ÜTEMEZŐ ALGORITMUSOK HIBAFÜGGVÉNYEI

(Processzorszámot minimalizáló algoritmusok)

Iványi Antal, 2011. április 7.

1. Bevezetés

Tegyük fel, hogy egymástól független programokat kell futtatnunk, és minden számítógép csak egységnyi ideig működik (az egység lehet például 8 óra). Célunk a programok olyan szétosztása a gépek között, hogy minél kevesebb gépet használjunk.

A feladat formális megfogalmazása a következő. Legyen $n \geq 1$ egész szám, $\mathbf{t} = (t_1, t_2, \ldots, t_n)$ pedig valós számokat tartalmazó vektor, ahol $t_i \in (0, 1])$, minden $i = 1,\ 2, \ldots,\ n$ indexre. Osszuk fel a sorozatot minél kevesebb olyan részsorzatra, melyekben az elemek összege legfeljebb 1.

A feladat egydimenziós ládapakolási feladatként is ismert [19, 15, 24, 28]. A feladat NP-teljes [19, 24] (például visszavezethető az összegzési feladatra [19, 24]), ezért a gyakorlati megoldásra közelítő algoritmusokat [15, 23, 22] alkalmaznak.

Ebben a cikkben a legismertebb közelítő algoritmusok (NF, FF, BF, NFD, BFD, FFD) legrosszabb esetére vonatkozó eredményeket foglaljuk össze [14, 26, 25, 27].

Két algoritmus (A és B) relatív hatékonyságát gyakran jellemzik a kiválasztott hatékonysági mérték értékeinek hányadosával, jelen esetben az $\mathrm{A}(\mathbf{t})/\mathrm{B}(\mathbf{t})$ relatív lemezszámmal. Ennek a hányadosnak a felhasználásával különböző jellemzők definiálhatók. Ezeket két csoportba szokás sorolni: egyik csoportba a legrosszabb, míg a másikba az átlagos esetet jellemző mennyiségek kerülnek.

Itt csak a legrosszabb esettel foglalkozunk (az átlagos eset vizsgálata rendszerint lényegesen nehezebb).

Legyen $\mathcal{D}_n$ azon valós listák halmaza, amelyek n elemet tartalmaznak, és legyen $\mathcal{D}$ az összes valós lista halmaza, azaz

$$D = \cup_{i=1}^{\infty} D_i \ .$$

Legyen $\mathcal{A}_{lsz}$ az ***állományelhelyező,*** (azaz a minden $\mathbf{t} \in \mathcal{D}$ listához egy nemnegatív valós számot hozzárendelő, és így a $\mathcal{D} \to \mathbb{R}_0^+$ leképezést megvalósító) algoritmusok halmaza.

Legyen $\mathcal{A}_{opt}$ a minden listához az optimális lemezszámot rendelő algoritmusok halmaza, és OPT ennek a halmaznak egy eleme (azaz egy olyan algoritmus, amely minden $\mathbf{t} \in D$ listához megadja a listához tartozó fájlok elhelyezéséhez szükséges és elégséges lemezek számát).

Legyen $\mathcal{A}_{köz}$ azon $\mathrm{A} \in \mathcal{A}_{lsz}$ algoritmusok halmaza, amelyekre $\mathrm{A}(\mathbf{t}) \geq \mathrm{OPT}(\mathbf{t})$ minden $\mathbf{t} \in D$ listára, és van olyan $\mathbf{t} \in D$ lista, amelyre $\mathrm{A}(\mathbf{t}) > \mathrm{OPT}(\mathbf{t})$. Legyen A_{becs} azon $\mathrm{E} \in \mathcal{A}_{lsz}$ algoritmusok halmaza, amelyekre $\mathrm{E}(\mathbf{t}) \leq \mathrm{OPT}(\mathbf{t})$ minden $\mathbf{t} \in D$ listára, és van olyan $\mathbf{t} \in D$ lista, amelyre $\mathrm{E}(\mathbf{t}) < \mathrm{OPT}(\mathbf{t})$.

