Υπολογιστική πολυπλοκότητα αλγορίθμων γραμμικού προγραμματισμού

Διπλωματική Εργασία του φοιτητή Οβελίδη Παρίση
Α.Μ.: 27/11 για το Μεταπτυχιακό στο Τμήμα Εφαρμοσμένης Πληροφορικής

Επιβλέπων Καθηγητής:
Σαμαράς Νικόλαος

Πανεπιστήμιο Μακεδονίας
Εγνατίας 156, Θεσσαλονίκη

Εισαγωγή

- Τα τελευταία χρόνια πολλοί επιστήμονες ασχολήθηκαν με την αναζήτηση βέλτιστων μεθόδων επίλυσης γραμμικών προβλημάτων.
- Εκτός των κλασικών μεθόδων του τύπου PSA (Primal Simplex Algorithm) και άλλων παραλλαγών του, όπως και μια γενικότερη μέθοδο, EPSA (Exterior Point Simplex Algorithm).

Γενική μορφή προβλημάτων γραμμικού προγραμματισμού

- $Ax=b$ όπου ο άγνωστος είναι ο x για τον οποίο θέλουμε να ισχύει
- $x \geq 0$ ενώ τέλος να ελαχιστοποιείται η τιμή
- $c^T x$, όπου c^T είναι ο ανάστροφος του c.

Περιγραφή αλγορίθμου εξωτερικών σημείων

- **Αλγόριθμος** Exterior Point Simplex Algorithm

- **Βήμα 0**. (Αρχικοποίηση). Αρχικός εφικτός διαχωρισμός του πίνακα A σε (B, N). Υπολογισμός των διανυσμάτων και των πινάκων B^{-1}, x_B, w, s_N. Εύρεση των συνόλων $P = \{ j \in N : s_j < 0 \}$ και $Q = \{ j \in N : s_j \geq 0 \}$. Επιλογή ενός αυθαίρετου διανύσματος $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_{|P|}) > 0$, υπολογισμός του s_0 βάση της σχέσης:

  $\mathrm{s}_0 = \sum_{j \epsilon \mathrm{P}} \lambda_j s_j$ και του διανύσματος

  $\mathrm{d}_B = \sum_{j \epsilon \mathrm{P}} \lambda_j h_j$ με $h_j = B^{-1} A_{.j}$

Περιγραφή αλγορίθμου εξωτερικών σημείων

- **Βήμα 1**. (Έλεγχος τερματισμού)

(Έλεγχος Βελτιστότητας). Αν P = Ø, STOP. Το πρόβλημα (P.1) είναι βέλτιστο.

(Επιλογή της μεταβλητής αποχώρησης). Αν $\mathrm{d}_B \geq 0$, STOP.

Αλλιώς αν $s_0 = 0$, το πρόβλημα (P.1) είναι βέλτιστο.

Αλλιώς, επέλεξε τη μεταβλητή αποχώρησης $x_{B[r]} = x_k$ χρησιμοποιώντας τη σχέση:

$$\alpha = \frac{x_{B[r]}}{-\mathrm{d}_{B[r]}} = \min\left\{\frac{x_{B[i]}}{-\mathrm{d}_{B[i]}} : \mathrm{d}_{B[i]} < 0\right\}$$

Αν α = + ∞ Το πρόβλημα (P.1) είναι μη περιορισμένο.

Περιγραφή αλγορίθμου εξωτερικών σημείων

- **Βήμα 2**. (Επιλογή της μεταβλητής εισχώρησης). Υπολογισμός των διανυσμάτων $H_{rP} = (B^{-1})_{r.}A_P$ και $H_{rQ} = (B^{-1})_{r.}A_Q$. Επίσης εύρεση των αναλογιών θ_1 και θ_2 χρησιμοποιώντας τις σχέσεις:

$$\theta_1 = \frac{-s_p}{h_{rp}} = min\left\{\frac{-s_j}{h_{rj}} : \ h_{rj} > 0 \ και \ j \in \mathrm{P}\right\}$$

$$\theta_2 = \frac{-s_q}{h_{rq}} = min\left\{\frac{-s_j}{h_{rj}} : \ h_{rj} > 0 \ και \ j \in \mathrm{Q}\right\}$$

Προσδιορισμός των ευρετηρίων t_1 και t_2 τέτοιο ώστε $P(t_1) = p$ και $Q(t_2) = q$. Αν $\theta_1 \leq \theta_2$, θέσε $l = p$. Αλλιώς, θέσε $l = q$. Η μη βασική μεταβλητή x_l εισάγεται στη βάση.

