ΠΑΡΑΛΛΗΛΟΙ ΑΛΓΟΡΙΘΜΟΙ

ΕΔΙΠ Μαρία Λουκά

Τμήμα Πληροφορικής και Τηλεπικοινωνιών

ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
Εθνικόν και Καποδιστριακόν
Πανεπιστήμιον Αθηνών

23 Μαρτίου 2020

Κεφάλαιο 2
Επιλογή (Selection)

Το πρόβλημα της επιλογής

Δίνεται μια ακολουθία S από n στοιχεία και ένας ακέραιος k, όπου $1 \leq k \leq n$. Να προσδιοριστεί το k μικρότερο στοιχείο στην S.

Υπόθεση: Η ανάπτυξη του παράλληλου αλγόριθμου θα γίνει για το SM SIMD μοντέλο.

Ορισμός

Τα στοιχεία ενός συνόλου A ικανοποιούν μια γραμμική διάταξη $<$ τότε και μόνον τότε αν

(i) για $a, b \in A, a < b, a = b$ ή $b < a$ και

(ii) για $a, b, c \in A$, αν $a < b$ και $b < c$, τότε $a < c$

Παράδειγμα: Το σύνολο των ακεραίων αριθμών ικανοποιεί μια γραμμική διάταξη.

Το πρόβλημα της επιλογής

Βαθμός: Για μια ακολουθία $S = \{s_1, s_2, \cdots, s_n\}$ της οποίας τα στοιχεία είναι και στοιχεία ενός συνόλου με γραμμική διάταξη, ο *βαθμός* ενός στοιχείου s_i του S είναι το πλήθος των στοιχείων του S που προηγούνται του s_i συν 1.

Παράδειγμα

$S = \{9, -4, 3, -6, 7, 0\}$ ο βαθμός του 0 είναι 3.
Αν $s_i = s_j$ τότε το s_i προηγείται του s_j τότε και μόνον τότε αν $i < j$.

Επιλογή

Δίνεται μία ακολουθία $S = \{s_1, s_2, \cdots, s_n\}$, της οποίας τα στοιχεία έχουν παρθεί από ένα γραμμικά διατεταγμένο σύνολο, και ένας ακέραιος $k,\ \{1 \leq k \leq n\}$. Ζητείται ο προσδιορισμός του στοιχείου της S με βαθμό k. Το στοιχείο με βαθμό k θα συμβολίζεται με $s_{(k)}$.

Ένας ακολουθιακός αλγόριθμος

- Αναδρομικός αλγόριθμος
- Προσέγγιση "διαίρε και βασίλευε" (divide and conquer)
- απορρίπτει ένα πλήθος στοιχείων σε κάθε στάδιο

Ένας ακολουθιακός αλγόριθμος

Διαδικασία 1: SEQUENTIAL SELECT (S, k)

• **Βήμα 1**

αν $|S| < Q$ **τότε**
  ταξινόμησε την S και επίστρεψε το $k^{\text{στο}}$ στοιχείο
αλλιώς
  υποδιαίρεσε την S σε $|S|/Q$ υποακολουθίες με Q στοιχεία η κάθε μία (μέχρι $Q - 1$ εναπομείναντα στοιχεία)
τέλος

• **Βήμα 2**

Ταξινόμησε κάθε υποακολουθία και υπολόγισε το μέσο της

• **Βήμα 3**

Κάλεσε την SEQUENTIAL SELECT αναδρομικά για τον υπολογισμό του m, το μέσο των $|S|/Q$ μέσων που βρέθηκαν στο βήμα 2

Ένας ακολουθιακός αλγόριθμος

- **Βήμα 4**

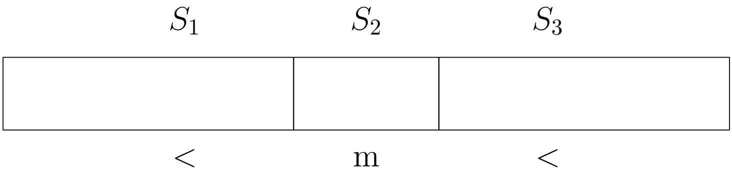
Δημιούργησε τρεις υποακολουθίες $S_1, S_2,$ και S_3 με στοιχεία της S μικρότερα, ίσα και μεγαλύτερα από το m, αντίστοιχα

