

ÜTEMEZŐ ALGORITMUSOK HIBAFÜGGVÉNYEI

(Processzorszámot minimalizáló algoritmusok)

Iványi Antal, 2011. április 14.

(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 egy 8 órás műszak). 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, $\mathbf{t} = (t_1, t_2, \dots, t_n)$ pedig valós számokat tartalmazó vektor, ahol $t_i \in (0, 1]$ minden $i = 1, 2, \dots, n$ indexre. Osszuk fel a sorozatot minél kevesebb olyan részsorozatra, melyekben az elemek összege legfeljebb 1.

A feladat egydimenziós ládapakolási feladatként is ismert [19, 20, 30, 40]. A feladat NP-teljes [19, 30] (például visszavezethető az összegzési feladatra [19, 30]), ezért a gyakorlati megoldásra közelítő algoritmusokat [20, 26, 28] alkalmaznak.

Az A és B algoritmus relatív hatékonyságát gyakran jellemzik a kiválasztott hatékonysági mérték értékeinek hányadosával, jelen esetben az $A(t)/B(t)$ relatív ládaszámmal. Ennek a hányadosnak a felhasználásával különböző jellemzők definiálhatók, például a relatív ládaszám legrosszabb, várható és legjobb esete.

Ebben a cikkben a legismertebb közelítő (NF, FF, BF, NFD, BFD, FFD) [17, 32, 33, 34, 39] és becslő (MAX, SUM, BIG, UPP) [2, 3, 39] algoritmusok legrosszabb esetére vonatkozó eredményeket foglaljuk össze

Az átlagos eset vizsgálata rendszerint lényegesen nehezebb, míg a legjobb eset vizsgálata rendszerint könnyű.

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

$$D = \bigcup_{i=1}^{\infty} D_i. \quad (1)$$

Legyen \mathcal{A}_{pak} a minden $\mathbf{t} \in \mathcal{D}$ listához egy nemnegatív valós számot hozzárendelő, és így a $\mathcal{D} \rightarrow \mathbb{R}_0^+$ leképezést megvalósító algoritmusok halmaza. Ezeket az algoritmusokat *ládapakoló* algoritmusoknak nevezzük annak ellenére, hogy nem állítanak elő konkrét pakolást, legfeljebb az optimális ládaszám valamilyen becslését adják.

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 tárgyak elhelyezéséhez szükséges és elégséges ládák számát).

Legyen $\mathcal{A}_{köz}$ azon $A \in \mathcal{A}_{pak}$ algoritmusok halmaza, amelyekre $A(\mathbf{t}) \geq \text{OPT}(\mathbf{t})$ minden $\mathbf{t} \in D$ listára, és van olyan $\mathbf{t} \in D$ lista, amelyre $A(\mathbf{t}) > \text{OPT}(\mathbf{t})$. Legyen \mathcal{A}_{becs} azon $E \in \mathcal{A}_{pak}$ algoritmusok halmaza, amelyekre $0 \leq E(\mathbf{t}) \leq \text{OPT}(\mathbf{t})$ minden $\mathbf{t} \in D$ listára, és van olyan $\mathbf{t} \in D$ lista, amelyre $E(\mathbf{t}) < \text{OPT}(\mathbf{t})$.

Legyen F_n azon valós listák halmaza, amelyekre $\text{OPT}(\mathbf{t}) = n$, azaz $F_n = \{\mathbf{t} \mid \mathbf{t} \in D \text{ és } \text{OPT}(\mathbf{t}) = n \ (n = 1, 2, \dots)\}$. Az A és $B \in \mathcal{A}_{köz \cup \mathcal{A}_{becs}}$ algoritmusok $R_{A,B,n}$ hibafüggvényét, $R_{A,B}$ abszolút hibáját (röviden: hibáját) és $R_{A,\infty}$ aszimptotikus hibáját a következőképpen definiáljuk:

$$R_{A,B,n} = \sup_{\mathbf{t} \in F_n} \frac{A(\mathbf{t})}{B(\mathbf{t})}, \quad (2)$$

$$R_{A,B} = \sup_{\mathbf{t} \in F} \frac{A(\mathbf{t})}{B(\mathbf{t})}, \quad (3)$$

$$R_{A,\infty} = \limsup_{n \rightarrow \infty} R_{A,B,n}. \quad (4)$$

Ezek a mennyiségek főleg akkor érdekesek, ha $B \in \mathcal{A}_{opt}$. Ilyenkor az egyszerűség kedvéért a jelölésekből elhagyjuk a B-t, és az $A \in \mathcal{A}_{köz}$, illetve az $E \in \mathcal{A}_{becs}$ 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 [38]. Ma inkább D. S. Johnson megvédett értekezésére [39] hivatkoznak.

