ÜTEMEZŐ ALGORITMUSOK HIBAFÜGGVÉNYEI

(Processzorszámot minimalizáló algoritmusok)

Iványi Antal, 2011. április 7.

(Ez a kézirat letölthető a következő címről:
http://compalg.inf.elte.hu/∼ tony/Kutatas/TAMOP/Papers-in-journals/)

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 részsorozatra, melyekben az elemek összege legfeljebb 1.

A feladat egydimenziós ládapakolási feladatként is ismert [16, 17, 25, 32]. A feladat NP-teljes [16, 25] (például visszavezethető az összegzési feladatra [16, 25]), ezért a gyakorlati megoldásra közelítő algoritmusokat [17, 22, 23] alkalmaznak.

Ebben a cikkben a legismertebb közelítő algoritmusok (NF, FF, BF, NFD, BFD, FFD) legrosszabb esetére vonatkozó eredményeket foglaljuk össze [14, 26, 27, 28].

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 $\mathcal{A}_{lsz}$ a ***ládapakoló,*** (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 ládaszámot 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

Az NF algoritmusra vonatkozó első jellemzés 1972-ben jelent meg, D. S. Johnson PhD disszertációjának kéziratában. Ma inkább a megvédett értekezésére [31] hivatkoznak.

1.1. tétel. (Johnson, 1972) *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

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

Bizonyítás. Johnson a felső korlátot úgy bizonyította, hogy feltette a korlát ellenkezőjét, miszerint van olyan $L = (t_1, t_2, \ldots, T_n)$ lista, amelyre $\mathrm{OPT}(L) = n$ és $\mathrm{FF}(L) \geq 2n$. Mivel NF csak akkor nyit meg új ládát, ha a soron következő tárgy nem fér be az utolsó megnyitott ládába, ezért az FF ütemezésében a szomszédos ládákban lévő tárgyak súlyösszege nagyobb egynél, így a $2n$ ládában n-nél nagyobb lenne a súlyösszeg, és OPT nem tudná a listát n ládába bepakolni.

A tétel második részét pedig olyan u_k listával bizonyította, amely k darab egyest tartalmazott, és olyan v_k listával, amely $4k - 4$ elemet tartalmazott úgy, hogy a páratlan indexű elemek súlya $1/2$ volt, míg a páros indexűeké $1/(2k - 2)$. ■

1984-ben a [26] cikkben ennél pontosabb (éles) korlátot adtunk.

Legyen ν azon ládapakoló algoritmusok halmaza, amelyek ütemezésében a szomszédos ládákba kerülő tárgyak súlyösszege mindig nagyobb egynél.

1.2. tétel. (Iványi, 1984) *Ha $\mathbf{t} \in F_n$, akkor minden $A \in \nu$ ládapakoló algoritmusra fennáll*

$$n = \mathrm{OPT}(\mathbf{t}) \leq \mathrm{A}(\mathbf{t}) \leq 2\mathrm{OPT}(\mathbf{t}) - 1 = 2n - 1 \; . \tag{4}$$

Továbbá, ha $k \in \mathbb{Z}$, akkor létezik olyan $B \in \nu$ ládapakoló algoritmus és léteznek olyan $\mathbf{u}_k$ és $\mathbf{v}_k$ listák, melyekre

$$k = \mathrm{OPT}(\mathbf{u}_k) = \mathrm{B}(\mathbf{u}_k) \tag{5}$$

és

$$k = \mathrm{OPT}(\mathbf{v}_k) \text{ és } \mathrm{B}(\mathbf{v}_k) = 2k - 1 \; . \tag{6}$$

Bizonyítás. Johnson bizonyításán csak annyit változtatunk, hogy a v_k lista $4k - 2$ elemet tartalmaz úgy, hogy a páratlan indexű elemek súlya $1/2$, míg a páros indexű elemek súlya $1/(4k - 2)$. ■

Ez a tétel magyarul is megtalálható a [30] könyv 500. oldalán, mint 11.9. tétel.

A NF ládapakoló algoritmus jellemző adatai ismertek.

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

$$n = \mathrm{OPT}(\mathbf{t}) \leq \mathrm{NF}(\mathbf{t}) \leq 2\mathrm{OPT}(\mathbf{t}) - 1 = 2n - 1 \; . \tag{7}$$

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

$$k = \mathrm{OPT}(\mathbf{u}_k) = \mathrm{NF}(\mathbf{u}_k) \tag{8}$$

és

$$k = \mathrm{OPT}(\mathbf{v}_k) \text{ és } \mathrm{NF}(\mathbf{v}_k) = 2k - 1 \ . \tag{9}$$

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

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

$$R_{\mathrm{NF},n} = 2 - \frac{1}{n} \ , \tag{10}$$

továbbá

$$R_{\mathrm{NF}} = R_{\mathrm{NF},\infty} = 2 \ . \tag{11}$$

3. FF algoritmus

M. R. Garey, R. L. Graham, J. D. Ullman [24] 1973-ban publikáltak egy olyan L listát, amelyre $\mathrm{OPT}(L) = 10$ és $\mathrm{FF}(L) = 17$. Egyúttal azt sejtették, hogy ha az optimális ládaszám nagyobb, mint 10, akkor fennáll az

$$\mathrm{FF(L)} < 1,7\mathrm{OPT(L)}.$$

Ezt a sejtést sikerült a [26, 27] cikkekben megcáfolni.

