I

° Επιστημονικός Υπολογισμός I °

Κεφάλαιο 1

1. Ορισμός (σελ 6)
2. Κατηγορίες χρήσεων τεχνικών του ΕΥ: (σελ. 6)
   α) προσομοίωση
   β) ανάλυση δεδομένων (ήχος, εικόνα)
   γ) υπολογιστική υποστήριξη των γραφικών
   δ) εφαρμογές που απαιτούν ανταπόκριση σε πραγματικό χρόνο
3. Αφού αναλυθούν οι κώδικες και δούμε ποιά τμήματα (σελ 7) κώδικα έχουν το μεγαλύτερο κόστος διακρίναμε 2 κατηγορίες προβλημάτων:
   α) αυτοί που επιταχύνονται ικανοποιητικά μέσω των βελτιστοποιήσεων που επιτυγχάνει ο μεταφραστής
   β) αυτά τα οποία απαιτούν την χρήση πληροφορίας «υψηλότερου επιπέδου» για την αποτελεσματική τους επιτάχυνση
4. Για να επιταχύνουμε τον κώδικα τι μπορούμε να χρησιμοποιήσουμε; (σελ.7) → αποτελεσματικότερους αλγορίθμους μέσω τ[illegible] αποδοτικής της μαθηματικής [illegible] → αποτελεσματικότερη μετάφραση
5. Επικρατεί η γραμμική άλγεβρα (Ακόμα & στο μετασχηματισμό Fourier) η οποία θα αποτελέσει το όχημα για την παρουσίαση των τεχνικών του ΕΥ [σελ 8 (γενικά) - σελ 18 διαφάνειες
6. Ποιά τα κύρια κριτήρια που χρησιμοποιούνται για την αξιολόγηση των εφαρμογών του επιστημονικού υπολογισμού;
   α) Ακρίβεια των αποτελεσμάτων (Σελ 9)
   β) Ταχύτητα των υπολογισμών
   γ) Κόστος της συνολικής διαδικασίας
7. Όταν μιλάμε για την ακρίβεια των αποτελεσμάτων τι ακριβώς εννοούμε; (σελ 9)
8. Που μπορεί να οφείλονται τα σφάλματα που προκύπτουν επιλύοντας το πρόβλημα στον υπολογιστή; (σελ 9)
9. Εκτίμηση επίδρασης των σφαλμάτων ~~για~~ στην ακρίβεια της λύσης (σελ 9)

10. Στα προβλήματα του ΕΥ ζητάμε μεθόδους που να λύνουν το πρόβλημα στον ελάχιστο δυνατό χρόνο. Πως μπορούμε να επιταχύνουμε τη λύση του προβλήματος & τι δυσκολεύει το εγχείρημά μας; (σελ 10-11)

Υλικό (ταχύτητα επεξεργαστή)
λογισμικό (μεταφραστές
υλοποίηση, αρχιτεκτονική
(παραλληλία), αλγόριθμος

} Καλός συνδυασμός ⟹ Βέλτιστη λύση

Εμπόδια: ① ποικιλία & γρήγορη εξέλιξη των αρχιτεκτονικών των Η/Υ.
② Έλλειψη πρακτικά αποδεκτών μοντέλων για τις νέες αρχιτεκτονικές.

11. Γιατί είναι σημαντικό το κόστος; (σελ 11)

12. Η αξιολόγηση των μεθόδων και των εργαλείων του ΕΥ, γίνεται με βάση την ακρίβεια των αποτελεσμάτων, την ταχύτητα των υπολογισμών, και το κόστος της συνολικής διαδικασίας. (σελ 11)

13. Για ποιους λόγους πολλοί ταχείς αλγόριθμοι που χρησιμοποιούνται στη θεωρία του υπολογισμού, απορρίπτονται ως μη πρακτικοί στον ΕΥ; (σελ 11 - 12) (3 λόγοι)
- Κακή αριθμητική συμπεριφορά
- Απαγορευτικά μεγάλο μέγεθος προβλ.
- Ιδιαίτερα περίπλοκοι

14. Δύο παραδείγματα για την ερώτηση 13 (σελ 12)