1.1. tétel. (Johnson, 1972) *Ha $\mathbf{t} \in F_n$, akkor*

$$n = \text{OPT}(\mathbf{t}) \leq \text{NF}(\mathbf{t}) \leq 2\text{OPT}(\mathbf{t}) - 1 = 2n - 1. \quad (5)$$

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

$$k = \text{OPT}(\mathbf{u}_k) = \text{NF}(\mathbf{u}_k) \quad (6)$$

és

$$k = \text{OPT}(\mathbf{v}_k) \text{ és } \text{NF}(\mathbf{v}_k) = 2\text{OPT}(v_k) - 2. \quad (7)$$

Bizonyítás. Johnson a felső korlátot úgy bizonyította, hogy feltette a korlát ellenkezőjét, miszerint van olyan $L = (t_1, t_2, \dots, T_n)$ lista, amelyre $\text{OPT}(L) = n$ és $\text{FF}(L) \geq 2n$. 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 [32] 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*

$$n = \text{OPT}(\mathbf{t}) \leq A(\mathbf{t}) \leq 2\text{OPT}(\mathbf{t}) - 1 = 2n - 1. \quad (8)$$

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

$$k = \text{OPT}(\mathbf{u}_k) = B(\mathbf{u}_k) \quad (9)$$

és

$$k = \text{OPT}(\mathbf{v}_k) \text{ és } B(\mathbf{v}_k) = 2k - 1. \quad (10)$$

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 [37] könyv 500. oldalán, mint 11.9. tétel.

A NF ládapakoló algoritmus jellemző adatai ismertek.

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

$$n = \text{OPT}(\mathbf{t}) \leq \text{NF}(\mathbf{t}) \leq 2\text{OPT}(\mathbf{t}) - 1 = 2n - 1. \quad (11)$$

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

$$k = \text{OPT}(\mathbf{u}_k) = \text{NF}(\mathbf{u}_k) \quad (12)$$

és

$$k = \text{OPT}(\mathbf{v}_k) \text{ és } \text{NF}(\mathbf{v}_k) = 2k - 1. \quad (13)$$

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

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

$$R_{\text{NF},n} = 2 - \frac{1}{n}, \quad (14)$$

továbbá

$$R_{\text{NF}} = R_{\text{NF},\infty} = 2. \quad (15)$$

3. FF algoritmus

R. L. Graham, 1972-ben [31] 1972-ben megmutatta, hogy az $L = (0.06^7, 0.51^5, 0.1^2, 0.16^3, (0.34^2)^5, (0.51)^{10})$ listára $\text{FF}(L) = 17$, és ha a listát csökkenő sorrendbe rendezzük, akkor az így kapott $L' = (0.51^{10}, 0.34^{10}, 0.16^{10}, 0.10^{10})$ listára $\text{FF}L' = \text{OPT}(L') = 10$.

Graham ebben a cikkében megfogalmazta azt a sejtést, hogy ha $\text{OPT}(L)$ elég nagy, akkor $R_{\text{FF}}/\text{OPT}(L) < 1.7$.

A [32, 33] dolgozatban megmutattuk, hogy az FF relatív hibája tetszőlegesen nagy optimális ládaszám esetén is felveszi az 1,7 értéket.

1.5. tétel. *Iványi, 1984. Ha ...*

A FF és BF ládapakoló algoritmus legrosszabb esetére vonatkozik a következő állítás [35]

1.6. tétel. (Iványi, 1986) *Ha $\mathbf{t} \in F_n$, akkor*

$$\text{FF}(\mathbf{t}) < \frac{7}{4}. \quad (16)$$

1994-ben D. Simchi-Levi [42] egy valamivel gyengébb állításra publikált bonyolultabb bizonyítást (viszont az ő bizonyítása a BF algoritmusra is jó.)

