Splitarea coloanelor dense

Neculai Andrei

Institutul de Cercetări în Informatică – ICI - Bucureşti ,
Centrul de Modelare şi Optimizare Avansată,
Bd. Alexandru Averescu, 8-10, Bucureşti 1, România,
Membru al Academiei Oamenilor de Ştiinţă din România
E-mail: nandrei@ici.ro

În cadrul simplificării modelelor de programare liniară se poate considera şi splitarea coloanelor dense, pe care o vom prezenta în această secţiune.

Experienţa acumulată prin rezolvarea a mii de probleme de programare liniară a confirmat faptul că succesul metodei simplex se bazează pe „două" considerente. Primul este faptul că bazele în cadrul algoritmului simplex sunt rare. Al doilea este faptul că de obicei de-a lungul iteraţiilor simplex bazele unei probleme de programare liniară de mari dimensiuni, prin permutări de linii şi coloane, se pot aduce la o formă aproape triunghiulară. Triunghiularitatea, sau aproape triunghiularitatea bazelor este aspectul esenţial al eficienţei algoritmului simplex..

Pentru a mări sparsitatea (numărul de elemente nenule în matrice) unei probleme de programare liniară ideea este de a *splita (desface, descompune, dedubla) coloanele dense ale matricei problemei într-un număr de coloane care conţin mai puţine elemente nenule.* Aceasta va introduce noi variabile şi restricţii.

După cum vom arăta în această secţiune, acest proces de splitare a coloanelor dense este **invers** procesului de eliminare a restricţiilor de balanţă. Acest fapt se datorează introducerii de restricţii suplimentare, care modelează egalitatea variabilelor adiţionale introduse ca urmare a splitării coloanelor dense.

Să considerăm o problemă de programare liniară în forma standard:

$$\min c^{\mathrm{T}} x \qquad (1)$$

referitor la:

$$Ax = b, \quad x \geq 0,$$

în care matricea A este presupusă că are o structură rară şi este de rang plin. Presupunem de asemenea că un număr relativ mic (3-5%) de coloane ale matricei sunt dense şi că acestea au fost identificate şi permutate astfel încât A se poate partiţiona sub forma: $A = [S \;\; D]$, unde S reprezintă „**partea rară**" a matricei, adică acele coloane sparse (cu număr mic de elemente nenule), iar D reprezintă „**partea densă**", adică acele coloane dense ale matricei A. Desigur, această clasificare a coloanelor ca sparse sau dense este o chestiune de judecată, care depinde de analistul care utilizează această idee a splitării coloanelor. O metodă

este de a fixa un parametru, care să reprezinte un prag, astfel încât orice coloană care are un număr de elemente nenule mai mare decât acest prag să fie considerată densă.

Partiționând corespunzător vectorul x problema (1) se scrie sub forma:

$$\min(c_s^{\mathrm{T}} x_s + c_d^{\mathrm{T}} x_d) \tag{2}$$

referitor la

$$Sx_s + Dx_d = b,$$
$$x_s \geq 0,$$

O metodă de a trata coloanele dense este de a le splita în mai multe coloane cu densități mai mici prin introducerea de variabile suplimentare. Astfel fiecare coloană a lui D este splitată într-un număr de coloane, fiecare dintre acestea conținând un număr de elemente nenule mai mic decât sau egal cu pragul stabilit anterior. Adică elementele nenule ale unei coloane dense se distribuie în mai multe coloane suplimentare, fiecare corespunzând unei noi variabile. Evident, toate aceste variabile trebuie să fie egale. Astfel vom obține o matrice a coeficienților, pentru noua problemă, mai rară decât cea originală.

Pentru a fi mai expliciți, vom considera cazul în care partea densă a lui A (submatricea D) constă numai dintr-o singură coloană d. Adică matricea A se partiționează ca $A = [S \ d]$. Acum, splităm coloana d într-o matrice cu p coloane, F, prin distribuirea în fiecare coloană a lui F a unui număr dat de elemente nenule din d. Relația dintre matricea F şi coloana d este:

$$d = Fe, \tag{3}$$

unde e este un p-vector cu toate elementele egale cu unu. În această situație s-au introdus un număr de $(p-1)$ variabile suplimentare, care sunt toate egale cu variabila corespunzătoare coloanei d din matricea problemei originale. Pentru a exprima egalitatea tuturor acestor variabile suplimentare este necesar să introducem deci $(p-1)$ restricții suplimentare.

