

ΠΑΡΑΛΛΗΛΗ ΕΠΕΞΕΡΓΑΣΙΑ

ΜΕΤΡΑ ΑΠΟΔΟΣΗΣ

ΕΡΓΑΣΤΗΡΙΟ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ ΥΨΗΛΩΝ
ΕΠΙΔΟΣΕΩΝ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ & ΠΛΗΡΟΦΟΡΙΚΗΣ
ΕΡΓΑΣΤΗΡΙΟ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ ΥΨΗΛΩΝ ΕΠΙΔΟΣΕΩΝ

Η ΑΠΟΔΟΣΗ ΩΣ ΖΗΤΟΥΜΕΝΟ

- Γενικός ορισμός:

Ένας παράλληλος υπολογιστής αποτελείται από ένα πλήθος επεξεργαστικών στοιχείων που επικοινωνούν και συνεργάζονται για να επιλύσουν μεγάλα προβλήματα γρήγορα [Almasi and Gottlieb 1989].

- Ζητήματα:
 - Πόσο μεγάλο είναι το πλήθος των στοιχείων;
 - Πώς πραγματοποιείται η επικοινωνία και η συνεργασία τους;
 - Πως μεταφράζεται αυτό σε απόδοση και ποια είναι αυτή;

ΜΕΤΡΑ ΑΠΟΔΟΣΗΣ

- Βασικό και απόλυτο μέτρο απόδοσης: Ο χρόνος εκτέλεσης. Γενικά, αναφερόμαστε στον χρόνο που χρειάζεται ένα υπολογιστικό σύστημα για να ολοκληρώσει έναν υπολογισμό ή στον χρόνο που απαιτεί ένας αλγόριθμος ή μία διαδικασία για να ολοκληρωθεί.
- Ακολουθιακός χρόνος εκτέλεσης
 - εκτέλεση σε ακολουθιακό σύστημα (σε ένα επεξεργαστή)
- Βέλτιστος ακολουθιακός χρόνος εκτέλεσης
 - ο ακολουθιακός χρόνος του ταχύτερου γνωστού αλγορίθμου (προγράμματος)
- Παράλληλος χρόνος εκτέλεσης
 - εκτέλεση σε παράλληλο / πολυεπεξεργαστικό σύστημα

ΜΕΤΡΑ ΑΠΟΔΟΣΗΣ

- Βασικό μέτρο σύγκρισης της απόδοσης: Η χρονοβελτίωση (**speedup**).

- Ορίζεται ως:

$$\text{speedup} = \text{sequential time} / \text{parallel time}$$

- Απόλυτη χρονοβελτίωση: Σύγκριση με τον βέλτιστο ακολουθιακό χρόνο εκτέλεσης στο συγκεκριμένο σύστημα.
- Χρονοβελτίωση (σχετική): Σύγκριση με την εκτέλεση του παράλληλου αλγορίθμου ακολουθιακά (μια διεργασία/ένα νήμα)

ΒΑΣΙΚΕΣ ΜΕΤΡΙΚΕΣ ΛΕΙΤΟΥΡΓΙΩΝ

- Χρόνος απόκρισης (latency): Ο χρόνος που μεσολαβεί για την ολοκλήρωση μιας λειτουργίας
 - Ρυθμός υπολογισμού ή ρυθμαπόδοση (bandwidth): Ο ρυθμός με τον οποίο εκτελούνται οι λειτουργίες
 - Κόστος (cost): Το αντίκτυπο της εκτέλεσης των λειτουργικών στο χρόνο εκτέλεσης του προγράμματος.
-
- Σε ένα ιδεατό κόσμο:
 - Ο ρυθμός υπολογισμού θα ήταν αντίστροφος του χρόνου απόκρισης
 - Το κόστος θα ήταν το γινόμενο του χρόνου απόκρισης επί του αριθμού των λειτουργιών