Garey, Graham, Ullman, 1973: [29].

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

$$\text{OPT}(\mathbf{t}) \leq \text{FF}(\mathbf{t}), \text{BF}(\mathbf{t}) \leq 1.7\text{OPT}(\mathbf{t}) + 2. \quad (17)$$

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

$$k = \text{OPT}(\mathbf{u}_k) = \text{FF}(\mathbf{u}_k) = \text{BF}(\mathbf{u}_k), \quad (18)$$

valamint

$$k = \text{OPT}(\mathbf{v}_k) \text{ és } \text{FF}(\mathbf{v}_k) = \text{BF}(\mathbf{v}_k) = \lfloor 1.7k \rfloor. \quad (19)$$

A FF algoritmusra egy erősebb felső korlát is érvényes.

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

$$\text{OPT}(\mathbf{t}) \leq \text{FF}(\mathbf{t}) < 1.7\text{OPT}(\mathbf{t}) + 1. \quad (20)$$

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

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

$$\frac{\lfloor 1.7n \rfloor}{n} \leq R_{\text{FF},n} \leq \frac{\lceil 1.7n \rceil}{n} \quad (21)$$

és

$$\frac{\lfloor 1.7n \rfloor}{n} \leq R_{\text{BF},n} \leq \frac{\lfloor 1.7n + 2 \rfloor}{n} \quad (22)$$

továbbá

$$R_{\text{FF},\infty} = R_{\text{BF},\infty} = 1.7. \quad (23)$$

Ha n osztható tízzel, akkor a (21) egyenlőtlenségben az alsó és felső határok megegyeznek, azaz ebben az esetben $1.7 = R_{FF,n}$.

Baker-Coffman [5], Iványi [32, 33, 34]

Korlát a legrosszabb esetre: [42].

Xia és Tan [43] 2010-ben a következő tételt bizonyították.

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

$$\text{OPT}(\mathbf{t}) \leq \text{FF}(\mathbf{t}) \leq 1.7\text{OPT}(\mathbf{t}) + \frac{7}{10}. \quad (24)$$

4. BF algoritmus

Korlát a legrosszabb esetre: [42]

5. NFD algoritmus

6. FFD algoritmus

Korlát a legrosszabb esetre: [42]

Legjobb korlát a legrosszabb esetre: [44]

7. BFD algoritmus

Korlát a legrosszabb esetre: [42]

8. Átlagos eset

Coffman et al.: [15]

Iványi-Zolotarev: [36]

Bramel et al.: [13]

9. Érdekeségek

Párhuzamos algoritmusok: Anderson et al. [1]

Duális probléma: [25]

Online ládalefedő: [21]

Korlátozott ládapakolás: [27]

Lineáris programozás: [41]

Szimuláció: [38, 39, 45]

@@

Az irodalomjegyzékben **letölthető** megjegyzéssel szereplő művek letölthetők a megadott honlapról, a **digitálisan** megjegyzéssel szereplő művek letölthetők a <http://compalg.inf.elte.hu/~tony/Kutatas/BinPacking/> honlapról, míg a **nyomtatva** megjegyzéssel szereplő műveket nyomtatott formában sikerült megszerezni.

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

Budapest, 2011. április 14.