Legyen F_n azon valós listák halmaza, amelyekre $\mathrm{OPT}(\mathbf{t}) = n$, azaz $F_n = \{\mathbf{t} | \mathbf{t} \in D \text{ és } \mathrm{OPT}(\mathbf{t}) = n\}$ $(n = 1, 2, \ldots)$. A továbbiakban csak $\mathcal{A}_{lsz}$-beli algoritmusokat fogunk vizsgálni. Az A és B algoritmusok ($\mathrm{A},\ \mathrm{B} \in \mathcal{A}$) $R_{\mathrm{A,B},n}$ hibafüggvényét, $R_{\mathrm{A,B}}$ hibáját (abszolút hibáját) és $R_{A,\infty}$ aszimptotikus hibáját a következőképpen definiáljuk:

$$R_{A,B,n} = \sup_{t \in D_n} \frac{\mathrm{A}(\mathbf{t})}{\mathrm{B}(\mathbf{t})} \ ,$$

$$R_{A,B} = \sup_{t \in D} \frac{\mathrm{A}(\mathbf{t})}{\mathrm{B}(\mathbf{t})} \ ,$$

$$R_{A,\infty} = \lim_{n \to \infty} R_{A,B,n} \ .$$

Ezek a mennyiségek főleg akkor érdekesek, ha $\mathrm{B} \in \mathcal{A}_{opt}$. Ilyenkor az egyszerűség kedvéért a jelölésekből elhagyjuk a B-t, és az $\mathrm{A} \in \mathcal{A}$, illetve az $\mathrm{E} \in \mathcal{A}$ algoritmusok hibafüggvényéről, hibájáról és aszimptotikus hibájáról beszélünk.

2. NF algoritmus

A NF fájlelhelyező algoritmus jellemző adatai ismertek.

1.1. tétel. *Ha* $\mathbf{t} \in F_n$, *akkor*

$$n = \mathrm{OPT}(\mathbf{t}) \leq \mathrm{NF}(\mathbf{t}) \leq 2\mathrm{OPT}(\mathbf{t}) - 1 = 2n - 1 \ . \tag{1}$$

Továbbá, ha $k \in \mathbb{Z}$, *akkor léteznek olyan* $\mathbf{u}_k$ *és* $\mathbf{v}_k$ *listák, melyekre*

$$k = \mathrm{OPT}(\mathbf{u}_k) = \mathrm{NF}(\mathbf{u}_k) \tag{2}$$

és

$$k = \mathrm{OPT}(\mathbf{v}_k) \ \textit{és} \ \mathrm{NF}(\mathbf{v}_k) = 2k - 1 \ . \tag{3}$$

Ebből az állításból adódik a NF fájlelhelyező algoritmus hibafüggvénye, abszolút hibája és aszimptotikus hibája.

1.2. következmény. *Ha* $n \in \mathbb{Z}$, *akkor*

$$R_{\mathrm{NF},n} = 2 - \frac{1}{n} \ , \tag{4}$$

továbbá

$$R_{\mathrm{NF}} = R_{\mathrm{NF},\infty} = 2 \ . \tag{5}$$

Iványi [25, 26]

3. FF algoritmus

A FF és BF fájlelhelyező algoritmus legrosszabb esetére vonatkozik a következő állítás.

1.3. tétel. *Ha* $\mathbf{t} \in F_n$, *akkor*

$$\mathrm{OPT}(\mathbf{t}) \leq \mathrm{FF}(\mathbf{t}),\ \mathrm{BF}(\mathbf{t}) \leq 1.7\mathrm{OPT}(\mathbf{t}) + 2 \ . \tag{6}$$

Továbbá, ha $k \in \mathbb{Z}$, *akkor léteznek olyan* $\mathbf{u}_k$ *és* $\mathbf{v}_k$ *listák, melyekre*

$$k = \mathrm{OPT}(\mathbf{u}_k) = \mathrm{FF}(\mathbf{u}_k) = \mathrm{BF}(\mathbf{u}_k) \tag{7}$$

valamint

$$k = \mathrm{OPT}(\mathbf{v}_k) \ \textit{és}\ \mathrm{FF}(\mathbf{v}_k) = \mathrm{BF}(\mathbf{v}_k) = \lfloor 1.7k \rfloor \ . \tag{8}$$

A FF algoritmusra egy erősebb felső korlát is érvényes.