Περιγραφή αλγορίθμου εξωτερικών σημείων

- **Βήμα 3**. (Περιστροφή). Θέσε B[r] = l. Αν $\theta_1 \leq \theta_2$ θέσε P = P \ { l } και Q = Q U { k }. Αλλιώς, θέσε $Q[t_2] = k$. Χρησιμοποιώντας το νέο διαχωρισμό (B,N), όπου N = (P,Q), ενημέρωση των διανυσμάτων και των πινάκων B^{-1}, x_B, w, s_N. Επίσης, ενημέρωση του διανύσματος $\bar{d}_B$ χρησιμοποιώντας τη σχέση $\bar{d}_B = E^{-1} d_B$ όπου το E^{-1} υπολογίζεται από:

$$E^{-1} = 1 - \frac{1}{\alpha_{pq}}\left(a_q - e_q\right){e_q}^T = \begin{bmatrix} 1 & & -a_{lq}/\alpha_{pq} & & \\ & \ddots & \vdots & & \\ & & 1/\alpha_{pq} & & \\ & & \vdots & \ddots & \\ & & -a_{mq}/\alpha_{pq} & & 1 \end{bmatrix}$$

Αν $l \in \mathrm{P}$ θέσε $\bar{d}_{B[r]} \leftarrow \mathrm{d}_{B[r]} + \lambda_l$. Πήγαινε στο βήμα 1.

Χειρόγραφη επίλυση παραδείγματος (ορισμός προβλήματος)

- A =

612	823	715	833	584
691	607	547	796	938
892	944	582	569	596

- b =
  25
  56
  5776
- c =
  69
  653
  246
  770
  -129
- Eqin = 1
  1
  -1

Χειρόγραφη επίλυση παραδείγματος (επίλυση – μέρος 1ο)

- $B = \begin{matrix} 612 & 823 & 715 \\ 691 & 607 & 547 \\ 892 & 944 & 582 \end{matrix}$ $\quad N = \begin{matrix} 833 & 584 \\ 796 & 938 \\ 569 & 596 \end{matrix}$
- Συνεχίζουμε με τον υπολογισμό των B^{-1}, x_B, w, s_N.
- $B^{-1} = \begin{matrix} -0{,}0032 & 0{,}0039 & 0{,}0003 \\ 0{,}0017 & -0{,}0056 & 0{,}0031 \\ 0{,}0022 & 0{,}0031 & -0{,}0039 \end{matrix}$
- $x_B = (B^{-1}b)^T =$ 2,0053 18,1178 -22,5359
- $w^T = c_B^T - B^{-1} =$ 1,4394 -2,6360 1,1318
- $s_N = c_N^T - wN =$ 1025,2231 828,4008

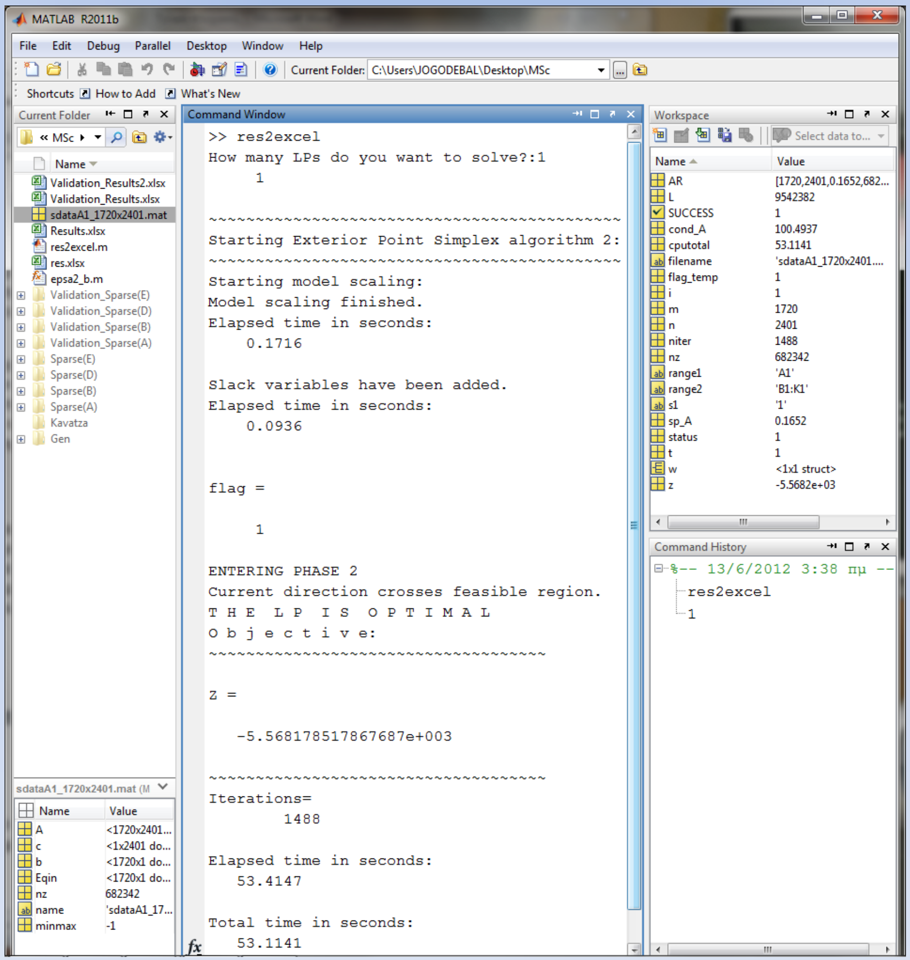
Χειρόγραφη επίλυση παραδείγματος (επίλυση – μέρος 2ο)