15. Στο σχεδιασμό υπολογιστικών εργαλείων πρέπει να λαμβάνεται υπόψιν το περιβάλλον υλοποίησης. Ποια στοιχεία γνωρίζετε της σύγχρονης τεχνολογίας Η/Υ τα οποία έχουν καθοριστική επίδραση στον ΕΥ; (σελ 12)

⟹ Υλικό & Αρχιτεκτονική (RISC) - Γλώσσες, Μεταφραστές & Περιβάλλοντα Επίλυσης Προβλημάτων

16. Τι ρόλο έπαιξε η αρχιτεκτονική Η/Υ στην εξέλιξη του ΕΥ; (σελ 12)

17. Ανέφερε κάποιες αρχιτεκτονικές & πες λίγα λόγια. (σελ 12-13)
- RISC
- παράλληλα συστήματα
- ιεραρχική οργάνωση

18. Τι ξέρετε για την ιεραρχική οργάνωση; (σελ 12 - 13)
& παραδείγματα on-chip μνήμης (→ σελ 13)

19. Οι χρήστες θέλουν ένα σύστημα που να καταλαβαίνει τα μαθηματικά σύμβολα που χρησιμοποιούνται στην διατύπωση των μαθηματικών μοντέλων και που επιλύουν τα αντίστοιχα προβλήματα.
Λύση: Περιβάλλοντα Επίλυσης Προβλημάτων (ΠΕΠ) & Γλώσσες προγραμματισμού υψηλού επιπέδου. (σελ 14)
Ζητάμε: ευκολία προγραμματισμού συγχρόνως με καλή επίδοση.

Κεφάλαιο 2

1. Ορισμός μοντέλου (σελ 20 - 21)
2. Φυσικά μοντέλα → Μαθηματικά μοντέλα → Προσομοίωση
Που βασίζεται το φαινόμενο της προσομοίωσης; (σελ 21)
3. Γιατί τα μαθηματικά μοντέλα πρίν την επεξεργασία σε ψηφιακό Η/Υ πρέπει να διακριτοποιηθούν; Αναφέρατε 3 παραδείγματα (σελ 22 - 23)
4. Οι αλγόριθμοι σχεδιάζονται για ένα «θεωρητικό» μοντέλο Η/Υ αλλά στο τέλος υλοποιούνται μέσω προγραμμάτων σε συγκεκριμένα υπολογιστικά συστήματα που χρησιμοποιούν ανάμεσα στα άλλα και ένα συγκεκριμένο σύστημα κωδικοποίησης για την αναπαράσταση των πραγματικών αριθμών & την διεκπεραίωση της αριθμητικής (σελ 23)
5. Απο την αξιοπιστία των αποτελεσμάτων που παρέχει ένα μοντέλο μπορούμε να αξιολογήσουμε & το βαθμό πιστότητάς του & να τον χρησιμοποιήσουμε για περαιτέρω βελτιώσεις!! (σελ 23)
6. Ποιά είδη μοντέλων σχετίζονται με τον στόχο μας & ποιός είναι αυτός; (σελ 24)
7. Ένα υπολογιστικό μοντέλο περιγράφει μία ιδεατή μηχανή για την οποία μπορούμε να γράψουμε λογισμικό. Γιατί η μηχανή αυτή είναι ιδεατή; (σελ 25)
8. Τι γνωρίζετε για το κλασσικό υπολογιστικό μοντέλο RAM; (σελ 2[illegible]
Απο τι αποτελείται αυτό το μοντέλο; Επαρκεί το μοντέλο αυτό για-

την επίλυση των προβλημάτων του ΕΥ; (σελ 25)

9. Πως γίνεται η ανάλυση του κόστους των αλγορίθμων «σε χρόνο»? (σελ 26)

10. Για ποιούς λόγους το μοντέλο RAM δεν επαρκεί;
(Ποια τα κύρια χαρακτηριστικά των σύγχρονων συστημάτων Η/Υ;) (σελ 25)

11. Μπορεί 2 αλγόριθμοι να επιλύουν μία απλή αλγεβρική πράξη με ακριβώς τον ίδιο αριθμό στοιχειωδών πράξεων ενώ οι επιδόσεις τους είναι εντελώς διαφορετικές όταν εκτελούνται στο ίδιο υπολογιστικό ~~κόλλημα~~ σύστημα (σελ 26)