Notând cu y vectorul variabilelor suplimentare (incluzând aici şi variabila corespunzătoare coloanei d) şi cu L restricțiile suplimentare, problema de programare liniară cu o coloană densă splitată în p coloane rare este:

$$\min(c_s^{\mathrm{T}} x_s + c_d^{\mathrm{T}} y) \tag{4}$$

referitor la:

$$Sx_s + Fy = b,$$
$$Ly = 0,$$
$$x_s \geq 0, \quad y \geq 0,$$

unde c_s este subvectorul din funcția obiectiv care corespunde părții rare a matricei, iar c_d este nul exceptând o singură componentă egală cu valoarea coeficientului din funcția obiectiv corespunzător coloanei d. Matricea L, care este $(p-1)\times p$-dimensională, numită **matrice de legătură**, are structura:

$$L = \begin{bmatrix} 1 & -1 & & & & \\ & 1 & -1 & & & \\ & & & \ddots & & \\ & & & & 1 & -1 \\ & & & & & 1 & -1 \end{bmatrix}. \qquad (5)$$

Problema (4) are $n+p-1$ coloane şi $m+p-1$ linii, fiind de rang plin pe linii. Este uşor de arătat că noua problemă este echivalentă cu cea originală.

Să considerăm acum cazul general în care matricea A are mai multe coloane dense, adică $A=[S\ D]$, unde partea densă (coloanele dense) a fost concentrată în submatricea D, iar partea rară în S. Presupunem că D are q coloane. Presupunem că a i a coloană a lui D o splităm în p_i coloane rare. Atunci, o splitare a lui D este dată de o matrice F care are m linii şi $\sum_{i=1}^{q} p_i$ coloane care satisface relaţia:

$$D = FE, \qquad (6)$$

unde E este o matrice bloc diagonală, fiecare element diagonal fiind un p_i vector coloană cu toate elementele egale cu unu. Cu aceasta se poate imediat construi matricea $(p_i-1)\times p_i$-dimensională L_i ca în cazul anterior în care am considerat numai o coloană densă, cu structura (5), cu care definim matricea generală de legătură L, care conţine submatricele L_i pe diagonala principală.

Cu aceste construcţii algebrice simple, problema de programare liniară originală se poate rescrie sub forma:

$$\min(c_s^{\mathrm{T}} x_s + c_F^{\mathrm{T}} x_F) \qquad (7)$$

referitor la:

$$Sx_s + Fx_F = b,$$
$$Lx_F = 0,$$
$$x_s \geq 0, \quad x_F \geq 0,$$

unde x_F colectează toate variabilele suplimentare, iar c_F este subvectorul din funcţia obiectiv corespunzător variabilelor din partea densă a matricei problemei. Este uşor de demonstrat că problema (7) este echivalentă cu problema originală.

Această tehnologie de modificare a modelului problemei de programare liniară prin splitarea coloanelor dense comportă anumite critici. Avantajul problemei cu coloane dense splitate este sparsitatea. În aceste circumstanţe este foarte probabil ca reprezentarea inversei bazei sub formă de produs de factori matriceali elementari să fie aproape triunghiulară, minimizând astfel numărul de elemente nenule nou create în procesul de inversare.

Pe de altă parte, dezavantajul metodei de splitare a coloanelor dense îl constituie apariţia elementelor nule în membrul drept al problemei modificate. Aceasta ar putea introduce anumite complicaţii în algoritmul simplex prin apariţia degenerării, care implică, cum am arătat, fenomenul de ciclare sau stagnare. Totuşi, pentru metodele de punct interior metoda de splitare a coloanelor dense este total avantajoasă. Într-adevăr, în cadrul metodelor de punct interior este

necesar să se rezolve sistemul ecuațiilor normale care implică matricea AA^{T}. *Aceasta este una dintre operațiile algebrice cele mai consumatoare de timp, care „încarcă” matricea. De aceea tehnica splitării coloanelor dense concentrează apariția acestor elemente nenule nou create în cadrul înmulțirii numai în anumite „ferestre”, reducând astfel umplerea matricei prin înmulțire.*