ΠΑΡΑΔΕΙΓΜΑ

- Έστω παράλληλο σύστημα όπου:
- Χρόνος απόκρισης μιας εντολής: 100 nsec
 - Άρα ρυθμός 10 εκατομμύρια εντολές το δευτερόλεπτο
- Αν όμως υπάρχει οργάνωση σε αγωγό 10 σταδίων
 - τότε ο ρυθμος είναι 100 εκατομμύρια εντολές ανά δευτ.
- Αν η εφαρμογή εκκινεί μια εντολή κατά μέσο όρο κάθε 200 nsec
 - τότε ο ρυθμος είναι 5 εκατομμύρια εντολές ανά δευτ.
- Αν μια εφαρμογή χρησιμοποιήσει το συγκεκριμένο παράλληλο σύστημα για να εκτελέσει 100 εκατ. εντολές που κυμαίνεται το κόστος της εκτέλεσης τους;
- Μέγιστο: 10 seconds
- Ελάχιστο: 1 second
- Αν υπάρχουν εξαρτήσεις;

ΜΕΤΡΑ ΑΠΟΔΟΣΗΣ

ΧΡΟΝΟΙ ΕΚΤΕΛΕΣΗΣ (1/8)

Ακολουθιακός χρόνος εκτέλεσης

Αν υποθέσουμε ότι ένας ακολουθιακός αλγόριθμος απαιτεί n αριθμητικές πράξεις για να εκτελεστεί, και ότι η κάθε πράξη χρειάζεται μία χρονική μονάδα τ_α για να ολοκληρωθεί. Τότε ο ακολουθιακός χρόνος εκτέλεσης του αλγορίθμου θα είναι:

$$T_{seq}(n) = n * \tau_\alpha$$

τ_α = μέσος χρόνος εκτέλεσης μίας πράξης

Η τ_α είναι ανεξάρτητη του αλγορίθμου και εξαρτάται από τα χαρακτηριστικά του συστήματος.

Ο αριθμός των πράξεων n αναφέρεται ως μέγεθος του προβλήματος ή και ως υπολογιστικός φόρτος.

ΜΕΤΡΑ ΑΠΟΔΟΣΗΣ

ΧΡΟΝΟΙ ΕΚΤΕΛΕΣΗΣ (2/8)

Ρυθμός παραγωγής αποτελεσμάτων, είναι το πλήθος των παραγόμενων αποτελεσμάτων από τις εκτελούμενες πράξεις δια του χρόνου που απαιτήθηκε για τους υπολογισμούς αυτούς. Πρόκειται δηλαδή για μέσο ρυθμό υπολογισμού. Στην περίπτωση του ακολουθιακού αλγορίθμου έχουμε:

$$r_{seq}(n) = n / T_{seq}(n) = n / n * \tau_{\alpha} = 1 / \tau_{\alpha}$$

Και σ' αυτή την περίπτωση το μέγεθος r_{seq} είναι ιδιότητα του συστήματος, ανεξάρτητη του αλγορίθμου και του μεγέθους του προβλήματος n ή του υπολογιστικού φορτίου.

ΜΕΤΡΑ ΑΠΟΔΟΣΗΣ

ΧΡΟΝΟΙ ΕΚΤΕΛΕΣΗΣ (3/8)

Ακολουθιακή εκτέλεση παράλληλου αλγορίθμου

Η εκτέλεση ενός παράλληλου αλγορίθμου σε παράλληλο σύστημα, απαιτεί διάφορες χρονικές επιβαρύνσεις που οφείλονται στην παρουσία πολλαπλών επεξεργαστικών στοιχείων $PE > 1$ (Processing Elements). Τέτοιες επιβαρύνσεις μπορεί να οφείλονται για παράδειγμα, στην ανάγκη ανταλλαγής δεδομένων μεταξύ των επεξεργαστών προκειμένου να ολοκληρωθεί ένα σύνολο πράξεων. Εάν ο παράλληλος αλγόριθμος εκτελεστεί ακολουθιακά, τα επιπλέον στοιχεία που προκαλούν τις επιβαρύνσεις μπορούν να παραλειφθούν και να χρησιμοποιηθούν από τον αλγόριθμο μόνο οι απαραίτητες (ωφέλιμες) πράξεις. Θεωρούμε ότι ο αριθμός των ωφέλιμων πράξεων είναι $n_{//}$. Έτσι η ακολουθιακή εκτέλεση ενός παράλληλου αλγορίθμου απαιτεί χρόνο:

$$T_1(n) = n_{//} * T_a$$

ΜΕΤΡΑ ΑΠΟΔΟΣΗΣ

ΧΡΟΝΟΙ ΕΚΤΕΛΕΣΗΣ (4/8)