- **Βήμα 5**

αν $|S_1| \geq k$ **τότε**
    κάλεσε την SEQUENTIAL SELECT αναδρομικά για να βρείς το $k^{\text{στό}}$ στοιχείο του S_1
αλλιώς
    αν $|S_1| + |S_2| \geq k$ **τότε**
        επίστρεψε m
        {διότι το k στοιχείο ανήκει στο S_2}
    αλλιώς
        κάλεσε την SEQUENTIAL SELECT αναδρομικά για να βρείς το $(k - |S_1| - |S_2|)^{\text{στό}}$ στοιχείο του S_3
        {διότι το k στοιχείο ανήκει στο S_3}
    τέλος
τέλος

(Divide and conquer)

Ένας ακολουθιακός αλγόριθμος



Σχήμα: 1. Βήμα 5

Παράδειγμα SEQUENTIAL SELECT (S,k)

Δίνεται η ακολουθία

$$S = \{1, 14, 3, 4, 2, 6, 7, 4, 5, 8, 2, 1, 5, 12, 4, 3, 2, 8, 9, 10, 5, 2, 12\}$$

και ένας ακέραιος k, $1 < k < 23$. Ζητείται να προσδιοριστεί το στοιχείο της ακολουθίας με βαθμό 19, δηλαδή $k = 19$.

Λύση: Επιλέγουμε το Q ώστε να πληρείται η σχέση $\frac{n}{Q} + \frac{3n}{4} < n \Leftrightarrow Q > 4$. Έστω $Q = 5$.

Βήμα 1: Επειδή $|S| = 23 > 5 = Q$, θα υποδιαιρέσουμε την S σε $|S|/Q$ υποακολουθίες, δηλαδή σε 4 υποακολουθίες με 5 στοιχεία και σε 1 υποακολουθία με 3 στοιχεία.

1	14	3	4	2

6	7	4	5	8

2	1	5	12	4

3	2	8	9	10

5	2	12

Παράδειγμα SEQUENTIAL SELECT (S,k)

Βήμα 2: α) Ταξινόμηση κάθε υποακολουθίας β) Βρίσκουμε το μέσο m_i κάθε υποακολουθίας

1	2	3	4	14

$m_1 = 3$

4	5	6	7	8

$m_2 = 6$

1	2	4	5	12

$m_3 = 4$

2	3	8	9	10

$m_4 = 8$

2	5	12

$m_5 = 5$

Βήμα 3: Κλήση της Sequential Select για τον υπολογισμό του μέσου των μέσων.

B1:

m_1	m_2	m_3	m_4	m_5
3	6	4	8	5

$\rightarrow$ B2α:

3	4	5	6	8

Ταξινόμηση των m_i

$\rightarrow$ B2β: $\boxed{m = 5}$

Υπολογισμός μέσου

Βήμα 4: Δημιουργία 3 υποακολουθιών S_1, S_2, S_3 με στοιχεία $<, =, > 5$, αντίστοιχα.

1	3	4	2	1	2	1	4	3	2	2

S_1

5	5	5

S_2

14	6	7	8	12	8	9	10	12

S_3

Παρατηρούμε ότι $|S_1| = 11$, $|S_2| = 3$, $|S_3| = 9$.

Παράδειγμα SEQUENTIAL SELECT (S,k)

Βήμα 5: Παρατηρούμε ότι $k = 19 > |S_1| + |S_2| = 14$ οπότε καλούμε την Sequential Select στην S_3 για να βρούμε το $k_{new}^{(1)} = k - |S_1| - |S_2| = 19 - 14 \Rightarrow k_{new}^{(1)} = 5$, δηλαδή το 5^o στοιχείο της ακολουθίας S_3.

B1: Για $Q = 5$ είναι $|S_3| = 9 > 5 = Q$ οπότε $|S_3|/Q = 9/5 = 1.8$ δηλαδή θα υποδιαιρέσουμε την S_3 σε 1 υποακολουθία με 5 στοιχεία και σε 1 υποακολουθία με 4 στοιχεία.

14	6	7	8	12

8	9	10	12

B2: α)Ταξινόμηση κάθε υποακολουθίας β)Εύρεση μέσου κάθε υποακολουθίας.

6	7	8	12	14

$m'_1 = 8$

8	9	10	12

$m'_2 = 9$

B3: Το μέσο των 8,9 είναι το $m' = 8$

Παράδειγμα SEQUENTIAL SELECT (S,k)

B4: Δημιουργία 3 υποακολουθιών S'_1, S'_2, S'_3 με στοιχεία $<, =, > 8$, αντίστοιχα.