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] Aszalós Péter: Lower estimation of optimal number of bins. In: *Fourth Conference of program Designers* (Budapest, 1988. június 1–3, szerkesztette [Iványi Antal](#)). ELTE, Budapest, 1988. 81–90. **nyomtatva** [1](#)
- [3] Aszalós Péter, [Iványi Antal](#), Tóth Zoltán, Zsoldos Zsolt: Optimális processzorszám homogén hálózatokban. In: *Programozási rendszerek'88* (Szekszárd, 1988. április 20–23), 77–78. [1](#)
- [4] B. S. Baker. A new proof for the first-fit decreasing bin-packing algorithm. *J. Algorithms*, **6** (1985) 49–70. **nyomtatva**
- [5] 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. [6](#)
- [6] J. Békési, G. Galambos, H. Kellerer. A $5/4$ linear time bin packing algorithm. *JCSS*, **60** (2000) 145–160.
- [7] 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.
- [8] J. O. Berkey, P. Y. Wang. A systolic-based parallel bin packing algorithm. *IEEE Transactions on Parallel and Distributed Systems*, **5** (1994) 769–772.
- [9] 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.
- [10] J. Blazewicz, K. Ecker. A linear time algorithm for restricted bin packing and scheduling problems. *Oper. Res. Lett.*, **2** (1983) 80–83.
- [11] A. Bortfeldt. A heuristic for multiple container loading problems. *OR Spektrum*, **22** (2000) 239–261.
- [12] J. M. Bourjolly, V. Rebetz. An analysis of lower bound procedures for the bin packing problem. *Computers and Operations Research*, **32** (3) (2005) 395–405.

- [13] J. Bramel, W. T. Rhee, Rhee, D. [Simchi-Levi](#): Average-case analysis of the bin-packing problem with general cost structures. *Naval Res. Logist.* **44** (7) (1997) 673–686. MR1471152 (99d:90091) [6](#)
- [14] A. R. Brown. Optimum Packing and Depletion. American Elsevier, New York, 1971. **nyomtatva**
- [15] 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](#)
- [16] 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.
- [17] 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](#)
- [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: *Algoritmuskok. Műszaki Kiadó*, Budapest, 1999 (az 1991-es első angol nyelvű kiadás fordítása). **Informatikai Könyvtár** [1](#)
- [20] [Csirik](#) János: Ládapokolási algoritmusok. Szeged, 1989. 123 oldal. **nyomtatva** [1](#)
- [21] J. Csirik, L. Epstein, Cs. Imreh, A. Levin: On the sum minimization version of the online bin covering problem. *Discrete Appl. Math.* **158** (13) (2010) 1381–1393. MR2651988 **letölthető** [6](#)
- [22] J. [Csirik](#), G. [Galambos](#). An $O(n)$ bin-packing algorithm for uniformly distributed data. *Computing* **36** (1986) 313–319.
- [23] J. [Csirik](#), G. [Galambos](#). On the expected behaviour of the NF algorithm for a [dual bin](#) packing problem. *Acta Cybernetica* **8** (1987) 5–9.
- [24] 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.
- [25] 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](#)

- [26] [Dósa György](#), [Imreh Csanád](#): *Online algoritmusok*. Elektronikus jegyzet. SZTE TTIK, Szeged, 2011 (kézirat). [1](#)
- [27] L. Epstein, Cs. [Imreh](#), A. Levin: Class constrained bin packing revisited. *Theoret. Comput. Sci.* **411** (34–36) (2010) 3073–3089. MR2676854 [letölthető](#) [7](#)
- [28] [Galambos Gábor](#): Ládapakolási feladatok közelítő algoritmusainak legrosszabb-eset vizsgálata. Kandidátusi értekezés tézisei. Szeged, 1991. [nyomtatva](#) [1](#)
- [29] 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. [5](#)
- [30] M. Garey, D. S. [Johnson](#): *Computers and Intractability*. 1979. **Informatikai Könyvtár** [1](#)
- [31] R. L. Graham: Bounds on multiprocessing anomalies and related packing algorithms. In: *Proceedings of the AFIPS Conference* **40**, AFIPS Press, Montvale, 1972, 205–217. [4](#)
- [32] 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](#), [6](#)
- [33] A. [Iványi](#): Estimation of the efficiency of bin-packing algorithms. (Russian) *Problemy Kibernet.* **41** (1984) 253–256. **digitálisan** [1](#), [4](#), [6](#)
- [34] 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](#), [6](#)
- [35] [Iványi A.](#): *Processzorütemezés*. Kézirat. ELTE, Numerikus és Gépi Matematika Tanszék, Budapest, 1986. **nyomtatva** [4](#)
- [36] A. [Iványi](#), V. M. Zolotarev: Probabilistic analysis of the optimal bin packing algorithm. In: ed. by A. Iványi, *Fifth Conference of Program Designers* (Budapest, 1989. augusztus 28–szeptember 1, szerkesztette Iványi Antal). ELTE, Budapest, 1989, 183–198. [6](#)
- [37] [Iványi A.](#): *Informatikai algoritmusok*. ELTE Eötvös Kiadó, Budapest, 2004. Elektronikus: ELTE Informatikai Kar, 1974. [4](#)
- [38] D. S. [Johnson](#): *Near-Optimal Bin Packing Algorithms*. MAC TR-109 számú technikai riport. Massachusetts Institute of Technology, Cambridge, 1973. **nyomtatva** [2](#), [7](#)
- [39] D. S. [Johnson](#): *Near-Optimal Bin Packing Algorithms*. PhD thesis, Massachusetts Institute of Technology, Department of Mathematics, Cambridge, 1973. [1](#), [2](#), [7](#)
- [40] Lovász László, Gács Péter: *Algoritmusok*. Tankönyvkiadó, Budapest, 1989. **ELTE Matematikai Könyvtárában** [1](#)