Πλεονασμός,

είναι το πηλίκο $n_{//}/n$ που ισούται με το λόγο των δύο ακολουθιακών χρόνων T_1 και T_{seq} . Ισχύει επίσης ότι $n_{//} \geq n$, μπορεί δηλαδή ο αντίστοιχος παράλληλος αλγόριθμος να απαιτεί επιπλέον πράξεις από αυτές που απαιτούνται στον ακολουθιακό, για να πράξει το αντίστοιχο αποτέλεσμα.

$$\text{Πλεονασμός} := \rho = n_{//}/n = T_1(n) / T_{seq}(n) = n_{//} * \tau_\alpha / n * \tau_\alpha \geq 1$$

ΜΕΤΡΑ ΑΠΟΔΟΣΗΣ

ΧΡΟΝΟΙ ΕΚΤΕΛΕΣΗΣ (5/8)

Παράλληλος χρόνος εκτέλεσης

Για την εκτέλεση του παράλληλου αλγορίθμου σε ένα παράλληλο σύστημα με P επεξεργαστές, θα υπολογίσουμε εκτός από το χρόνο εκτέλεσης των “ωφέλιμων” πράξεων και το χρόνο που θα χρειαστεί για τις επιπλέον, απαραίτητες διαδικασίες συντονισμού του αλγορίθμου αλλά και τις διαδικασίες συντονισμού του παράλληλου συστήματος, Χρόνος Επιβάρυνσης. Έτσι ο παράλληλος χρόνος θα είναι το άθροισμα των δύο επιμέρους χρόνων.

$T_{par} = (\text{Ωφέλιμες πράξεις}) * (\text{μέσο χρόνο πράξεων}) + \text{Επιβαρύνσεις} * (\text{μέσο χρόνο επιβαρύνσεων})$

$$T(n, P) = \Omega(n, P) * T_{\alpha} + E(n, P) * T_{\epsilon}$$

ΜΕΤΡΑ ΑΠΟΔΟΣΗΣ

ΧΡΟΝΟΙ ΕΚΤΕΛΕΣΗΣ (6/8)

Παράλληλος χρόνος εκτέλεσης (συνέχεια...)

Στην πραγματικότητα αν θελήσουμε να συγκρίνουμε τις δύο στοιχειώδεις μονάδες χρόνου T_α και T_ε , ισχύει ότι $T_\varepsilon > T_\alpha$. Αν και οι χρόνοι αυτοί δεν είναι απολύτως συγκρίσιμοι μπορούμε να πούμε, για παράδειγμα, ότι ο χρόνος που απαιτείται για τη μεταφορά ενός byte από ένα επεξεργαστή σε έναν άλλο είναι μεγαλύτερος από μία αριθμητική πράξη που μπορεί να εκτελεστεί μεταξύ δύο μεταβλητών του ενός Byte στο ίδιο υπολογιστικό σύστημα.

Όταν εκτελούμε με $P=1$ τότε ισχύει ότι $\Omega(n, 1) := n$ και $E(n, 1) := 0$ δηλαδή,

$T(n, 1) = T_1$ = χρόνος ακολουθιακής εκτέλεσης
παράλληλου αλγορίθμου

ΜΕΤΡΑ ΑΠΟΔΟΣΗΣ

ΧΡΟΝΟΙ ΕΚΤΕΛΕΣΗΣ (7/8)

Χρονοβελτίωση (Speedup),
είναι το πηλίκο

$$S(n, P) = T_{seq}(n) / T(n, P),$$

ισχύει ότι

$$1 \leq S \leq P$$

Η χρονοβελτίωση δείχνει πόσες φορές πιο γρήγορος είναι ο παράλληλος αλγόριθμος από τον αντίστοιχο ακολουθιακό.