6	7

S'_1

8	8

S'_2

14	12	9	10	12

S'_3

B5: Παρατηρούμε ότι $|S'_1| = 2, |S'_2| = 2, |S'_3| = 5$. Παρατηρούμε ότι $k_{new}^{(1)} = 5 > |S'_1| + |S'_2| = 4$ οπότε καλούμε τη Sequential Select στην S'_3 για να βρούμε το $k_{new}^{(2)} = k_{new}^{(1)} - |S'_1| - |S'_2| = S - 4 \Rightarrow k_{new}^{(2)} = 1$, δηλαδή το 1^o στοιχείο της ακολουθίας S'_3.

Παράδειγμα SEQUENTIAL SELECT (S,k)

Β1΄: Είναι $|S_3'| = 5 = Q$ οπότε ταξινομούμε την S_3' και επιστρέφουμε το 1^o στοιχείο. Είναι

14	12	9	10	12

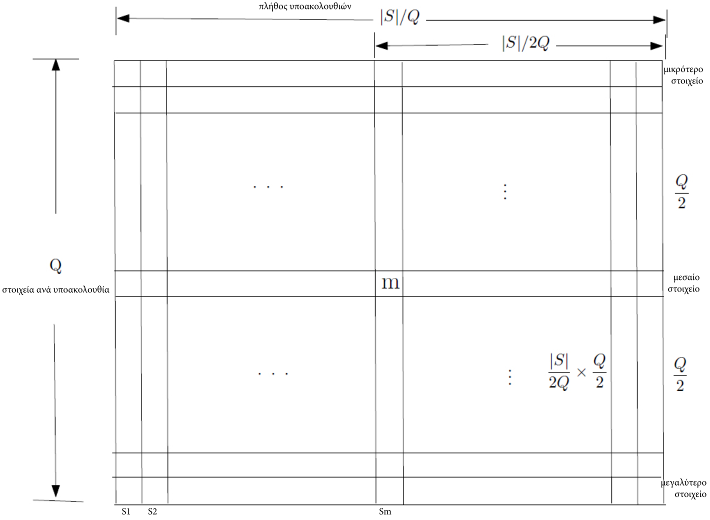
S_3'

9	10	12	12	14

S_3' ταξινομημένη

Το 1^o στοιχείο της S_3' μετά την ταξινόμησή της είναι το 9 και κατά συνέπεια αυτό είναι το ζητούμενο στοιχείο.

Ένας ακολουθιακός αλγόριθμος



Σχήμα: 2. Κύρια ιδέα της διαδικασίας SEQUENTIAL SELECT

Ανάλυση Πολυπλοκότητας της SEQUENTIAL SELECT

Βήμα 1: Ταξινόμηση της S όταν $|S| < Q$ απαιτεί σταθερό χρόνο $O(1)$. Διαφορετικά, υποδιαίρεση της S απαιτεί χρόνο $c_1 n$.

Βήμα 2: Η ταξινόμηση κάθε $|S|/Q$ υποακολουθίας (έχει Q στοιχεία) απαιτεί σταθερό χρόνο.
Για όλες τις υποακολουθίες απαιτείται χρόνος $c_2 n$.

Βήμα 3: $t(n/Q), t(n)$: χρόνος εκτέλεσης της SEQUENTIAL SELECT.

Ανάλυση Πολυπλοκότητας της SEQUENTIAL SELECT

Βήμα 4: Απαιτείται χρόνος $c_3 n$.

Βήμα 5: $(|S|/2Q) \times (Q/2) = |S|/4$ στοιχεία της S (βλέπε σχήμα (2)) είναι μεγαλύτερα ή ίσα με m. Συνεπώς, $|S_1| \leq 3|S|/4$. Όμοια, $|S_3| \leq 3|S|/4$. Συνεπώς μια αναδρομική κλήση της SEQUENCIAL SELECT απαιτεί $t(3n/4)$ χρόνο.