Exemplul 1. Fie problema:

$$\min \begin{array}{|c|c|c|c|c|c|c|} \hline 1 & 2 & 1 & 2 & 1 & 2 & 1 \\ \hline \end{array}\ x$$

referitor la:

$$\begin{array}{|c|c|c|c|c|c|c|} \hline 2 & & & -1 & & 8 & 11 \\ \hline & 1 & 1 & & & 9 & 4 \\ \hline 1 & -1 & & 3 & & 10 & 5 \\ \hline & & -2 & & 1 & 3 & 12 \\ \hline \end{array}\ x = \begin{array}{|c|} \hline 23 \\ \hline 81 \\ \hline 74 \\ \hline 19 \\ \hline \end{array}$$

$x_i \geq 0,\ \ i = 1,\ldots,7.$

Splităm ultimele două coloane sub forma:

$$\begin{bmatrix} 8 & 11 \\ 9 & 4 \\ 10 & 5 \\ 3 & 12 \end{bmatrix} = \begin{bmatrix} 8 & & & & 11 & & & \\ 9 & & & & & 4 & & \\ & 10 & & & & & 5 & \\ & & 3 & & & & & 12 \end{bmatrix} \begin{bmatrix} 1 & \\ 1 & \\ 1 & \\ & 1 \\ & 1 \\ & 1 \\ & 1 \end{bmatrix}.$$

Atunci obținem următoarea problemă echivalentă cu problema de mai sus, dar cu coloanele dense splitate.

$$\text{Min} \begin{array}{|c|c|c|c|c|c|c|c|c|c|c|c|} \hline 1 & 2 & 1 & 2 & 1 & 2 & & & 1 & & & \\ \hline \end{array}\ x$$

referitor la:

$$\begin{array}{|c|c|c|c|c|c|c|c|c|c|c|c|} \hline 2 & & & -1 & & 8 & & & 11 & & & \\ \hline & 1 & 1 & & & 9 & & & & 4 & & \\ \hline 1 & -1 & & 3 & & & 10 & & & & 5 & \\ \hline & & -2 & & 1 & & & 3 & & & & 12 \\ \hline & & & & & 1 & -1 & & & & & \\ \hline & & & & & & 1 & -1 & & & & \\ \hline & & & & & & & & 1 & -1 & & \\ \hline & & & & & & & & & 1 & -1 & \\ \hline & & & & & & & & & & 1 & -1 \\ \hline \end{array}\ x = \begin{array}{|c|} \hline 23 \\ \hline 81 \\ \hline 74 \\ \hline 19 \\ \hline 0 \\ \hline 0 \\ \hline 0 \\ \hline 0 \\ \hline 0 \\ \hline \end{array}$$

$x_i \geq 0,\ \ i = 1,\ldots,12.$

Este instructiv să arătăm aici soluțiile acestor probleme. Tabelul 1 conține soluția problemei originale şi a celei cu coloanele dense splitate. Valoarea funcțiilor obiectiv pentru aceste probleme este aceeaşi, egală cu 119.372093. ♦

Tabelul 1.
Soluțiile problemelor.

Problema originală	Problema splitată
$x_1 = 0$	$x_1 = 0$
$x_2 = 34.1163$	$x_2 = 34.1163$
$x_3 = 0$	$x_3 = 0$
$x_4 = 18.6744$	$x_4 = 18.6744$
$x_5 = 3.37209$	$x_5 = 3.37209$
$x_6 = 5.2093$	$x_6 = 5.2093$
	$x_7 = 5.2093$
	$x_8 = 5.2093$
$x_7 = 0$	$x_9 = 0$
	$x_{10} = 0$
	$x_{11} = 0$
	$x_{12} = 0$

După cum am văzut, utilizând această tehnică, orice problemă de programare liniară poate fi transformată într-o alta echivalentă cu un număr suplimentar de linii cu o structură foarte simplă şi cu o densitate a elementelor nenule mult mai mică decât a problemei originale. Evident, eficiența splitării coloanelor dense depinde de euristica de splitare a acestora. O strategie de splitare constă, de exemplu, în fixarea unui „prag” t şi orice coloană cu mai mult de t elemente nenule este splitată într-un număr de coloane suplimentare, fiecare dintre acestea având exact t elemente nenule, exceptând ultima care conține restul elementelor nenule. Elementele nenule din coloanele dense sunt alocate în coloanele rare în ordinea în care apar în coloana densă considerată. Aceasta este o strategie, dar se pot imagina şi altele.

