ԵՐԵՎԱՆԻ ՊԵՏԱԿԱՆ ՀԱՄԱԼՍԱՐԱՆԻ ԳԻՏԱԿԱՆ ՏԵՂԵԿԱԳԻՐ
УЧЕНЫЕ ЗАПИСКИ ЕРЕВАНСКОГО ГОСУДАРСТВЕННОГО УНИВЕРСИТЕТА

Բնական գիտություններ 1, 2005 Естественные науки

Ինֆորմատիկա

УДК 517.19

Խ. Մ. ՀԱՅՐԱՊԵՏՅԱՆ

ԱՐԱԳ ԱԼԳՈՐԻԹՄ ՀԱՏԱԿԱԳԾՄԱՆ ԽՆԴՐԻ ԼՈՒԾՄԱՆ ՀԱՄԱՐ

Ներածություն: Ֆիզիկական նախագծման կարևոր խնդիրներից մեկը հատակագծման (*floorplaning*) խնդիրն է, որը ի հայտ է գալիս այն դեպքում, երբ սխեմայի չափսերը շատ մեծ են, այսինքն այն բաղկացած է միլիոնավոր տրանզիստորներից: Այդ դեպքում սխեման տրոհում են ենթասխեմաների և ամեն մի բլոկի մեջ առանձին լուծում են առաջացող խնդիրները (օրինակ՝ ուղեգծման, տեղաղրման և այլն): Հատակագծման խնդիրը տարբերվում է տեղաղրման խնդրից նրանով, որ որոշում է ոչ միայն էլեմենտների դիրքերը, այլ նաև նրանց ձևերը: Սխեման, որում բոլոր էլեմենտները տեղաղրված են և հայտնի են նրանց ձևերը, կոչվում է հատակագիծ:

Այս հոդվածում դիտարկվում է հատակագծման խնդիրը առանց շղթաները (*net*) հաշվի առնելու: Այսինքն, տրված էլեմենտների բազմության համար պետք է որոշել կամայական էլեմենտի դիրքը և ձևը այնպես, որ այդ էլեմենտները պարունակող նվազագույն ուղղանկյան մակերեսը լինի հնարավորին չափ փոքր: Նվազագույն մակերեսով ուղղանկյունը, որը պարունակում է բոլոր բլոկների ձևերը տեղաղրությունից հետո, անվանենք այդ տեղաղրության սահմանային ուղղանկյուն (ՍՈՒ): Ընդհանուր դեպքում այս խնդիրը NP-լրիվ է [1]: Մենք կնկարագրենք ալգորիթմ (բարդությունը՝ $O(\ln(n))$, n-ը էլեմենտների քանակն է), որը գործնականում տալիս է պիտանի արդյունքներ, իսկ եթե էլեմենտների մակերեսները իրար մոտ են, ապա մեր ալգորիթմը տալիս է օպտիմալ լուծմանը մոտ արդյունք: Անվանենք այս ալգորիթմը «արագ տեղաղրման ալգորիթմ»:

Խնդրի դրվածքը:

ՄՈՒՏՔ: Դիցուք տրված է $B=\{b_1,...,b_n\}$ էլեմենտների (բլոկների) բազմություն: Յուրաքանչյուր b_i բլոկի համար տրված է $S^i=\{s^i_1,...,s^i_{k_i}\}$ ուղղանկյունների բազմությունը, այնպես, որ S^i-ի բոլոր ուղղանկյուններն ունեն միևնույն մակերեսը, իսկ նրանց բարձրությունները իրարից տարբեր են: Անվանենք այդ ուղղանկյունների բազմությունը տրված բլոկի ձևերի բազմություն: Ուղղանկյունը իրենից ներկայացնում է (x, y, w, h) քառյակը, որտեղ x, y-ը ուղղանկյան ձախ-ներքևի գագաթի կոորդինատներն են, իսկ w, h-ը համապատասխանաբար ուղղանկյան լայնությունն ու բարձրությունն են: Կասենք, որ (x_0, y_0) կետը (x, y, w, h) ուղղանկյան ներքին կետ է, եթե միաժամանակ տեղի ունեն հետևյալ անհավասարությունները. $x_0 > x$, $y_0 > y$, $x_0 < x + w$ և $y_0 < y + w$: Կասենք նաև, որ երկու ուղղանկյուններ հատվում են, եթե գոյություն ունի (x_0, y_0) կետ, որը երկու ուղղանկյան համար էլ ներքին կետ է:

ԵԼՔ: Անհրաժեշտ է կամայական b_i բլոկի համար

1. որոշել ձևը իր ձևերի բազմությունից,
2. գտնել ընտրված ձևի (x_i, y_i) կոորդինատները այնպես, որ բլոկների ոչ մի զույգի ձևերը չհատվեն և տեղադրությունից հետո ՍՈՒ-ի մակերեսը լինի մինիմալ:

Ալգորիթմի կառուցվածքը: Մեր կողմից նկարագրվող ալգորիթմը կազմված է 2 մասից.

1. բլոկների բազմության տրոհում,
2. բլոկների ձևերի միավորում:

Առաջին մասում մենք կառուցում ենք բինար ծառ (անվանենք այն հարևանության ծառ՝ T), որի յուրաքանչյուր գագաթին համապատասխանում է բլոկների բազմություն: Նշանակենք $P(i)$-ով i գագաթին համապատասխանող բլոկների բազմությունը, իսկ $A(i)$-ով i գագաթին համապատասխանող բլոկների մակերեսների գումարը: T ծառը կամայական i-ի համար պետք է բավարարի հետևյալ պայմաններին.

ա) $P(i)=B$, եթե i-ն արմատն է,

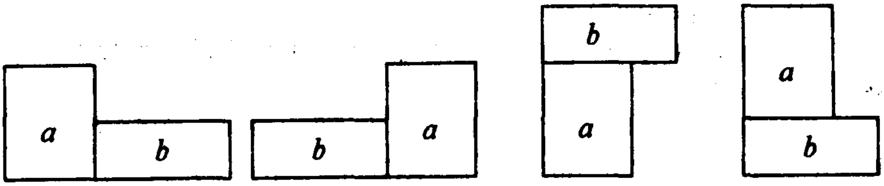
բ) $|P(i)|=1$, եթե i-ն տերև է, այսինքն տերևային գագաթները համապատասխանում են մեկ բլոկի,

գ) $P(i) = P(j) \cup P(k)$ և $P(j) \cap P(k) = \varnothing$, որտեղ i-ն ոչ տերևային գագաթ է, իսկ j, k-ն նրա որդիներն են,

դ) կամայական ոչ տերևային i գագաթի համար $A(j) \approx A(k)$, որտեղ j, k-ն i-ի որդիներն են:

Ինչպես նկարագրված է [2]-ում, Կարպի–Կարմարկարի դ) կետի լուծման մոտավոր ալգորիթմը (այսինքն՝ տրված էլեմենտների բազմությունը, որոնցից յուրաքանչյուրը ունի որոշակի մակերես, տրոհել երկու ենթաբազմությունների այնպես, որ դրանց էլեմենտների մակերեսների գումարները լինեն մոտավորապես իրար հավասար) ունի $O(n \ln(n))$ բարդությունը, որը մինչև այժմ լավագույնն է գործնականում: Ռեկուրսիվորեն կիրառելով Կարպի–Կարմարկարի ալգորիթմը մեր B բազմության, ապա նրա ենթաբազմությունների վրա՝ կստանանք առաջին մասի խնդրի լուծման համար ալգորիթմ: Այժմ դիտարկենք ալգորիթմի երկրորդ քայլը: Առաջին քայլից հետո ունենք բլոկների տրոհման բինար ծառը: Մենք փորձելու ենք գտնել բլոկների տեղադրում՝ օգտագործելով հարևանության ծառը, ընդ որում, եթե i, j-ն միևնույն գագաթի որդիներն են, ապա դրանց համապատասխանող բլոկները կամ բլոկների բազմությունների ՍՈՒ-երը իրար հարևան են: Կասենք, որ երկու ուղղանկյուն իրար հարևան են, եթե նրանք ունեն ընդհանուր կողմ: Այս խնդրի վերաբերյալ կան տարբեր աշխատանքներ, բայց դրանցից ամենահայտնին Ստոկմեյերի ալգորիթմն է [1]:

Դիցուք ունենք a և b ուղղանկյուններ, որոնք իրար հետ կարող ենք միավորվել 4 հնարավոր տարբերակներով (տես նկար 1): Ակնհայտ է, որ ա և բ (ինչպես նաև գ և դ) տարբերակներով միավորելու ժամանակ ստանում ենք միևնույն ուղղանկյունը: Այդ պատճառով կենթադրենք, որ կամայական երկու ողղանկյունների համար գոյություն ունի միավորման երկու տարբերակ՝ հորիզոնական և ուղղաձիգ:



ա բ գ դ

Նկ. 1:

Ստոկմեյերի աշխատանքում [1] դիտարկվում է այս խնդրի մասնավոր դեպքը, երբ ամեն մի բլոկ ունի ընդամենը 2 թույլատրելի ձև և հարևանության ծառում ձևերի միավորման ուղղությունը հայտնի է: Այսինքն, տրված է, թե այդ բլոկների կամ բլոկների բազմությունների ձևերը հորիզոնակա՞ն, թե ուղղաձիգ ուղղությամբ են միավորվելու:

Սահմանում: Կասենք, որ A տեղադրությունը պարունակում է B տեղադրությունը, եթե ՍՈՒ(A)$\le$ՍՈՒ(B), այսինքն ՍՈՒ(A) ուղղանկյան լայնությունը և բարձրությունը մեծ (կամ հավասար) են ՍՈՒ(B) ուղղանկյան լայնությունից և բարձրությունից:

Ձևակերպենք Ստոկմեյերի թեորեմը [1]:

Թեորեմ 1: Եթե բլոկներից յուրաքանչյուրը կարող է ունենալ ամենաշատը 2 ձև, ապա T հարևանության ծառ ունեցող իրարից տարբեր և իրար չպարունակող տեղադրությունների քանակը $\le L(T)+1$, որտեղ $L(T)$-ն T տրոհման ծառի տերևների քանակն է:

Ստոկմեյերի թեորեմի ապացույցում նկարագրվում է նաև ալգորիթմ, թե ինչպես կարող ենք ստանալ այդ բոլոր տեղադրությունները: Այնուհետև մնում է միայն ընտրել այդ բոլոր հատակագծերից այն, որի ՍՈՒ-ն ունի մինիմալ մակերես:

Ստոկմեյերի թեորեմի ընդհանրացումը: Այժմ դիտարկենք ընդհանուր խնդիրը, երբ կամայական b_i բլոկ ունի p_i քանակությամբ ձևեր և հարևանության ծառում ձևերի միավորման ուղղությունը հայտնի չէ: Մենք ցույց կտանք, որ Ստոկմեյերի թեորեմը անմիջապես հետևում է ստորև բերվող թեորեմից:

Թեորեմ 2: Դիցուք տրված է T հարևանության ծառը բլոկների B բազմության համար: Այդ դեպքում T հարևանության ծառ ունեցող իրարից տարբեր և իրար չպարունակող տեղադրությունների քանակը՝

$$M \le \beta^d F - \beta^d L + (\beta^{2d}+2)/3, \qquad (1)$$

որտեղ d-ն ծառի խորությունն է, $L=|B|$, F-ը B բազմության բլոկների ձևերի գումարային քանակն է, $\beta=1$, եթե հարևանության ծառում ձևերի միավորման ուղղությունը հայտնի է, $\beta=2$, եթե անհայտ է:

Հետևանք 1: Երբ $\beta=1$ և ամեն մի բլոկ ունի ամենաշատը 2 ձև, ապա մենք ստանում ենք Ստոկմեյերի թեորեմը: Իրոք.

$$M \le \beta^d F - \beta^d L + (\beta^{2d}+2)/3 = 1^d 2L - 1^d L + (1^{2d}+2)/3 = L+1 :$$