Συνολικά:

$$t(n) = c_4 n + t(n/Q) + t(3n/4)$$

όπου

$$c_4 = c_1 + c_2 + c_3$$

Ανάλυση Πολυπλοκότητας της SEQUENTIAL SELECT

Καθορισμός του Q: Αν επιλεχθεί Q τέτοιο ώστε

$$n/Q + 3n/4 < n$$

τότε οι δυο αναδρομικές κλήσεις στη διαδικασία εφαρμόζονται σε φθίνουσες ακολουθίες. Οποιαδήποτε τιμή του $Q \geq 5$ ικανοποιεί την ανωτέρω ανισότητα. Για $Q = 5$ έχουμε:

$$t(n) = c_4 n + t(n/5) + t(3n/4)$$

Ανάλυση Πολυπλοκότητας της SEQUENTIAL SELECT

Υποθέτοντας :

$$t(n) \leq c_5 n$$

έχουμε

$$t(n) \leq c_4 n + c_5(19n/20)$$

και για $c_5 = 20c_4$:

$$t(n) \leq c_5(n/20) + c_5(19n/20) = c_5 n$$

$$t(n) = O(n)$$

βέλτιστος χρόνος!

Δύο χρήσιμοι αλγόριθμοι

Στο EREW SM SIMD μοντέλο δεν είναι δυνατό να προσπελάσουν δύο επεξεργαστές ταυτόχρονα την ίδια τοποθεσία μνήμης.

(i) **Μετάδοση (broadcasting):** Όλοι οι επεξεργαστές χρειάζονται να διαβάσουν ένα δεδομένο από μια συγκεκριμένη τοποθεσία της κοινής μνήμης.

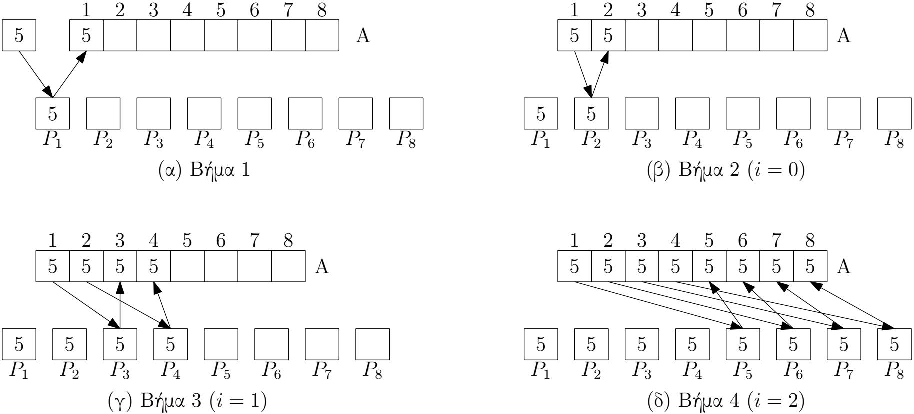
(ii) **Επικοινωνία (όλοι με ένα):** Κάθε επεξεργαστής πρέπει να κάνει υπολογισμούς στα δεδομένα τα οποία υπάρχουν στους άλλους επεξεργαστές και συνεπώς χρειάζεται να λάβει αυτά τα δεδομένα.

Οι δύο αυτές λειτουργίες δεν μπορούν να εκτελεστούν σε ένα βήμα στο EREW μοντέλο και πρέπει να προσομοιωθούν.

Δύο χρήσιμοι αλγόριθμοι

Υπόθεση

N επεξεργαστές $P_1, P_2, \cdots, P_N$ είναι διαθέσιμοι σε ένα EREW SM SIMD υπολογιστή.



(α) Βήμα 1

(β) Βήμα 2 $(i = 0)$

(γ) Βήμα 3 $(i = 1)$

(δ) Βήμα 4 $(i = 2)$

Σχήμα: Διαμοίραση δεδομένου σε 8 επεξεργαστές μέσω της διαδικασίας BROADCAST

Δύο χρήσιμοι αλγόριθμοι

Διαδικασία 2: BROADCAST (D, N, A)

- **Βήμα 1**

Ο επεξεργαστής P_1
(i) διαβάζει την τιμή από την D,
(ii) την αποθηκεύει στην μνήμη του, και
(iii) τη γράφει στην $A(1)$.

- **Βήμα 2**

για $i = 0$ **έως** $(logN - 1)$ **κάνε**
    για $j = 2^i + 1$ **έως** 2^{i+1} **κάνε παράλληλα**
        Ο επεξεργαστής P_j
        (i) διαβάζει την τιμή από την $A(j - 2^i)$,
        (ii) την αποθηκεύει στην μνήμη του, και
        (iii) τη γράφει στην $A(j)$.
    τέλος
τέλος

Δύο χρήσιμοι αλγόριθμοι

Υπολογισμός των αθροισμάτων:

Υπόθεση

P_i έχει το α_i στην τοπική μνήμη.