12. Από τι αποτελείται ένα υπολογιστικό μοντέλο που αυτοαποκρίνεται καλύτερα από το RAM στα σύγχρονα συστήματα Η/Υ; (σελ 26)

13. Ποια τα χαρακτηριστικά του μοντέλου; (σελ 26)

14. Ποια τα πλεονεκτήματα & ποια τα μειονεκτήματα του υπολογιστικού μοντέλου με ιεραρχία μνήμης; (σελ 27)

15. Η ταχύτητα αποτελεί ένα από τα βασικά κριτήρια αξιολόγησης των εργασιών του ΕΥ. Ποιούς μετρητές & δείκτες χρησιμοποιούμε για την ανάλυση της επίδοσης του μοντέλου με ιεραρχία μνήμης; (σελ 27-28)

16. Αν ο αριθμός των δεδομένων εισόδου είναι n και ο αλγόριθμος τα χρησιμοποιεί όλα για να υπολογίσει τουλάχιστον ένα αποτέλεσμα, τότε $\Phi_{min} \geq n+1$ (σελ 28) + Απόδειξη.

17. Τι γνωρίζετε για τον ~~μετρητή~~ δείκτη MFlop/s (σελ 28)

18. Ποιό το κόστος T εκτέλεσης μιας υλοποίησης; (σελ 28)

19. Για ποιούς λόγους πρέπει να είμαστε επιφυλακτικοί με το μοντέλο με ιεραρχία μνήμης; (σελ 29)

20. Τι είναι η διαδικασία προμεταφοράς & με ποιους τρόπους μπορεί να γίνει; (σελ 29)

Κεφάλαιο 3

1. Σε ποιούς λόγους μπορεί να οφείλεται ένα σφάλμα ή «η απώλεια πληροφορίας»? (σελ 33)

Ερωτήσεις Θεωρίας

Κεφάλαιο 1

1. Ποιος ο σκοπός του Επιστημονικού Υπολογισμού;
2. Ποιες οι πιο σημαντικές κατηγορίες χρήσεων Τεχνικών του ΕΥ; | κατηγορί εφαρμογ
3. Κατηγορίες προβλημάτων ΕΥ (διαφάνειες).
   - ▷ Άμεσα Προβλήματα $Tx=y$ T, x γνωστά $y=;$
   - ▷ Αντίστροφα Προβλήματα $Tx=y$ T, y —"— , $x=;$
   - ▷ Προβλήματα Ταυτοποίησης $Tx=y$ x, y —"— , $T=;$

   μικρή / δυσκολία επίλυσης / μεγάλη

4. Για την αντιμετώπιση των μεγάλων υπολογιστικών προβλημάτων πρέπει πρώτα να αναλυθούν προσεκτικά οι υπάρχοντες κώδικες (να φανεί ξεκάθαρα ποια τμήματα του κώδικα έχουν το μεγαλύτερο κόστος)
   Διακρίνουμε 2 κατηγορίες προβλημάτων: (1.) Αυτά που επιταχύνονται ικανοποιητικά μέσω των βελτιστοποιήσεων που επιτυγχάνει ο μεταφραστής και (2.) Αυτά που απαιτούν τη χρήση παρέμβασης «υψηλότερου επιπέδου» για την αποτελεσματική τους επιτάχυνση.
   Πχ: μαθηματικής παρέμβασης για την κατασκευή ταχύτερου αλγορίθμου.
   Τι πρέπει να κάνουμε για να επιταχύνουμε ένα κώδικα;

5. Τι ονομάζουμε υπολογιστικούς πυρήνες; (3 παραδείγματα)
   - πολ/σμός μητρώου x διάνυσμα
   - ταχύς μετασχηματισμός Fourier
   - γεννήτρια τυχαίων αριθμών

6. Ποια τα κύρια κριτήρια για την αξιολόγηση των εργαλείων του ΕΥ;
   α) Ακρίβεια των αποτελεσμάτων β) Ταχύτητα των υπολογισμών
   γ) Κόστος της συνολικής διαδικασίας

7. Τι εννοούμε λέγοντας ακρίβεια των αποτελεσμάτων;

8. Που οφείλονται τα σφάλματα που προκύπτουν επιλύοντας ένα πρόβλημα;