A FF és BF fájlelhelyező algoritmus legrosszabb esetére vonatkozik a következő állítás [29]

1.5. tétel. (Ivanyi1986) *Ha $\mathbf{t} \in F_n$, akkor*

$$\mathrm{FF}(\mathbf{t}) < \frac{7}{4}. \tag{12}$$

1994-ben D. Simchi-Levi [33] egy valamivel gyengébb állításra publikált bonyolultabb bizonyítást (viszont az ő bizonyítása a BF algoritmusra is jó).)

[24].

1.6. tétel. (Garey, Graham, Ullman, 1973) *Ha $\mathbf{t} \in F_n$, akkor*

$$\mathrm{OPT}(\mathbf{t}) \leq \mathrm{FF}(\mathbf{t}), \ \mathrm{BF}(\mathbf{t}) \leq 1.7\mathrm{OPT}(\mathbf{t}) + 2 \ . \tag{13}$$

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

$$k = \mathrm{OPT}(\mathbf{u}_k) = \mathrm{FF}(\mathbf{u}_k) = \mathrm{BF}(\mathbf{u}_k) \tag{14}$$

valamint

$$k = \mathrm{OPT}(\mathbf{v}_k) \text{ és } \mathrm{FF}(\mathbf{v}_k) = \mathrm{BF}(\mathbf{v}_k) = \lfloor 1.7k \rfloor \ . \tag{15}$$

A FF algoritmusra egy erősebb felső korlát is érvényes.

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

$$\mathrm{OPT}(\mathbf{t}) \leq \mathrm{FF}(\mathbf{t}) < 1.7\mathrm{OPT}(\mathbf{t}) + 1 \ . \tag{16}$$

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

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

$$\frac{\lfloor 1.7n \rfloor}{n} \leq R_{\mathrm{FF},n} \leq \frac{\lceil 1.7n \rceil}{n} \tag{17}$$

és

$$\frac{\lfloor 1.7n \rfloor}{n} \leq R_{\mathrm{BF},n} \leq \frac{\lfloor 1.7n + 2 \rfloor}{n} \tag{18}$$

továbbá

$$R_{\mathrm{FF},\infty} = R_{\mathrm{BF},\infty} = 1.7 \ . \tag{19}$$

Ha n osztható tízzel, akkor a (17) egyenlőtlenségben az alsó és felső határok megegyeznek, azaz ebben az esetben $1.7 = R_{FF,n}$.

Baker-Coffman [3], Iványi [26, 27, 28]

Korlát a legrosszabb esetre: [33].

Xia és Tan [34] 2010-ben a következő tételt bizonyították.

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

$$\mathrm{OPT}(\mathbf{t}) \leq \mathrm{FF}(\mathbf{t}) \leq 1.7\mathrm{OPT}(\mathbf{t}) + \frac{7}{10} \ . \tag{20}$$

4. BF algoritmus

Korlát a legrosszabb esetre: [33]

5. NFD algoritmus

6. FFD algoritmus

Korlát a legrosszabb esetre: [33]

Legjobb korlát a legrosszabb esetre: [35]

7. BFD algoritmus

Korlát a legrosszabb esetre: [33]

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/BinPacking/ 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** 6

[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. 5

[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 procedures 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. 6

[13] E. G. Coffman, Jr., 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., 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] 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.

[16] 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). **Informatikai Könyvtár** 1

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

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

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

[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. 6

[22] Dósa György, 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. R. Garey, R. L. Graham, J. D. Ullman: An analysis of some packing algorithms. In: *Combinatorial Algorithms*, ed. R. Rustin. Algorithmic Press, 1973, 39-48. 4

[25] M. Garey, Johnson: *Computers and Intractability.* 1979. **Informatikai Könyvtár** 1

[26] 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, 5

[27] A. Iványi: Estimation of the efficiency of bin-packing algorithms. (Russian) *Problemy Kibernet.* **41** (1984) 253–256. **digitálisan** 1, 4, 5

[28] 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, 5

[29] Iványi A.: *Processzorütemezés.* Kézirat. ELTE, Numerikus és Gépi Matematika Tanszék, Budapest, 2004. 4

[30] Iványi A.: *Informatikai algoritmusok.* ELTE Eötvös Kiadó, Budapest, 2004. Elektronikusan: ELTE Informatikai Kar, ??? 3

[31] D. S. Johnson: *Near-Optimal Bin Packing Algorithms.* PhD thesis, Massachusetts Institute of Technology, Department of Mathematics, Cambridge, 1973. 2

[32] Lovász László, Gács Péter: *Algoritmusok.* Tankönyvkiadó, Budapest, 1989. **ELTE Matematikai Könyvtárában** 1

[33] D. Simchi-Levi: New worst-case results for the bin-packing problem. *Naval Research Logistics* **41** (1994) 579–585. **digitálisan** 4, 5, 6

[34] B. Xia, Z. Than: Tighter bounds of the First Fit algorithm for the bin-packing problem. *Discrete Applied Mathematics* **158** (2010) 1668–1675. **letölthető** 5

[35] M. Yue: A simple proof of the inequaliy $\mathrm{FFD}(L \leq 11/9)\mathrm{OPT}(L)+1$ *Acta Math. Applicatae Sinica* **7** (4) (1991) 322–331. **digitálisan** 5

A szerző címe: Iványi Antal: tony@compalg.inf.elte.hu

Budapest, 2011. április 7.

Tárgymutató

Névmutató