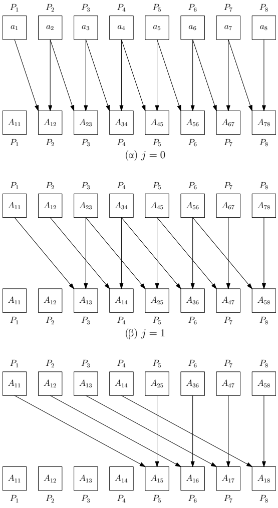
$$P_i := \alpha_1 + \alpha_2 + \cdots + \alpha_i$$

Δύο χρήσιμοι αλγόριθμοι

Διαδικασία 3: ALLSUMS $(\alpha_1, \alpha_2, \cdots, \alpha_N)$

για $j = 0$ **έως** $(logN - 1)$ **κάνε**
  για $i = 2^j + 1$ **έως** N **κάνε παράλληλα**
    Ο επεξεργαστής P_i
    (i) διαβάζει το α_{i-2^j} από το P_{i-2^j} μέσω κοινής μνήμης, και
    (ii) αντικαθιστά το α_i, με το $\alpha_{i-2^j} + \alpha_i$.
  τέλος
τέλος

Δύο χρήσιμοι αλγόριθμοι



(α) $j = 0$

(β) $j = 1$

(γ) $j = 2$

Σχήμα: Υπολογισμός αθροίσματος με χρήση της ALLSUMS

Ένας αλγόριθμος για την παράλληλη επιλογή (EREW SM SIMD)

Υποθέσεις

1. $S = \{S_1, S_2, \cdots, S_n\}, k, 1 \leq k \leq n$
2. $P_1, P_2, \cdots, P_n$ επεξεργαστές
3. Κάθε επεξεργαστής έχει λάβει το n και έχει υπολογίσει το x από την $N = n^{1-x}$, όπου $0 < x < 1$.
4. Κάθε ένας από τους n^{1-x} επεξεργαστές μπορεί να αποθηκεύσει n^x στοιχεία στην τοπική μνήμη του
5. Κάθε επεξεργαστής μπορεί να εκτελέσει τις SEQUENTIAL SELECT, BROADCAST και ALLSUMS.
6. M είναι ένας μονοδιάστατος πίνακας στην κοινή μνήμη μήκους N.

Ένας αλγόριθμος για την παράλληλη επιλογή (EREW SM SIMD)

Διαδικασία 4: PARALLEL SELECT (S, k)

- **Βήμα 1**

αν $|S| \leq 4$ **τότε**
  ο P_1 χρησιμοποιεί το πολύ πέντε συγκρίσεις για να επιστρέψει το $k^{\text{στο}}$ στοιχείο
αλλιώς
  (i) Η ακολουθία S υποδιαιρείται σε $|S|^{1-x}$ υποακολουθίες S_i, μήκους $|S|^x$ η κάθε μία, όπου $1 \leq i \leq |S|^{1-x}$.
  (ii) Η υπακολουθία S_i ανατίθεται στον επεξεργαστή P_i.
τέλος

- **Βήμα 2**

για $i = 1$ **έως** $|S|^{1-x}$ **κάνε παράλληλα**
  (2.1) Ο P_i αποκτά το μέσο m_i, δηλαδή το $\lceil |S_i|/2 \rceil$-οστό στοιχείο της υποακολουθίας του SEQUENTIAL SELECT $(S_i, \lceil |S_i|/2 \rceil)$.
  (2.2) Ο P_i αποθηκεύει το m_i, στο M(i)
τέλος

- **Βήμα 3**

{Το υποπρόγραμμα καλείται αναδρομικά για να αποκτηθεί το μέσο m του M}
PARALLEL SELECT $(M, \lceil |M|/2 \rceil)$.

- **Βήμα 4**

Η ακολουθία S υποδιαιρείται σε τρεις υποακολουθίες:
$L = \{s_i \in S : s_i < m\}$,
$E = \{s_i \in S : s_i = m\}$, και
$G = \{s_i \in S : s_i > m\}$.