① δεδομένα

② διακριτοποίηση των εξισώσεων

③ διακριτοποίηση των πραγματικών αριθμών σε Η/Υ και στις περιορισμένης ακρίβειας αριθμητικές πράξεις με αυτά

④ Στον περιορισμό της πεπερασμένης επανάληψης στις επαναληπτικές μεθόδους για την εύρεση αποτελέσματος πχ. εύρεση ριζών μη γραμμικής εξίσωσης, ιδιοτιμών πίνακα. Στις μεθόδους αυτές βασιζόμαστε στη σύγκλιση στον υπολογιστή όμως είμαστε υποχρεωμένοι να σταματήσουμε τις επαναλήψεις όταν κάποιος δείκτης μέτρησης σφάλματος γίνει αρκετά μικρός

⑤ Σε δεδομένα ή ενδιάμεσα αποτελέσματα τα οποία δεν έχουν προβλεφθεί από τη λογική της μεθόδου επίλυσης

9. Τι εννοούμε όταν λέμε ταχύτητα των υπολογισμών;

10. Τι εννοούμε όταν λέμε κόστος της αριθμητικής διαδικασίας;
βαθμός προσέγγισης & κόστος → άμεσα συνυφασμένα.
Καθώς οι πόροι είναι πεπερασμένοι και περιορισμένοι, η χρήση τους έρχεται με κόστος το οποίο έχει ληφθεί υπόψη στην επιλογή των εργαλείων και των μεθόδων
Σχεδιάζοντας μεθόδους που έχουν γενική χρήση και είναι ανοικτές και επεκτάσιμες έχουμε ταχύτερη απόσβεση του κόστους και μειώνουμε το κόστος προσαρμογής της μεθόδου σε νέες ανάγκες

11. Για ποιους λόγους πολλοί ταχείς αλγόριθμοι που χρησιμοποιούνται στη θεωρία του υπολογισμού απορρίπτονται ως μη πρακτικοί; (3)

2. Πως ορίζεται το απόλυτο & πως το σχετικό σφάλμα; (σελ. 34)
3. Με ποιους τρόπους μπορούμε να χαρακτηρίσουμε το σφάλμα σε ποσότητες σαν τα διανύσματα & τα μητρώα; (σελ. 34-35)
4. Τα στοιχεία του συστήματος α.κ.υ. f είναι υποσύνολο του R μορφής: $y = \pm m \times \beta^{e-t}$ (σελ. 36)

Σταθερές:
- β: βάση του f ($\beta=2$)
- t: αριθμός των ~~[illegible]~~ ψηφίων που χρησιμοποιούνται για την κωδικοποίηση του m στη βάση β.
- m: ακέραιος («ακέραιο μέρος του y»)
- $m/\beta^t < 1$ δεκαδικός («δεκαδικό μέρος του y» ή mantissa)

5. Χαρακτηριστικά του συνόλου f. (σελ. 36-37)
6. Ποιοι οι παράμετροι του Wilkinson; (σελ. 37).
7. Ποια τα ιδιαίτερα χαρακτηριστικά των αριθμών κινητής υποδιαστολής; (σελ. 37)
8. Πως ορίζεται η υπερχείλιση, υποχείλιση και η στρογγύλευση; (σελ. 37)
9. Πως λειτουργεί η απεικόνιση στην περίπτωση όπου $x \in G$ και $x \notin f$ (Στρογγύλευση). (σελ. 38)
(Μέθοδοι Στρογγύλευσης 4)
10. Τι γνωρίζεται για την κανονικοποίηση & το κρυμμένο bit; (σελ. 39)
11. Ποιες οι ιδιότητες του συστήματος α.κ.υ.; (σελ. 39-40)
12. Για κάθε σύστημα α.κ.υ ισχύει ότι υπάρχει ένας (μικρός) αριθμός ε για τον οποίο ισχύει ότι κάθε α.κ.υ x τ.ω $0 \le x < \varepsilon$ ικανοποιεί τη σχέση $1 \mp x = 0$ (σελ.
13. Έστω ότι για κάθε ακυ, έστω x μπορούμε να ορίσουμε τον αμέσως μεγαλύτερό του α.κ.υ. έστω x^+ δηλ. του α.κ.υ εκείνο για τον οποίο ισχύουν ότι $x < x^+$ και ότι δεν μεσολαβεί κανένας άλλος α.κ.υ μεταξύ τους. Έτσι για κάθε α.κ.υ μπορούμε να ορίσουμε την απόστασή του, έστω $\delta(x, x^+) = x^+ - x$, από τον αμέσως επόμενο α.κ.υ x^+. (σελ. 41)
14. Ορισμός του «ε της μηχανής» (σελ. 41)
!! 15. Ορισμός μονάδας στρογγύλευσης (σελ. 40)