- Έπειτα χρειαζόμαστε τα P, Q, το αυθαίρετο διάνυσμα λ και τα s_0 d_B
- P = { j ϵ N : s_j < 0 } = Ø
- Q = { j ϵ N : s_j ≥ 0 } = { 1 , 2 }
- Διαλέγουμε το διάνυσμα λ = $(\lambda_1,\lambda_2,\ldots,\lambda_{|P|}) > 0$ αυθαίρετα.
- λ = {Ø}
- $\mathrm{s}_0 = \sum_{j\epsilon \mathrm{P}} \lambda_j s_j = 0$
- $\mathrm{d}_B = \sum_{j\epsilon \mathrm{P}} \lambda_j h_j = 0$
- Αμέσως μετά πηγαίνουμε στο βήμα 1 όπου βλέπουμε ότι P = Ø οπότε ο αλγόριθμος σταματάει εδώ και το πρόβλημά μας είναι βέλτιστο. Τα x_B και B αποτελούν τη βέλτιστη λύση.

Επίλυση παραδείγματος με τη βοήθεια του αλγορίθμου (περιβάλλον Matlab)

- Άνοιγμα του Matlab
- προσθήκη των προβλημάτων στο φάκελο που περιέχει το αρχείο res2excel.m και epsa2_b.m
- Εκτέλεση της εντολής res2excel
- Απάντηση βάση του πλήθους των προβλημάτων που θα επιλυθούν.

Επίλυση παραδείγματος με τη βοήθεια του αλγορίθμου (περιβάλλον Matlab)



Περιγραφή του dataset

Κατηγορία προβλημάτων	Χαρακτηριστικά	Πλήθος
Random sparse A	mxn, m≤2n, και όλοι οι περιορισμοί είναι τύπου ≤ ή ≥	2500
Random sparse B	mxn, 2m≥n, και όλοι οι περιορισμοί είναι τύπου ≤ ή ≥	1500
Random sparse D	mxm, και όλοι οι περιορισμοί είναι τύπου ≤ ή ≥	1500
Random sparse E	mxm, όλοι οι περιορισμοί είναι τύπου ≤, = ή ≥ και ο χρήστης έχει την δυνατότητα να δώσει όλα τα απαραίτητα στοιχεία μόνος του.	558
	Σύνολο	6058

Επιλογές δημιουργίας dataset

- Sparse A - Sparse D

maximum number of rows
path which the problems are saved
number of different LPs

- Sparse B

maximum number of columns

- Sparse E

number of rows (lower)
number of columns (upper)
Step
Εύρος τιμών για το διάνυσμα eqin
εύρος τιμών για τη μήτρα A
εύρος τιμών για το διάνυσμα c
εύρος τιμών για το διάνυσμα b
τύπο της αντικειμενικής συνάρτησης (0 min/1 max/2 random)
LPs βέλτιστα? (1 yes/2 no)
κάτω όριο για το b1
άνω όριο για το b1

Τρόπος αποθήκευσης αποτελεσμάτων

- ID
- Name
- m
- n
- Density
- Nnz
- L
- Cond(A)
- Niter
- Cpu
- Status
- flag

ID	Name	m	n	density	nnz	L	cond(A)	niter	cpu	status	flag
1	sdataA1000_58x78.mat	58	78	0,1244	563	10303	22,40	83	0,363941163	0	0
2	sdataA1001_1422x2549.mat	1422	2549	0,2100	761362	9655356	64,37	1372	46,20936966	1	1
3	sdataA1002_2504x3164.mat	2504	3164	0,1807	1431930	19248528	185,23	1906	135,037323	1	1
4	sdataA1003_797x1585.mat	797	1585	0,1269	160343	2547665	31,44	962	8,33631134	1	1