- **Βήμα 5**

αν $|L| \geq k$ **τότε**
  PARALLEL SELECT (L,k)
αλλιώς
  αν $|L| + |E| \geq k$ **τότε**
    επίστρεψε m
  αλλιώς
    PARALLEL SELECT ($G, k - |L| - |E|$)
  τέλος
τέλος

Ένας αλγόριθμος για την παράλληλη επιλογή (EREW SM SIMD)

Ανάλυση της Διαδικασίας PARALLEL SELECT

Βήμα 1: Για την εκτέλεση του βήματος αυτού απαιτείται η διεύθυνση αρχής A της ακολουθίας S στην κοινή μνήμη, το μέγεθος $|S|$ και η τιμή k. Οι τιμές αυτές μεταδίδονται σε όλους τους επεξεργαστές με την BROADCAST σε $O(logn^{1-x})$ χρόνο. Αν $|S| \leq 4$, τότε ο P_1 επιστρέφει το k - οστό στοιχείο σε σταθερό χρόνο. Διαφορετικά ο P_i υπολογίζει τη διεύθυνση του πρώτου και του τελευταίου στοιχείου της S_i από τους

$$A + (i-1)n^x$$

και

$$A + in^x - 1$$

σε σταθερό χρόνο. Συνεπώς το βήμα 1 απαιτεί $c_1 logn$ μονάδες χρόνου για κάποια σταθερά c_1.

Ένας αλγόριθμος για την παράλληλη επιλογή (EREW SM SIMD)

Βήμα 2: Η SEQUENTIAL SELECT βρίσκει το μέσο μιας ακολουθίας μήκους n^x σε $c_2 n^x$ μονάδες χρόνου για κάποια σταθερά c_2.

Βήμα 3: Καλείται η PARALLEL SELECT για μια ακολουθία μήκους n^{1-x}. Συνεπώς το παρόν βήμα απαιτεί $t(n^{1-x})$ χρόνο.

Ένας αλγόριθμος για την παράλληλη επιλογή (EREW SM SIMD)

Βήμα 4: Η ακολουθία S χωρίζεται στις υποακολουθίες L, E και G ως εξής:

(i) Το m μεταδίδεται σε όλους τους επεξεργαστές σε $O(logn^{1-x})$ χρόνο με τη χρήση της BROADCAST.

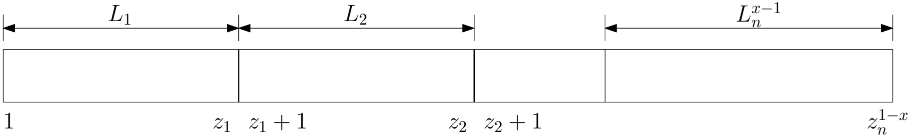
(ii) Κάθε επεξεργαστής P_i χωρίζει την S_i σε τρείς υποακολουθίες L_i, E_i και G_i με στοιχεία μικρότερα, ίσα και μεγαλύτερα από το m, αντίστοιχα. Αυτή η εργασία μπορεί να γίνει σε χρόνο γραμμικό ως πρός το μέγεθος της S_i, δηλαδή σε $O(n^x)$ χρόνο.

(iii) Οι υποακολουθίες L_i, E_i και G_i συγχωνεύονται για να σχηματίσουν τις L, E και G. Η εργασία αυτή μπορεί να γίνει για την L_i. ʼΟμοια και με τον ίδιο χρόνο μπορεί να γίνει για τη συγχώνευση των E_i και G_i, αντίστοιχα. Θέτοντας $\alpha_i = |L_i|$, υπολογίζεται το άθροισμα

$$z_i = \sum_{j=1}^{i} \alpha_j, \quad 1 \leq i \leq n^{1-x}$$

Ένας αλγόριθμος για την παράλληλη επιλογή (EREW SM SIMD)

Τα αθροίσματα αυτά υπολογίζονται σε $O(logn^{1-x})$ χρόνο από τους n^{1-x} επεξεργαστές, χρησιμοποιώντας την ALLSUMS. Όλοι οι επεξεργαστές συγχωνεύουν τις L_i υποακολουθίες τους για τον σχηματισμό της L.



$$z_1 = a_1 = |L_1|, \quad z_2 = a_1 + a_2 = |L_1| + |L_2|$$

Σχήμα: Σχηματισμός της L

Ο επεξεργαστής P_i αντιγράφει την L_i στην L ξεκινώντας από τη θέση $z_{i-1} + 1$, όπου $z_0 = 0$. Η εργασία αυτή μπορεί να γίνει σε $O(n^x)$ χρόνο. Συνεπώς ο χρόνος που απαιτείται για το παρόν βήμα είναι $c_3 n^x$ για κάποια σταθερά c_3.

