikasnost kako bi pronašli optimalna rešenja ili dobra približna rešenja problema tačnog pokrivača.

3 Slike i tabele

Slike i tabele treba da budu u svom okruženju, sa odgovarajućim naslovima, obeležene labelom da koje omogućava referenciranje.

Primer 3.1 *Ovako se ubacuje slika. Obratiti pažnju da je dodato i* `\usepackage{graphicx}`

Slika 1: Pande

Na svaku sliku neophodno je referisati se negde u tekstu. Na primer, na slici 1 prikazane su pande.

Primer 3.2 *I tabele treba da budu u svom okruženju, i na njih je neophodno referisati se u tekstu. Na primer, u tabeli 1 su prikazana različita poravnanja u tabelama.*

Tabela 1: Razlčita poravnanja u okviru iste tabele ne treba koristiti jer su nepregledna.

centralno poravnanje	levo poravnanje	desno poravnanje
a	b	c
d	e	f

4 Prvi naslov

Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst.

4.1 Prvi podnaslov

Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst.

4.2 Drugi podnaslov

Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst.

4.3 ... podnaslov

Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst.

5 n-ti naslov

Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst.

5.1 ... podnaslov

Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst.

5.2 ... podnaslov

Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst.

6 Zaključak

Ovde pišem zaključak. Ovde pišem zaključak. Ovde pišem zaključak. Ovde pišem zaključak. Ovde pišem zaključak. Ovde pišem zaključak. Ovde pišem zaključak. Ovde pišem zaključak. Ovde pišem zaključak. Ovde pišem zaključak. Ovde pišem zaključak.

Literatura

[1] Maxime Chabert and Christine Solnon. A global constraint for the exact cover problem: Application to conceptual clustering. *Journal of Artificial Intelligence Research*, 67:509–547, 2020.

[2] Irit Dinur and Gillat Kol. Covering csps. *Weizmann Institute*, 2013.

[3] Donald E. Knuth. Dancing links. *Millenial Perspectives in Computer Science*, pages 187–214, 2000. Comments: Abstract added by Greg Kuperberg.