16. Η μονάδα στρογγύλευσης είναι $u = 2^{-t}$. Άρα $\varepsilon_M = 2u$ (σελ. 41)

17. Γιατί το σύστημα δεν είναι κλειστό για τις συνηθισμένες πράξεις του R; Δώσε παράδειγμα. (σελ. 43)

18. Για ποιους λόγους συνέβαινε μία στοιχειώδης πράξη με αριθμούς κινητής υποδιαστολής να οδηγεί σε αδικαιολόγητα μεγάλο σφάλμα (σελ. 44)

19. Τι λέμε «ακριβή στρογγύλευση»; (σελ. 44)

20. Τι είναι το ψηφίο προστασίας και ποιες οι συνέπειες της μη χρήσης του; (σελ. 45)

21. Ποιά τα αξιώματα της προσθέσεως & ποιά του πολ/σμου; για το R (σελ. 45)

22. ——— " ——— για το F (σελ. 45-46)

23. Ποιες οι βασικές σχέσεις που ισχύουν για το σφάλμα που μπορεί να προκύψει μετά από την εκτέλεση μίας από τις συνηθισμένες πράξεις της αριθμητικής στο μοντέλο α.κ.υ; (σελ. 47)

24. Ποιό το μοντέλο για τη μετάδοση του σφάλματος; (σελ. 47)

Διακριτοποιηση των εξισωσεων

$$x_{i+1} = x_i + \frac{f(x_i)}{f'(x_i)} \longrightarrow \text{Newton}$$

$$x_{i+1} = x_i + \frac{f(x_i)\,(x_i - x_{i-1})}{f(x_i) - f(x_{i-1})} = x_i + \frac{f(x_i)}{\left.\dfrac{f(x_i) - f(x_{i-1})}{x_i - x_{i-1}}\right\}} \Longrightarrow$$

Διακρι- τοποιηση παραγωγου

Ανανεωση πρωτης ταξης.

$$A \longleftarrow A + x y^T$$

$$\begin{array}{c} n \times n \\ [\;\;] \end{array} \longleftarrow \begin{array}{c} n \times n \\ [\;\;] \end{array} + \begin{array}{c} n \times 1 \\ [\;] \end{array} \begin{array}{c} 1 \times n \\ [\;\;] \end{array}$$

$A: n\times n \quad x: n\times 1 \quad y: 1\times n$

▷ **Ασκηση 1.7.1**

a. $p(x) = \sum_{i=0}^{n-1} a_i x^i = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4 + \dots + a_{n-1} x^{n-1}$

Εστω $Z = [\zeta_1, \dots, \zeta_m]$ ΝΔΟ $p = V \cdot A$ Επαληθευση

$$\begin{bmatrix} p(\zeta_1) \\ p(\zeta_2) \\ p(\zeta_3) \\ \vdots \\ p(\zeta_m) \end{bmatrix} = V a$$

$$p(\zeta_1) = a_0 + a_1 \zeta_1 + a_2 \zeta_1^2 + a_3 \zeta_1^3 + \dots + a_{n-1} \zeta_1^{n-1}$$

$$p(\zeta_2) = a_0 + a_1 \zeta_2 + a_2 \zeta_2^2 + a_3 \zeta_2^3 + \dots + a_{n-1} \zeta_2^{n-1}$$

$$\vdots$$

$$p(\zeta_m) = a_0 + a_1 \zeta_m + a_2 \zeta_m^2 + a_3 \zeta_m^3 + \dots + a_{n-1} \zeta_m^{n-1}$$

Σημείωση:

πχ $\left.\begin{array}{l} a_{11}x_1 + a_{12}x_2 = b_1 \\ a_{21}x_1 + a_{22}x_2 = b_2 \end{array}\right\} \Rightarrow \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$