Ένας αλγόριθμος για την παράλληλη επιλογή (EREW SM SIMD)

Βήμα 5: Το μέγεθος της L που χρειάζεται είναι z_n^{1-x}. Παρομοίως για τα μεγέθη των E και G.

Εν συνεχεία υπολογίζεται ο χρόνος που απαιτείται για κάθε μία από τις δύο αναδρομικές κλήσεις της PARALLEL SELECT. *Επειδή το m είναι το μέσο του M, έπεται ότι $n^{1-x}/2$ στοιχεία είναι μεγαλύτερα από αυτό. Επίσης κάθε στοιχείο του M είναι μικρότερο από $n^x/2$ τουλάχιστον στοιχεία της S.*

Άρα $\frac{n^{1-x}}{2} \cdot \frac{n^x}{2} = \frac{n}{4}$ της L είναι τουλάχιστον μεγαλύτερα ή ίσα του m. Συνεπώς $|L| \leq \frac{3n}{4}$. Όμοια, $|G| \leq \frac{3n}{4}$. Συνολικά το βήμα 5 απαιτεί το πολύ $t(3n/4)$ χρόνο.

Ένας αλγόριθμος για την παράλληλη επιλογή (EREW SM SIMD)

Η προηγούμενη ανάλυση έχει σαν αποτέλεσμα την ακόλουθη αναδρομική εξίσωση για το συνολικό χρόνο:

$$t(n) = c_1 logn + c_2 n^x + t(n^{1-x}) + c_3 n^x + t(3n/4)$$

της οποίας η λύση είναι $t(n) = O(n^x)$ για $n > 4$. Επίσης

$$c(n) = p(n) \cdot t(n) = n^{1-x} \cdot O(n^x) = O(n)$$

Το κόστος αυτό είναι βέλτιστο αφού $\Omega(n)$.

Ένας αλγόριθμος για την παράλληλη επιλογή (EREW SM SIMD)

Παράδειγμα

$$S = \{3, 14, 16, 20, 8, 31, 22, 12, 33, 1, 4, 9, 10, 5, 13, 7, 24, 2, 14, 26, 18, 34, 36, 25, 14, 27, 32, 35, 33\}$$

$$n = 29, \quad k = 21, \quad N = 5$$

Συνεπώς,

$$|S|^{1-x} = 5 \text{ ή } 29^{1-x} = 5 \text{ ή } 1 - x = 0.47796$$

Ένας αλγόριθμος για την παράλληλη επιλογή (EREW SM SIMD)

3	14	16	20	8	31	22	12	33	1	4	9	10	5	13	7	24	2	14	26	18	34	36	25	14	27	32	35	33

(α)

3	14	16	20	8

31	22	12	33	1	4	9

10	5	13	7	24	2

14	26	18	34	36	25

14	27	32	35	33

(β)

M	14	9	7	25	32

(γ)

L											E			G														
3	8	12	1	4	9	10	5	13	7	2	14	14	14	16	20	31	22	33	24	26	18	34	36	25	27	32	35	33

(δ)

16	20	31	22	33

24	26	18	34	36

25	27	32	35	33

(ε)

M	22	26	32

(στ)

L						E	G							
16	20	22	24	18	25	26	31	33	34	36	27	32	35	33

(ζ)

Σχήμα: Επιλογή 21 στοιχείων μιας ακολουθίας με τη διαδικασία PARALLEL SELECT

Ένας αλγόριθμος για την παράλληλη επιλογή (EREW SM SIMD)

Αφού $|L| = 11, |E| = 3, |L| + |E| < k$ αναδρομική κλήση με $S = G$ και $k = 21 - (11 + 3) = 7$.
Επειδή $|G| = 15$ χρησιμοποιούνται $15^{1-x} = 3.6485$ δηλ. 3 επεξεργαστές
Επειδή $|L| = 6, |E| = 1, |L| + |E| = 7 = k$ άρα $m = 26$.

- $p(n) < n^{1/2}$, προσαρμοστικός
- $t(n)$ μικρό, προσαρμοστικό
- $c(n) = p(n) \times t(n)$ βέλτιστο κόστος όταν $c(n) = \Omega(n)$ του ακολουθιακού αλγόριθμου