Posudek vedoucího diplomové práce
Pavla Žohy: Algoritmy konstrukce sufixového pole

Sufixové pole je datová struktura, navržená v roce 1990 jako prostorově úsporná varianta sufixového stromu, která je tradičně využívána pro kompresi dat (implementace Burrowsovy-Wheelerovy transformace), její hlavní aplikací se však ukazuje být efektivní řešení úloh z oblasti vyhledávání v textu. Nedávná práce dokonce dokázala, že algoritmy pracující nad sufixovým stromem lze systematicky nahradit algoritmy nad sufixovým polem s dodatečnými informacemi bez zhoršení asymptotické časové složitosti. I tento výsledek se stal motivací pro vznik nových metod pro konstrukci sufixového pole, a to jak algoritmů pracujících v lineárním čase, tak jiných se složitostí v nejhorším případě až superkvadratickou, které však jsou při praktickém použití často rychlejší.

Cílem posuzované práce bylo provést praktické porovnání efektivity algoritmů pro konstrukci sufixového pole. Autor se měl na základě studia literatury pokusit vytvořit taxonomii známých konstrukčních algoritmů, vybrat několik typických reprezentantů, implementovat je v jednotném prostředí a poté porovnat jejich efektivitu. Výsledkem měl být nejen obvyklý odhad časové a prostorové náročnosti na vhodně vybraném souboru experimentálních dat, ale též pokus o konstrukci mezních případů, které by odhalily slabiny jednotlivých algoritmů. Součástí zadání bylo též doporučení provést srovnání s nějakou příbuznou sufixovou datovou strukturou.

Popis vybraných konstrukčních algoritmů a jejich implementace, vysvětlení výběru a způsobu generování testovacích dat jakož i přehled výsledků experimentů s jejich analýzou jsou obsahem textové části práce. Komentované zdrojové kódy programů, testovací data a výsledky v práci popsaných experimentů tvoří její elektronickou přílohu.

Na textu práce oceňuji kultivovaný výklad a srozumitelné vyjadřování, díky kterému bych práci neváhal doporučit jako úvodní text všem zájemcům o tuto datovou strukturu. Po výkladu nezbytného teoretického pozadí se autor věnuje popisu jednoho klasického a tří novějších algoritmů pro přímou konstrukci sufixového pole, a dále popisuje, jak dosáhnout téhož cíle nepřímo prostřednictvím konstrukce sufixového stromu. Součástí výkladu jsou i vlastní implementační poznámky; jedním z cílů práce bylo totiž implementovat algoritmy na základě jejich popisu v původních článcích, bez použití cizího zdrojového kódu, pokud takový existuje.

Následuje zřejmě nejzajímavější část textu, věnovaná srovnání algoritmů. Autor se nejprve stručně věnuje otázce, jak odstínit měření prostorové a časové náročnosti výpočtu programu od zkreslení, způsobeného během procesů operačního systému. Následuje popis experimentů, v nichž byly testovány popsané vlastní implementace pěti konstrukčních algoritmů spolu s volně dostupnými verzemi čtyř klasických třídících algoritmů (quicksort, mergesort, heapsort a shellsort). Každý z algoritmů byl rozšířen tak, aby pracoval se vstupy jak nad obvyklou 8bitovou, tak rozšířenou 16bitovou abecedou. Testy byly prováděny na reálných souborech, náhodně generovaných datech a speciálních posloupnostech. Pro určení závislosti na délce vstupu bylo ze souborů náhodně extrahováno vždy deset úseků stejné velikosti 2^7, 2^8,..., 2^{21} znaků. V případě náhodně generovaných vstupů byly posloupnosti těchto délek generovány pro abecedy velikostí 2^1, 2^2,..., 2^{16}, což umožnilo shrnout výsledky experimentů nejen do obvyklých tabulek, ale i do grafu, který pro každou kombinaci délky souboru a velikosti abecedy udává nejúspěšnější metodu. Existence testovacích dat zadaných délek i velikostí abecedy umožnilo poměrně přesně analyzovat chování algoritmů v závislosti na těchto veličinách.

Hlavním výsledkem experimentů je očekávané potvrzení výsadního postavení algoritmu Manzini-Ferragina, který je obecně nejrychlejší, i když jeho časová složitost měřená nejhorším případem je asymptoticky nejhorší. Naopak algoritmus Kärkkäinen-Sanders s lineární časovou složitostí je v průměrném případě pomalý, i když poměrně stabilní v tom smyslu, se doba běhu pro data stejného rozsahu se příliš neliší. Je zde uvedena i řada dalších, méně zásadních, leč zajímavých pozorování, např. o typu dat, kritických pro jednotlivé algoritmy, o úspěchu klasického třídění sléváním nad náhodně generovanými daty či o rozšiřitelnosti všech metod, s výjimkou konstrukce sufixového stromu, na abecedy s více nežli 256 znaky.

Dle mého soudu jde o zdařilou práci, která poskytuje cenné svědectví o efektivitě vybraných konstrukčních algoritmů. Její přínos ve srovnání s podobně zaměřenými komparativními studiemi vidím v tom, že zkoumá jak reálná, tak náhodně generovaná data, nabízí výsledky v závislosti na délce vstupu i velikosti abecedy, zkoumá chování i nad většími abecedami a konečně konstruuje speciální vstupy, které jsou pro jednotlivé algoritmy kritické. Myslím, že výsledky by mohly být zajímavé i pro širší publikum, a dalo by se tedy uvažovat o jejich publikaci, předtím by ale bylo zřejmě žádoucí rozšířit popsané experimenty i na další algoritmy konstrukce sufixového pole (od roku 2000 jich bylo publikováno alespoň deset).

Pokud jde o celkové hodnocení práce, domnívám se, že její obsah splňuje všechny požadavky, uvedené v pokynech pro vypracování tohoto tématu, a proto mohu doporučit, aby byla přijata jako práce diplomová.

V Praze dne 28. ledna 2007