1.4. tétel. *Ha* $\mathbf{t} \in F_n$, *akkor*

$$\mathrm{OPT}(\mathbf{t}) \leq \mathrm{FF}(\mathbf{t}) < 1.7\mathrm{OPT}(\mathbf{t}) + 1 \ . \tag{9}$$

Ebből a két állításból adódik FF és BF aszimptotikus hibája, valamint hibafüggvényük jó becslése.

1.5. következmény. *Ha* $n \in \mathbb{Z}$, *akkor*

$$\frac{\lfloor 1.7n \rfloor}{n} \leq R_{\mathrm{FF},n} \leq \frac{\lceil 1.7n \rceil}{n} \tag{10}$$

és

$$\frac{\lfloor 1.7n \rfloor}{n} \leq R_{\mathrm{BF},n} \leq \frac{\lfloor 1.7n + 2 \rfloor}{n} \tag{11}$$

továbbá

$$R_{\mathrm{FF},\infty} = R_{\mathrm{BF},\infty} = 1.7 \ . \tag{12}$$

Ha n osztható tízzel, akkor a (10) egyenlőtlenségben az alsó és felső határok megegyeznek, azaz ebben az esetben $1.7 = R_{FF,n}$.

Baker-Coffman [3], Iványi [25, 26, 27]

Xia és Tan [29] 2010-ben a következő tételt bizonyították.

1.6. tétel. *Ha* $\mathbf{t} \in F_n$, *akkor*

$$\mathrm{OPT}(\mathbf{t}) \leq \mathrm{FF}(\mathbf{t}) \leq 1.7\mathrm{OPT}(\mathbf{t}) + \frac{7}{10} \, . \tag{13}$$

4. BF algoritmus

5. NFD algoritmus

6. FFD algoritmus

7. BFD algoritmus

8. Átlagos eset

[12]

9. Érdekességek

Párhuzamos algoritmusok: Anderson et al. [1]

Duális probléma: [21]

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

Az irodalomjegyzékben **letölthető** megjegyzéssel szereplő művek letölthetők a megadott honlapról, a **digitálisan** megjegyzéssel szereplő művek letölthetőek a http://compalg.inf.elte.hu/∼tony/Kutatas/Proc-szam/ honlapról, míg a **nyomtatva** megjegyzéssel szereplő műveket nyomtatott formában sikerült megszerezni.

Irodalomjegyzék

[1] R. J. Anderson, E. W. Mayr, M. K. Warmuth. Parallel approximation algorithms for bin packing. *Inf. and Comp.*, **82** (1989) 262–277. **letölthető** 4

[2] B. S. Baker. A new proof for the first-fit decreasing bin-packing algorithm. *J. Algorithms*, **6** (1985) 49–70. **nyomtatva**

[3] B. S. Baker, E. G. Coffman, Jr. A tight asymptotic bound for next-fit-decreasing bin-packing. *SIAM J. Alg. Disc. Meth.*, **2** (1981) 147–152. 4

[4] J. Békési, G. Galambos, H. Kellerer. A 5/4 linear time bin packing algorithm. *JCSS*, **60** (2000) 145–160.

[5] R. Berghammer, F. Reuter. A linear approximation algorithm for bin packing with absolute approximation factor 3/2. *Science of Computer Programming*, **48** (2003) 67–80.

[6] J. O. Berkey, P. Y. Wang. A systolic-based parallel bin packing algorithm. *IEEE Transactions on Parallel and Distributed Systems*, **5** (1994) 769–772.

[7] G. Bilchev. Evolutionary metaphors for the bin packing problem. In *Evolutionary Programming V: Proceedings of the Fifth Annual Conference on Evolutionary Programming* 1996, 333–341.

[8] J. Blazewicz, K. Ecker. A linear time algorithm for restricted bin packing and scheduling problems. *Oper. Res. Lett.*, **2** (1983) 80–83.

[9] A. Bortfeldt. A heuristic for multiple container loading problems. *OR Spektrum*, **22** (2000) 239–261.

[10] J. M. Bourjolly, V. Rebetez. An analysis of lower bound pro- cedures for the bin packing problem. *Computers and Operations Research*, **32** (3) (2005) 395–405.