ΜΕΤΡΑ ΑΠΟΔΟΣΗΣ

ΧΡΟΝΟΙ ΕΚΤΕΛΕΣΗΣ (8/8)

Αποδοτικότητα (Efficiency),

είναι ένα μέτρο της χρονοβελτίωσης, σε σχέση με το πλήθος των επεξεργαστών που απαιτήσε για να επιτευχθεί:

$$E = S(n, P) / P = T_{seq}(n) / PT(n, P) \leq 1$$

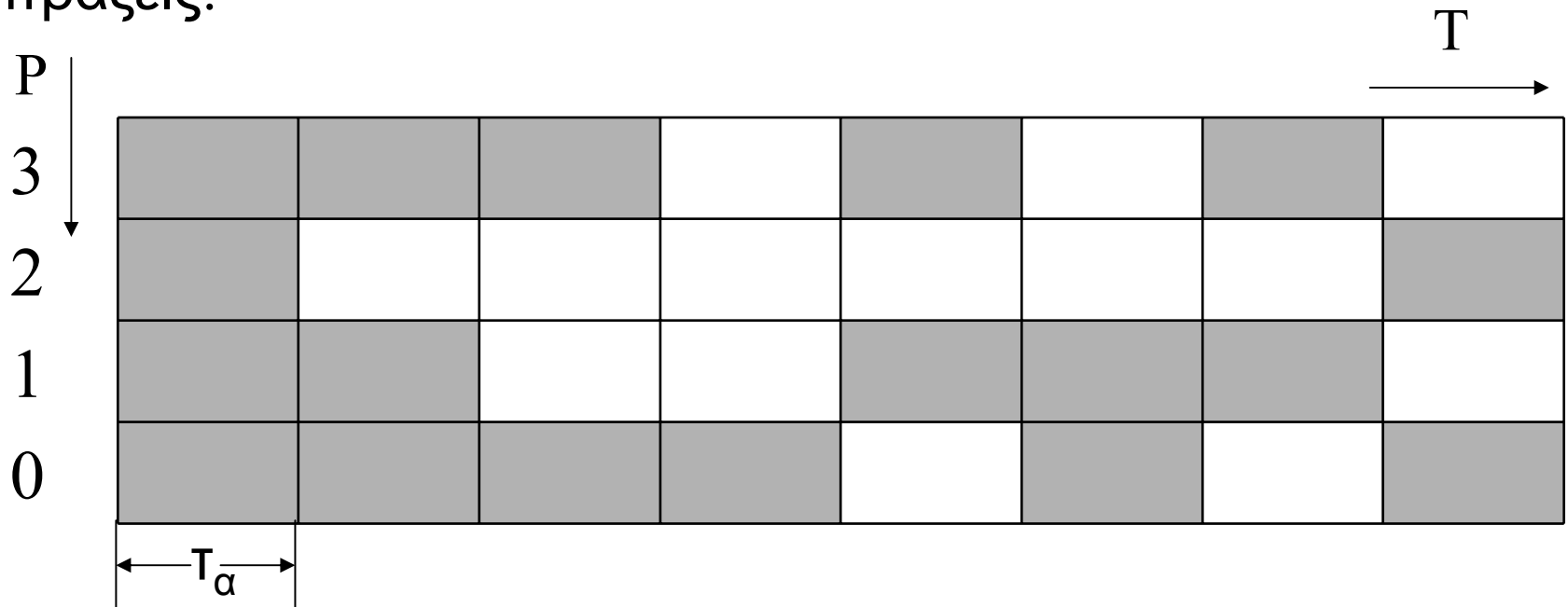
Η ανισότητα ισχύει διότι η χρονοβελτίωση δεν μπορεί να υπερβαίνει την τιμή P (πλήθος επεξεργαστών).