Ομοίως προκύπτει:

$$\begin{bmatrix} p(\zeta_1) \\ p(\zeta_2) \\ \vdots \\ \vdots \\ \vdots \\ p(\zeta_m) \end{bmatrix} = \overset{V}{\begin{bmatrix} 1 & \zeta_1 & \zeta_1^2 & \zeta_1^3 & \dots & \zeta_1^{n-1} \\ 1 & \zeta_2 & \zeta_2^2 & \zeta_2^3 & \dots & \zeta_2^{n-1} \\ \vdots & & & & & \vdots \\ \vdots & & & & & \vdots \\ \vdots & & & & & \vdots \\ 1 & \zeta_m & \zeta_m^2 & \zeta_m^3 & \dots & \zeta_m^{n-1} \end{bmatrix}} \cdot \overset{a}{\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ \vdots \\ a_{n-1} \end{bmatrix}}$$

1) $p(x) = \sum_{i=0}^{n-1} a_i x^i = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + \dots + a_{n-1} x^{n-1} =$

$= a_0 + (a_1 + a_2 x + a_3 x^2 + \dots + a_{n-1} x^{n-2}) \cdot x =$

$= a_0 + (a_1 + (a_2 + a_3 x + a_3 x^2 + \dots + a_{n-1} x^{n-3}) x \cdot) x =$

$= \dots = a_0 + (a_1 + (a_2 + (\dots + (a_{n-2}) + a_{n-1}) x \cdot) x \dots x) x) =$

```
function[s] = horner(x,a)
    s = a_{n+1}
    for i = n-1:0
        s = sx + a_{i+1}
    end
```

```
~~for i = 1:n~~
~~    p(i) = my_horner(S(i),a)~~
~~end~~
```

V^T Ο αλγόριθμος που υπολογίζει το $p(\zeta_i)$:

```
for i = 1:m
    p(i) = horner(ζi, a)
end
```

Άσκηση 1.8.1

$$X_k = \sum_{l=0}^{L-1} x_l e^{-j\frac{2\pi}{L} lk} \qquad \text{FFT}$$

$$\begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_L \end{bmatrix} \rightarrow \text{Δεδομένα } (x_l)$$

Θέτω $W_L = e^{-j\frac{2\pi}{L}}$ $\longrightarrow$ συναρτήσεις βάσης FFT

άρα $X_k = \sum_{l=0}^{L-1} x_l \cdot W_L^{lk} = x_0 + x_1 W_L^{k} + x_2 W_L^{2k} + \dots + x_{L-1} W_L^{L-1}$

για $k=0$ έχω $X_0 = x_0 + x_1 + x_2 + \dots + x_{L-1}$

$k=1$ έχω $X_1 = x_0 + x_1 W_1 + x_2 W_L^2 + \dots + W_{L-1} W_L^{(L-1)}$

$\vdots$

$k=L-1$ έχω $X_{L-1} = x_0 + x_1 W_L^{(L-1)} + x_2 W_L^{2(L-1)} + \dots + x_{L-1} W_L^{(L-1)(L-1)}$

όπως και πριν το γράφουμε σαν γινόμενο πινάκων:

$$\begin{bmatrix} X_0 \\ X_1 \\ X_2 \\ \vdots \\ X_k \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & W_L & W_L^2 & \dots & W_L^{L-1} \\ \vdots & & & & \\ 1^{L-1} & W_L^{L-1} & W_L^{2(L-1)} & & W_L^{(L-1)(L-1)} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_{L-1} \end{bmatrix}$$

$W = W_L$

for $i=1:L$ (-> δεν βάζω L-1 γιατί θα γίνει 0 και δεν το δέχεται η Matlab)

$w(i) = W_L^{i-1}$

end

$V = zeros(L,L)$

for $i=1:L$

όλες τις γραμμές $V(:,i) = W_L^{(i-1)}$ για κάθε στήλη

end

β) Να αναφέρετε 3 σημαντικά χαρακτηριστικά / τάσεις στις σύγχρονες αρχιτεκτονικές που έχουν επίδραση άμεση στο σχεδιασμό των αλγορίθμων για τον Ε.Υ.