Հետևանք 2: Դիտարկենք ևս մեկ մասնավոր դեպք, երբ հարևանության ծառում ձևերի միավորման ուղղությունը հայտնի է, սակայն բլոկները կարող են ունենալ տարբեր քանակությամբ ձևեր: Այդ դեպքում՝

$$M \le \beta^d F - \beta^d L + (\beta^{2d}+2)/3 = 1^d F - 1^d L + (1^{2d}+2)/3 = F - L + 1 :$$

Այժմ անցնենք թեորեմի ապացույցին:

Ապացույց: Կիրառենք ինդուկցիայի եղանակը: Եթե d=0 (T-ն տերև է), ապա ակնհայտ է, որ

$$M \le \beta^d F - \beta^d L + (\beta^{2d}+2)/3 = \beta^0 F - \beta^0 L + (\beta^0+2)/3 = F :$$

Եթե T-ն տերև է, ապա այդ դեպքում ակնհայտորեն $M = F$:

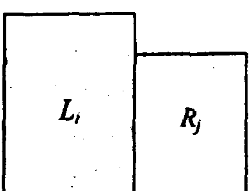
Այժմ ենթադրենք, թե (1) անհավասարությունը տեղի ունի T ծառի արմատի աջ և ձախ որդիների համար: Անհրաժեշտ է ապացուցել, որ այն տեղի ունի նաև T ծառի արմատի համար: Ենթադրենք նաև, որ ծառի գագաթի մի որդին ունի m, իսկ մյուսը՝ k քանակությամբ ձև: Ցույց տանք, որ միավորումից հետո ծառի գագաթը կարող է ունենալ ամենաշատը $\beta(m+k-1)$ ձև: Դիցուք ծառի գագաթի ձախ որդում համադրված է $L_1, L_2, \ldots, L_m$ ձևերի, իսկ աջ որդում՝ $R_1, R_2, \ldots, R_k$ ձևերի հաջորդականությունը: Ենթադրենք, թե նշված հաջորդականությունները ընթանում են բարձրությունների նվազման կարգով: Ակնհայտ է, որ եթե ձևերը ընթանում են ըստ բարձրությունների նվազման կարգի, ապա միաժամանակ կարգավորված են ըստ լայնությունների աճման: Ենթադրելով, որ ձևերի միավորման ուղղությունը հայտնի է (օրինակ, ուղղաձիգ է), դիտարկենք ձևերի միավորման հետևյալ գծային բարդությամբ ալգորիթմ, որի արդյունքում ստացվում է ձևերի $x_1, \ldots, x_t$ հաջորդականությունը:

Քայլ 1: i=1, j=1 և t=0:

Քայլ 2: Քանի դեռ $i \le m$ և $j \le k$, կատարել 3-րդ և 4-րդ քայլերը:

Քայլ 3: x_t-ին վերագրել L_i և R_j ձևերի միավորման արդյունքը (նկ. 2), և t=t+1:

Քայլ 4: Եթե L_i-ի բարձրությունը մեծ է, քան R_j-ինը, ապա i=i+1, եթե R_j-ի բարձրությունն է ավելի մեծ, քան L_i-ինը, ապա j=j+1, եթե բարձրությունները իրար հավասար են, ապա i=i+1 և j=j+1: Սա պայմանավորված է նրանով, որ եթե L_i-ն ավելի բարձր է քան R_j-ն, ապա անիմաստ է միավորել L_i-ն R ցուցակի ավելի փոքր բարձրություն ունեցող ձևի հետ, քանի որ միավորելուց հետո կստանանք նույն բարձրությամբ, սակայն ավելի մեծ լայնությամբ ձև (նկ. 2):



Նկ. 2:

Ակնհայտ է, որ այս ալգորիթմի արդյունքում կարող ենք ստանալ ամենաշատը $m+k-1$ ձև: Եթե ձևերի միավորման ուղղությունը հայտնի չէ, ապա կստանանք ամենաշատը $2(m+k-1)$ ձև, այդ պատճառով ընդհանուր դեպքում տրված գագաթի 2 որդիների ձևերը միավորելով՝ կստանանք ամենաշատը $\beta(m+k-1)$ ձև: Հետևաբար

$$M \le \beta(m+k-1) \le \beta(\beta^{d-1}F_1 - \beta^{d-1}L_1 + (\beta^{2d-2}+2)/3 + \beta^{d-1}F_2 - \beta^{d-1}L_2 +$$
$$+(\beta^{2d-2}+2)/3-1) = \beta^d(F_1+F_2) - \beta^d(L_1+L_2) + 2\,(\beta^{2d-1}+2\beta)/3 - \beta =$$
$$= \beta^d F - \beta^d L + (\beta^{2d}+2)/3,$$

քանի որ $2(\beta^{2d-1}+2\beta)/3-\beta = (\beta^{2d}+2)/3$, երբ $\beta = 1$ կամ 2: ►

«Արագ տեղադրման» ալգորիթմի նկարագրությունը: Նախ նկատենք, որ թեորեմ 2-ի ապացույցն իրենից ներկայացնում է նաև բլոկների ձևերի միավորման ալգորիթմ: Սակայն գործնականում այդ ալգորիթմը կիրառելի չէ, եթե հարևանության ծառի գագաթներում տրոհման ուղղությունը անհայտ է, քանի որ հարևանության ծառի գագաթներին համապատասխանող բլոկների ձևերի բազմության հզորությունը աստիճանային կարգով մեծանում է: Մենք կձևափոխենք այս ալգորիթմը, որպեսզի այն հնարավոր լինի կիրառել գործնականում: Նախ նշենք հետևյալ փաստը. եթե A-ն ու B-ն C գագաթի որդիներն են և A ու B գագաթների ձևերը դասավորվում են բարձրությունների նվազման կարգով, ապա թեորեմ 2-ում նկարագրված ալգորիթմի արդյունքում C գագաթի ձևերը նույնպես կլինեն դասավորված այդ կարգով: Դիցուք A գագաթն ունի a, իսկ B գագաթը՝ b քանակությամբ ձև: Ինչպես նշվել էր վերը, միավորումից հետո C գագաթը կունենա ամենաշատը $2(a+b-1)$ ձև: Դնենք սահմանափակում գագաթի ձևերի վրա: Ենթադրենք, որ յուրաքանչյուր գագաթ կարող է ունենալ ամենաշատը k ձև: Այդ դեպքում C գագաթի ձևերի քանակը հավասար կլինի $\min(2(a+b-1), k)$: Մնում է ցույց տանք, թե ինչպես ընտրել k քանակությամբ ձև C գագաթի ձևերի բազմությունից, երբ $k < 2(a+b-1)$: Ինչպես ասել ենք, C գագաթի ձևերի բազմությունը կարգավորված է ըստ բարձրությունների նվազման: Նշանակենք $t = 2(a+b-1)/k$, $(t > 1)$: Այնուհետև C բազմության ձևերից ընտրենք առաջինից սկսած k հատը՝ t քայլով: Ակնհայտ է, որ արդյունքում կստանանք k կամ $k-1$ ձև: Նկատենք, որ նշված եղանակով վերցված ձևերը նույնպես կարգավորված կլինեն ըստ բարձրության: Այս ձևափոխությամբ ալգորիթմի բարդությունը հավասար կլինի $O(kn)$:

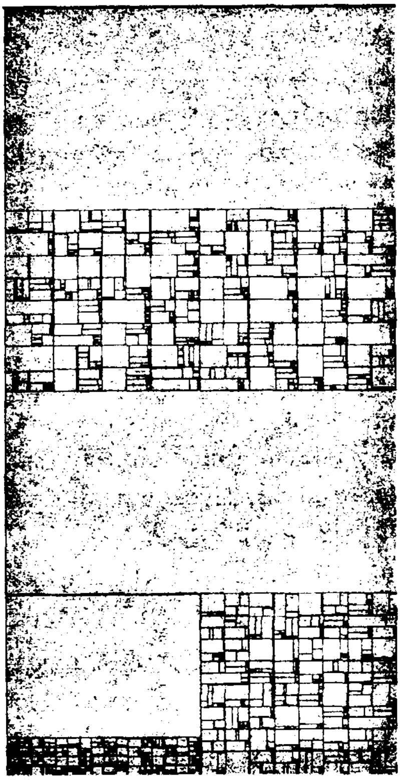
Փորձը ցույց է տվել, որ միջին հաշվով ավելի քիչ ազատ տարածք ենք կորցնում, երբ միավորում ենք մոտավորապես հավասար մակերեսներ ունեցող ձևեր: Այդ պատճառով առաջին քայլում հարևանության ծառը կառուցում ենք այնպես, որ յուրաքանչյուր գագաթի որդիներն ունենան մոտավորապես նույն մակերեսը: Ինչպես նշել էինք, «արագ տեղադրման» ալգորիթմը կազմված է երկու մասից.

1. բլոկների բազմության տրոհում, որի բարդությունը հավասար է $O(n \ln(n))$,

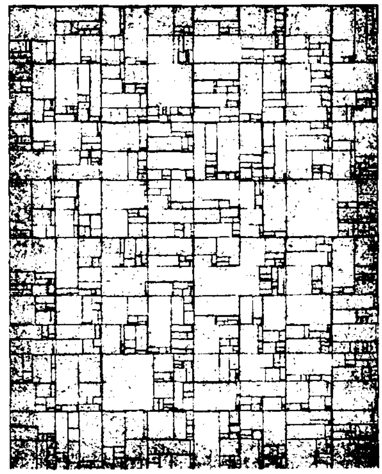
2. բլոկների ձևերի միավորում, որի բարդությունը հավասար է $O(kn)$:

Հետևաբար, ընդհանուր ալգորիթմի բարդությունը հավասար կլինի $O(kn) + O(n \ln(n))$: Այս ալգորիթմը նպատակահարմար է կիրառել հատակագծման ընդհանուր խնդրի լուծման ժամանակ, այսինքն երբ փնտրվում է այնպիսի տեղադրում, որի ժամանակ շղթաների գումարային երկարությունը լինի հնարավորին չափ փոքր (օրինակ, նկ. 3):

ա



բ



Նկ. 3: Ալգորիթմի աշխատանքի օրինակներ. ա) բլոկների քանակը՝ 7704, չօգտագործված մակերեսը՝ 0,76%, աշխատանքի տևողությունը՝ 50վ; բ) բլոկների քանակը՝ 4971, չօգտագործված մակերեսը՝ 1,76%, աշխատանքի տևողությունը՝ 10վ:

Մասնավորապես կարելի է ստանալ հետևյալ շատ կարևոր հարցի պատասխանը. հնարավո՞ր է արդյոք տրված բլոկների բազմությունը տեղադրել տրված ուղղանկյան մեջ, թե՝ ոչ, եթե հնարավոր չէ, ապա գտնվում է նվազագույն մակերեսով ուղղանկյուն, որի մեջ այդ բլոկների բազմությունը հնարավոր է տեղադրել:

Դիսկրետ մաթեմատիկայի ամբիոն *Ստացվել է 08.06.2004*

ԳՐԱԿԱՆՈՒԹՅՈՒՆ

1. **Larry J. Stockmeyer** – Information and Control 57, 1983, (2/3), p. 91–101:
2. Optimal Number Partitioning - Korf (1995) Richard E. Korf Computer Science Department University of California, Los Angeles, Ca. 90024 korf@cs.ucla.edu. August 18, 1995:

Х. М. АЙРАПЕТЯН

БЫСТРЫЙ АЛГОРИТМ ДЛЯ РЕШЕНИЯ ЗАДАЧИ РАЗМЕЩЕНИЯ

Резюме

В данной работе обобщена теорема Стокмеера и представлен быстрый и в практике эффективный алгоритм для решения задачи размещения.

K. M. HAYRAPETYAN

QUICK ALGORITHM FOR SOLVING FLOORPLANNING PROBLEM

Summary

In this article Stockmeyer's theorem is generalized and a quick and practically effective algorithm for solving floorplanning problem is represented.