[11] A. R. Brown. Optimum Packing and Depletion. American Elsevier, New York, 1971. **nyomtatva**

[12] E. G. Coffman, Jr., C. Courcoubetis, M. R. Garey, D. S. Johnson, P. W. Shor, R. R. Weber, M. Yannakakis. Bin packing with discrete item sizes, Part I: Perfect packing theorems and the average case behavior of optimal packings. *SIAM J. Disc. Math.*, **13** (2000) 384-402. 4

[13] E. G. Coffman, Jr., C. Courcoubetis, M. R. Garey, D. S. Johnson, P. W. Shor, R. R. Weber, M. Yannakakis. Perfect packing theorems and the average-case behavior of optimal and online bin packing. *SIAM Review*, **44** (2002) 95–108.

[14] E. G. Coffman, Jr., M. R. Garey, D. S. Johnson. Approximation algorithms for bin-packing: An updated survey. In G. Ausiello, M. Lucertini, P. Serafini, editors, Algorithm Design for Computer System Design. Springer-Verlag, Wien, 1984. CISM Courses and Lectures Number 284, 49–106. 1

[15] Csirik János: Ládapakolási algoritmusok. Szeged, 1989. 123 oldal. **nyomtatva** 1

[16] J. Csirik, G. Galambos. An $O(n)$ bin-packing algorithm for uniformly distributed data. *Computing* **36** (1986) 313–319.

[17] J. Csirik, G. Galambos. On the expected behaviour of the NF algorithm for a dual bin packing problem. *Acta Cybernetica* **8** (1987) 5–9.

[18] E. G. Coffman, Jr., M. R. Garey, D. S. Johnson, An application of bin-packing to multiprocessor scheduling, *SIAM Journal on Computing*, **7** 1978, 1–17.

[19] T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein, *Introduction to Algorithms*, The MIT Press/McGraw-Hill, 2009 (Harmadik kiadás). Magyarul: *Algoritmusok.* Műszaki Kiadó, Budapest, 1999 (az 1991-es első angol nylvű kiadás fordítása). 1

[20] J. Csirik, J. B. G. Frenk, A. Frieze, G. Galambos, A. H. G. Rinnooy Kan. A probabilistic analysis of the next fit decreasing bin packing heuristic. *Oper. Res. Lett.*, **5** (1986) 233–236.

[21] J. Csirik, J. B. G. Frenk, G. Galambos, A. H. G. Rinnooy Kan. Probabilistic analysis of algorithms for dual bin packing problems. *J. Algorithms*, **12** (1991) 189–203. 4

[22] Dósa Gábor, Imreh Csanád: *Online algoritmusok.* Elektronikus jegyzet. SZTE TTIK, Szeged, 2011 (kézirat). 1

[23] Galambos Gábor: Ládapakolási feladatok közelítő algoritmusainak legrosszabb-eset vizsgálata. Kandidátusi értekezés tézisei. Szeged, ????. **nyomtatva** 1

[24] M. Garey, D. S. Johnson: *Computers and Intractability.* 1979 1

[25] A. Iványi: Performance bounds for simple bin packing algorithms. *Ann. Univ. Sci. Budapest. Sect. Comput.* **5** (1984) 77–82. MR0822601 (87h:68067) **letölthető** 1, 3, 4

[26] A. Iványi: Estimation of the efficiency of bin-packing algorithms. (Russian) *Problemy Kibernet.* **41** (1984) 253–256. **digitálisan** 1, 3, 4

[27] A. Iványi: Tight worst-case bounds for bin packing algorithms. Theory of algorithms (Pécs, 1984), Colloq. Math. Soc. János Bolyai, 44, North-Holland, Amsterdam, 1985, 233–240. MR0872310 (88g:68051) **nyomtatva** 1, 4

[28] Lovász László, Gács Péter: *Algoritmusok.* Tankönyvkiadó, Budapest, 1989. 1

[29] B. Xia, Z. Than: Tighter bounds of the First Fit algorithm for the bin-packing problem. *Discrete Applied Mathematics* **158** (2010) 1668–1675. **letölthető** 4

A szerző címe: Iványi Antal: tony@compalg.inf.elte.hu
Budapest, 2011. április 4.