1. αρχιτεκτονική RISC (αρχεία καταχωρητών - LOAD/STORE - pipelining)

2. Ιεραρχική οργάνωση συστήματος αποθήκευσης.

καταχωρητές
κρυφή μνήμη
κύρια μνήμη
δευτερεύουσα μνήμη.

συνυφασμένη με την τοπικότητα των αναφορών.

3. Παράλληλες και δικτυακές αρχιτεκτονικές

Ερωτήσεις Θεωρίας

Κεφάλαιο 2

1. Μοντέλα για την υπολογιστική επίλυση προβλημάτων:
φυσικό μοντέλο --> μαθηματικό μοντέλο --> διακριτό μοντέλο
--> αριθμητικό μοντέλο --> υπολογιστικό μοντέλο

2. Ποια μοντέλα μελετάμε και γιατί;
- Υπολογιστικό
- Αριθμητικό
- Διακριτό

} Έχουν καλές επιδόσεις και παρέχουν ακριβή αποτελέσματα

3. Τι είναι υπολογιστικό μοντέλο και ποια γνωρίζετε;
4. Τι γνωρίζετε για το υπολογιστικό μοντέλο RAM;
-> κόστος εκτέλεσης & χρονοπολυπλοκότητα -> # αριθμητικών πράξεων του προγράμματος
-> αποτελείται: 1 επεξεργαστή + μνήμη που αποθηκεύονται προγράμματα + δεδομένα

βασικό στοιχείο για το σχεδιασμό αλγορίθμων: ελαχιστοποίηση του κόστους των αριθμητικών πράξεων του.

5. Για ποιους λόγους το μοντέλο RAM δεν επαρκεί να αποδώσει τις επιδόσεις που επιτυγχάνονται; Κάνοντας χρήση μερικών από τα κύρια χαρακτηριστικά σύγχρονων συστημάτων Η/Υ, όπως: 1. - ιεραρχική διαμόρφωση μνήμης για την επίτευξη μεγαλύτερης επίδοσης
- ταχύτερη μείωση του κόστους των αριθμητικών πράξεων σε σχέση με τη μείωση του κόστους μεταφοράς μεταξύ μνήμης και επεξεργασίας
- Την παραλληλία που μπορεί να διαθέτει το σύστημα επεξεργασίας

6. Ποια τα κυρια χαρακτηριστικά του υπολογιστικού μοντελου με ιεραρχία μνήμης (το οποίο αναπαριστάται καλύτερα από το RAM για σύγχρονα συστήματα Η/Υ

Μετρητές και ανάλυση επίδοσης

- O: αριθμός των αριθμ. πράξεων.

πχ $\begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix} \begin{bmatrix} b_1 & \cdots & b_n \end{bmatrix}$ χρειάζομαι n πολ/σμούς και $n-1$ προσθέσεις άρα

$O = n + n - 1 = 2n - 1$ πράξεις

- Φ: αριθμός μεταφορών μεταξύ κυριας μνήμης & καταχωρητών ή κρυφής μνήμης

πχ $\begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix} \begin{bmatrix} b_1 & \cdots & b_n \end{bmatrix}$ Απαιτούνται: n μεταφορές για το διανυσμα a + n -"- διανυσμα b. Προκύπτει βαθμωτός άρα: $\Phi = n + n + 1$
(1 για την αποθήκευση του αποτελεσματος)

- Φ_{min} ο ελάχιστος αριθμός μεταφορών Φ που θα απαιτούνταν για την υλοποίηση αν διαθέταμε απεριόριστη μνήμη σε όλα τα επίπεδα.

Θεώρημα: Αν ο αριθμός των δεδομένων εισόδου είναι n, και ο αλγόριθμος τα χρησιμοποιεί όλα για να υπολογίσει τουλάχιστον ένα αποτέλεσμα τότε: $\Phi_{min} \geq n + 1$

* όταν έχω m αποτελέσματα: $\Phi_{min} \geq n + m$

- MFlops : μοναδα μετρησης της αποδοσης της υλοποιησης στο υπολογιστικο συστημα. $\boxed{\frac{\#\text{ πραξεων ανα}}{\text{Second}}}$