ΜΕΤΡΑ ΑΠΟΔΟΣΗΣ

ΑΞΙΟΠΟΙΗΣΗ (1/3)

Αξιοποίηση,

το μέτρο αυτό εκφράζει το ποσοστό του χρόνου κατά το οποίο οι επεξεργαστές ασχολήθηκαν με ωφέλιμες πράξεις.



ΜΕΤΡΑ ΑΠΟΔΟΣΗΣ

ΑΞΙΟΠΟΙΗΣΗ (2/3)

Το σκιασμένο μέρος του προηγούμενου σχήματος δείχνει την ενασχόληση των επεξεργαστών με ωφέλιμες πράξεις. Τέλεια αξιοποίηση θα είχαμε αν για όλο το χρονικό διάστημα όλοι οι επεξεργαστές έκαναν μόνο χρήσιμες πράξεις, δηλαδή αν όλο το ορθογώνιο εμβαδού $P * T$ ήταν σκιασμένο. Επειδή όμως αυτό δεν συμβαίνει, το ποσοστό για το συγκεκριμένο παράδειγμα είναι:

αξιοποίηση = (εμβαδόν σκιασμένου τμήματος) / (εμβαδόν ορθογωνίου = έργο)

$$18 * T / P * T = 18 / 32$$

Δηλαδή αν ο P_j εκτελεί ωφέλιμες πράξεις για μέρος χρόνου $q_j * T$, όπου $q_j \leq 1$, ενώ για τον υπόλοιπο χρόνο $(1 - q_j) * T$ είναι ανενεργός ή επιβαρύνεται με δραστηριότητες που οφείλονται στην ύπαρξη $P > 1$ επεξεργαστών (π.χ. επικοινωνία), τότε:

ΜΕΤΡΑ ΑΠΟΔΟΣΗΣ

ΑΞΙΟΠΟΙΗΣΗ

$$\text{αξιοποίηση} = \frac{1}{PT} \sum_{j=0}^{p-1} q_j T$$

ενώ

$$PT = \text{έργο ή κόστος} = \sum_{j=0}^{P-1} q_j T + \sum_{j=0}^{P-1} (1 - q_j) T$$

είναι το άθροισμα όλων ανεξαιρέτως των χρονικών διαστημάτων. Σημειώνεται ότι $q_j T = (\text{χρόνος ακολουθιακού υπολογισμού για τον επεξεργαστή } j) = n_j \tau_\alpha$ όπου n_j είναι ο αριθμός των πράξεων που εκτελεί ο επεξεργαστής j .

ΠΑΡΑΔΕΙΓΜΑ

Στο παρακάτω διάγραμμα εμφανίζεται η εκτέλεση ενός παράλληλου αλγόριθμου σε 4 επεξεργαστές. Κατά τα διαστήματα τ_α εκτελούνται χρήσιμες αριθμητικές πράξεις από τους επεξεργαστές για τους οποίους η αντίστοιχη περιοχή είναι σκιασμένη (οι υπόλοιποι επεξεργαστές είναι ανενεργοί) και κατά τα διαστήματα τ_ϵ γίνεται επικοινωνία.

Ο αντίστοιχος ακολουθιακός αλγόριθμος χρειάζεται $n=14$ πράξεις. Αν $\lambda = \tau_\epsilon / \tau_\alpha$, συμπληρώστε τις παρακάτω τιμές συναρτήσει των λ , και τ_α .

Ακολουθιακός Χρόνος = $14\tau_\alpha$ *Παράλληλος Χρόνος* = $6\tau_\alpha + 3\tau_\epsilon = 6 + 3\lambda$

Αξιοποίηση = $16\tau_\alpha / (24\tau_\alpha + 12\tau_\epsilon) = 16 / (24 + 12\lambda)$

Αποδοτικότητα = $14\tau_\alpha / (24\tau_\alpha + 12\tau_\epsilon) = 14 / (24 + 12\lambda)$