Procesul de splitare a coloanelor dense este **invers** celui de eliminare a restricțiilor de balanță. Într-adevăr, problema splitată are un număr de restricții suplimentare care sunt de balanță, deoarece acestea modelează egalitatea variabilelor suplimentare introduse pentru splitarea coloanelor dense. Observăm imediat că aceste restricții suplimentare din problema cu coloanele dense splitate au fiecare câte două elemente nenule. În aceste condiții, structura matricei problemei originale poate fi reconstruită prin utilizarea tehnicii de eliminare a restricțiilor suplimentare. Aceste procese inverse sunt ilustrate în figura 1.

Teorema 1. *Splitarea coloanelor dense ale unei probleme de programare liniară şi eliminarea restricțiilor de balanță sunt procese inverse.*

Demonstrație. Fie o problemă de programare liniară cu o singură coloană densă. Generalizarea la cazul mai multor coloane dense este imediat. Deci,

eventual permutând coloanele, matricea problemei se scrie sub forma $A = [S \;\; d]$, unde S conține toate coloanele rare ale lui A. Presupunem acum, fără pierderea generalității, că splităm coloana densă în două coloane: $d = [d_1 \;\; d_2]$, adică vom introduce o variabilă suplimentară x_{n+1}, precum şi restricția suplimentară $x_n - x_{n+1} = 0$. Problema astfel transformată este echivalentă cu cea originală. Acum, concentrându-ne atenția asupra ultimei restricții a problemei splitate, care este o restricție de balanță, putem să o eliminăm prin aplicarea la dreapta a unei transformări de eliminare. Aceasta va conduce la reconstrucția coloanei d din coloanele d_1 şi d_2 ale problemei splitate. ■

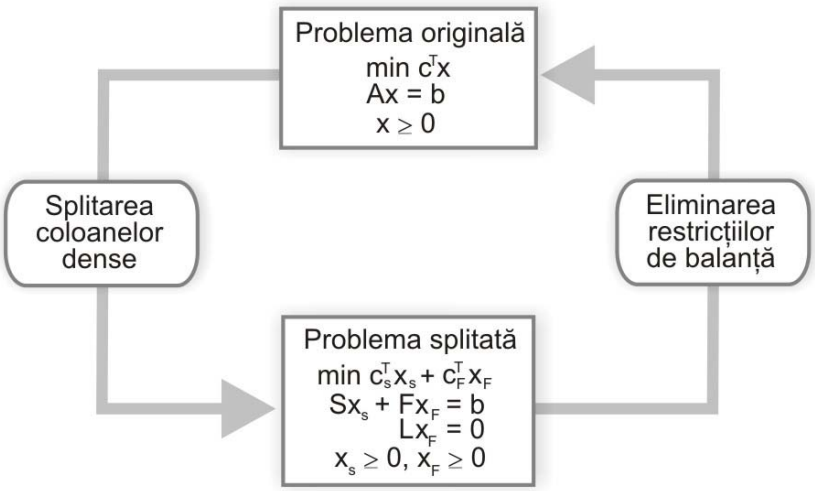

Fig. 1. Procesele inverse de splitare a coloanelor dense şi de eliminare a restricțiilor de balanță.

Importanța acestui rezultat, dincolo de aspectul computațional, este faptul că şi în acest caz se evidențiază un dualism al tehnicilor de calcul în programarea liniară, cel mai bogat domeniu din cadrul programării matematice.

Studiu numeric

În cele ce urmează să considerăm problema de programare liniară 25fv47 pe care am rezolvat-o atât cu algoritmul simplex primal cu forma de eliminare a inversei (FEI) în implementarea MINOS cât şi cu algoritmul de punct interior predictor-corector de ordin superior (PCOS) în implementarea HOPDM. Numărul maxim de elemente nenule din coloanele problemei 25fv47, considerată în acest studiu numeric asupra splitării, este egal cu 21. Pentru această problemă vom aplica tehnica de splitare a coloanelor dense descrisă mai sus, în care parametrul t, pragul care fixează numărul maxim de elemente nenule din coloanele problemei splitate, este constant crescut de la valoarea 1 până la 20.

Tabelul 2 arată caracteristicile problemelor splitate obținute din problema originală 25fv47 în care $t = 1, 2, \ldots, 20$.