- [41] L. Mui Ann Chan, D. [Simchi-Levi](#), J. Bramel: Worst-case analyses, linear programming and the bin-packing problem. *Math. Programming, Ser. A.* **83** (2) (1998) 213–227. MR1647849 (99h:90080) [7](#)
- [42] D. [Simchi-Levi](#): New worst-case results for the bin-packing problem. *Naval Research Logistics* **41** (1994) 579–585. **digitálisan** [5](#), [6](#)
- [43] B. Xia, Z. [Than](#): Tighter bounds of the [First Fit](#) algorithm for the bin-packing problem. *Discrete Applied Mathematics* **158** (2010) 1668–1675. **letölthető** [6](#)
- [44] M. Yue: A simple proof of the inequality $FFD(L) \leq (11/9)OPT(L) + 1$ *Acta Math. Applicatae Sinica* **7** (4) (1991) 322–331. **digitálisan** [6](#)
- [45] Zs. Zsoldos: Simulation of bin packing algorithms. In: *Fourth Conference of Program Designers* (Budapest, 1988, június 1–3, szerk. [Iványi](#) Antal). ELTE, Budapest, 1988. 161–170. **nyomtatva** [7](#)

TÁRGYMUTATÓ

B

BF = Best Fit, [1](#)
BFD = Best Fit Decreasing, [1](#)
BIG, [1](#)

F

FF = First Fit, [1](#)
FFD = First Fit Decreasing, [1](#)

L

ládapakoló algoritmus = bin packing
algorithm, [2](#)
legrosszabb eset korlát FFD-re = worst case
bound for FFD, [6](#)

M

MAX = Maximum, [1](#)

N

NF = Next Fit, [1](#)
NFD = Next Fit Decreasing, [1](#)

NP-teljes = NP-complete, [1](#)

P

párhuzamos algoritmus = parallel algorithm,
[6](#)

R

relatív ládaszám = relative number of bins, [1](#)
relatív ládaszám legjobb esete = best case of
the relative number of bins, [1](#)
relatív ládaszám legrosszabb esete = worst
case of the relative number of bins, [1](#)
relatív ládaszám várható értéke = expected
value of relative number of bins, [1](#)

S

SUM = Sum of the sizes, [1](#)

U, Ű

UPP = Upper, [1](#)

MAGYAR NÉVMUTATÓ

A, Á

Aszalós Péter, [8](#)

B

Bramel, Julien, [9](#)

C

Cormen, T. H., [1](#)

CS

Csirik János, [1](#), [9](#)

D

Dósa György, [1](#)

E, É

Epstein, Leah, [9](#), [10](#)

G

Galambos Gábor, [1](#)
Graham, Ronald L., [4](#)

I, Í

Imreh Csanád, [9](#), [10](#)

Iványi Antal, [3](#), [4](#)

L

Leisenson, Ch. E., [1](#)

Levin, Asaf, [9](#), [10](#)

R

Rivest, R. L., [1](#)

Rustín, R., [10](#)

S

Simchi-Levi, David, [9](#)

Stein, C., [1](#)

T

Tóth Zoltán, [8](#)

Z

Zolotarev, V. M., [10](#)

ZS

Zsoldos Zsolt, [8](#)