Εποπτεία αποτελεσμάτων

Σύνολο προβλημάτων(6058)			
unbounded/infeasible (102)			
optimal (5956)			
Χρόνος=16,60 ημέρες			
Sparse A (2500)	Sparse B (1500)	Sparse D (1500)	Sparse E (558)
unbounded/infeasible (46)	unbounded/infeasible (21)	unbounded/infeasible (32)	unbounded/infeasible (3)
optimal (2454)	optimal (1479)	optimal (1468)	optimal (555)
Χρόνος=6,24 ημέρες	Χρόνος=6,00 ημέρες	Χρόνος=2,48 ημέρες	Χρόνος=1,88 ημέρες

Υπολογιστικό περιβάλλον

CPU	Intel Core 2 Quad Q9300 Yorkfield Penryn (4 Processors)
Ram size	8 GB
L2 Cache size	4x3 MB
L1 Cache size	4x32 KB
HDD	320 GB @ 5400 rpm
Operating System	Microsoft Windows 7 Professional
MATLAB version	7.13.0.564 R2011b 64bit

Βέλτιστο υπόδειγμα κάθε εισόδου για εκτίμηση του niter

Είσοδος	Μη γραμμική παλινδρόμηση	R-Squared
m	Rational Polynomial Funtion 2	13,1%
n	Power (convex)	12,9%
density	Rational Polynomial Funtion 2	30,9%
nnz	Gompertz Growth	5,3%
L	Rational Polynomial Funtion 1	8,2%
cond(A)	Power (convex)	0,3%
flag	Linear	15%

Βέλτιστο υπόδειγμα κάθε εισόδου για εκτίμηση του cpu

Είσοδος	Μη γραμμική παλινδρόμηση	R-Squared
m	Rational Polynomial Funtion 2	28%
n	Quadratic	21,7%
density	Rational Polynomial Funtion 2	6%
nnz	Rational Polynomial Funtion 1	17,9%
L	Rational Polynomial Funtion 1	24,6%
cond(A)	Power (concave)	0,6%
niter	Rational Polynomial Funtion 1	67,7%
flag	Linear	1%

Επικρατέστερα υποδείγματα

- **Για τον υπολογισμό του niter**

niter = -6,22295e+008 + 1,14141 * m + 1,41985 * n + (6,22299e+008 + 1,63507e+009 * density - 3,60596e+009 * density ^ 2) / (1 + 2,62755 * density - 5,79481 * density ^ 2) - 0,000249455 * nnz - 0,00010902 * L + 5,95326e-015 * 'cond(A)' - 4242,97 * flag

- **Για τον υπολογισμό του cpu με χρήση του εκτιμώμενου niter**

cpu = -1585,64 - 0,0647845 * m - 0,137819 * n + 142,879 * density - 0,000335309 * nnz + 5,09833e-005 * L - 2,48515e-016 * 'cond(A)' + (1553,01 + 0,00764701 * niter) / (1 - 3,64243e-005 * niter + 4,01981e-010 * niter ^ 2) + 100,622 * flag

- **Για τον υπολογισμό του cpu χωρίς χρήση του εκτιμώμενου niter**

cpu = -6590,85 + 0,23912 * m + 0,173836 * n + 2145,96 * density + (6294,16 + 0,00195807 * nnz) / (1 + 4,9958e-007 * nnz - 4,04326e-015 * nnz ^ 2) + 8,87086e-006 * L + 1,66727e-015 * 'cond(A)' - 375,855 * flag

Αξιολόγηση υποδειγμάτων

- Υπόδειγμα υπολογισμού niter

	app. niter - niter	|app. niter - niter|
Σύνολο	−472146,1438	781087,9415
Ανά καταγραφή	−780,4068493	1291,054449

- Υπόδειγμα υπολογισμού cpu χωρίς χρήση εκτιμώμενου niter

	app. Cpu - cpu	|app. Cpu - cpu|
Σύνολο	−7735,669948	65444,95443
Ανά καταγραφή	−12,78623132	108,1734784

- Υπόδειγμα υπολογισμού cpu με χρήση εκτιμώμενου niter

	app. Cpu with app. niter- cpu	|app. Cpu with app. niter- cpu|
Σύνολο	-45442,44972	71676,71447
Ανά καταγραφή	-75,11148714	118,4739082

Συμπεράσματα

- Τα υποδείγματα έχουν αισθητό σφάλμα
- Δίνουν όμως μια αρκετά καλή εκτίμηση των niter και cpu για αραιά προβλήματα μικρού και μεσαίου μεγέθους.
- Η μελέτη αυτή μπορεί να επεκταθεί σε σύγκριση εκτιμήσεων άλλων μεθόδων.