Tabelul 2. Caracteristicile problemelor splitate.
t : pragul de splitare, m : numărul de restricții, n : numărul de variabile,
nz : numărul de elemente nenule, dens%: densitatea elementelor nenule.
Problema 25fv47.

t	m	n	nz	dens%
1	9650	10400	28058	0.028
2	4833	5583	18424	0.068
3	3281	4031	15320	0.116
4	2388	3138	13534	0.181
5	2066	2816	12890	0.222
6	1740	2490	12238	0.282
7	1342	2092	11442	0.408
8	1188	1938	11134	0.484
9	1099	1849	10956	0.539
10	1035	1785	10828	0.586
11	995	1745	10748	0.619
12	948	1698	10650	0.662
13	913	1663	10584	0.697
14	890	1640	10538	0.722
15	874	1624	10506	0.740
16	855	1605	10468	0.763
17	849	1599	10456	0.770
18	842	1592	10442	0.779
19	830	1580	10418	0.794
20	822	1572	10402	0.805
21	821	1571	10400	0.806

Observăm că pentru $t = 21$ obținem chiar problema originală (cu coloanele nesplitate) care are 821 de restricții, 1571 de variabile şi 10400 elemente nenule în matricea coeficienților. Pentru $t = 1$, adică dacă pragul de splitare este egal cu unu, obținem o problemă de programare liniară cu 9650 de restricții, 10400 de variabile şi 28058 de elemente nenule în matricea coeficienților, pentru care fiecare coloană are exact un singur element nenul. Vedem imediat că dacă pragul de splitare creşte, atunci obținem probleme splitate din ce în ce mai apropiate de problema originală. Cu cât pragul de splitare este mai mic, cu atât se obțin probleme splitate, echivalente cu problema originală, cu densități din ce în ce mai mici în matricea coeficienților.

În cele ce urmează să rezolvăm problema originală şi variantele ei splitate cu pachetele MINOS, respectiv HOPDM. Tabelul 3 conține numărul de iterații (#iter) şi timpul de calcul (cpu) în secunde pentru rezolvarea acestor probleme.

Tabelul 3. Caracteristicile procesului de rezolvare cu Pachetele MINOS şi HOPDM. Problema 25fv47.

	MINOS		**HOPDM**	
t	#iter	cpu	#iter	cpu
1	6820	20.86	29	3.10
2	6188	10.59	27	1.93
3	5538	7.7	27	1.79
4	6448	6.53	26	1.48
5	5465	5.02	26	1.76
6	6470	5.23	25	2.06
7	6251	4.27	25	1.96
8	6853	4.36	25	1.78
9	6568	4.0	24	1.87
10	6913	4.0	23	1.53
11	7298	4.22	24	1.20
12	7024	3.91	22	1.56
13	6178	3.36	24	1.40
14	6821	3.65	23	1.59
15	6506	3.45	23	1.76
16	**6368**	**3.27**	22	1.09
17	7341	3.80	24	**0.90**
18	6625	3.37	24	1.32
19	7595	3.91	23	1.40
20	7347	3.75	25	1.73
21	7760	**3.93**	24	**1.39**

Din tabelul de mai sus vedem că pentru $t = 16$ obţinem o variantă splitată a problemei originale pentru care pachetul MINOS o rezolvă în 3.27 secunde, faţă de 3.92 secunde cât este necesar pachetului MINOS pentru a rezolva problema originală. În cazul pachetului HOPDM, observăm că pentru $t = 17$ se obţine o variantă de problemă splitată pentru care sunt necesare doar 0.90 de secunde pentru a fi rezolvată de HOPDM.

Este important să notăm că simplificarea problemelor de programare liniară efectuează o anumită chirurgie asupra problemei, care în anumite cazuri o modifică foarte drastic. *Că aceasta este benefic din punctul de vedere al procesului de rezolvare a problemei, este o chestiune de cercetare. Atât pentru algoritmul simplex cât şi pentru metodele de punct interior, experienţa numerică cu aceste procese de simplificare a problemelor de programare liniară este foarte limitată. Încă nu este clar pentru ce clase de probleme de programare liniară tehnicile de eliminare a restricţiilor de balanţă sau de splitare a coloanelor dense conduc la îmbunătăţiri ale comportării algoritmului simplex sau a celor de punct interior.*

(Vezi pachetul SPLIT)

July 20, 2010