<u>Συνολικος χρονος εκτελεσης ενος προβληματος.</u>

$$\left.\begin{array}{ll} T = T_{αριθ} + T_{μετ} & (1) \\ T_{αριθ} = τ_{αριθ} \cdot \Omega & (2) \\ T_{μετ} = τ_{μετ} \cdot \Phi & (3) \end{array}\right\} \quad (1) \xRightarrow{2,3} T = τ_{αριθ} \cdot \Omega + τ_{μετ} \cdot \Phi =$$

$$= τ_{αριθ} \Omega \left(1 + \frac{τ_{μετ} \cdot \Phi}{τ_{αριθ} \cdot \Omega} \right) \quad \text{ομως} \quad \mu = \frac{\Phi}{\Omega}$$

οπου μ: αριθμος μεταφορων ανα αριθμητικη πραξη και αντιστοιχα

μ_{min}: ο ελαχιστος αριθμος μεταφορων ανα αριθμητικη πραξη $\mu_{min} = \frac{\Phi_{min}}{\Omega}$.

Αρα $T = T_{αριθ} \left(1 + \mu \cdot \frac{τ_{μετ}}{τ_{αριθ}} \right) \geq T_{αριθ} \left(1 + \mu_{min} \frac{τ_{μετ}}{τ_{αριθ}} \right)$

i. Αν διαφορετικες υλοποιησεις ενος αλγοριθμου απαιτουν τον ιδιο # αριθμητικων πραξεων αλλα διαφερουν στον # μεταφορων μπορουμε να συγκρινουμε την αποτελεσματικοτητα τους με βαση τα αντιστοιχα μ.

ii. Αν η προσβαση στη κεντρικη μνημη ηταν στιγμιαια τοτε $τ_{μετ} = 0$ αρα $T = T_{αριθ} = τ_{αριθ} \Omega$

J

- Τι είναι προμεταφορά και με ποιους τρόπους μπορεί να επιτευχθεί;

Ένας τρόπος για τη μείωση ή απόκρυψη της καθυστέρησης που οφείλεται στη μεταφορά στοιχείων από τη μνήμη, είναι χρήση προφόρτωσης ή προμεταφοράς (=prefetching). [illegible] Εφόσον σε σύγχρονα συστήματα επιτρέπεται η σύγχρονη εκτέλεση εντολών μεταφοράς και αριθμητικής επεξεργασίας είναι δυνατόν να ζητηθεί η μεταφορά στοιχείων που θα είναι χρήσιμα σε μελλοντική αριθμητική επεξεργασία. 2 Τρόποι:

- Μέσω του συστήματος: Ο μεταφραστής εισάγει στον κώδικα εντολές για την κλήση στοιχείων που πρόκειται να χρειαστούν μελλοντικά.

- Μέσω αμέσων εντολών: Ο χρήστης μετατρέπει τον κώδικα εισάγοντας εντολές που καλούν στοιχεία που πρόκειται να χρησιμοποιηθούν.

Για να πετύχει η προμεταφορά αν τα στοιχεία πρόκειται να χρειαστούν στον κύκλο T του προγράμματος η εντολή μεταφοράς τους στον κρυφή μνήμη πρέπει να δοθεί στον κύκλο T-τ όπου τ ο μέγιστος # των κύκλων που απαιτούνται για την μεταφορά των στοιχείων από τη μνήμη.

Άσκηση 2.3.1

$C \leftarrow C + AB$

$C \in R^{n_1 \times n_2}$
$A \in R^{n_1 \times n_3}$
$B \in R^{n_3 \times n_2}$

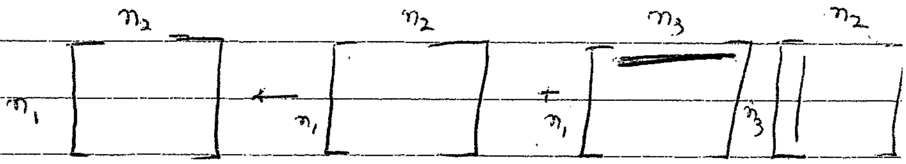
$$\mu_{min} = \frac{\Phi_{min}}{\Omega}$$

Υπολογισμός Φ_{min}

Για τη μεταφορά των στοιχείων του C έχω : $n_1 \times n_2$
— " — — " — A —"— : $n_1 \times n_3$
— " — — " — B —"— : $n_3 \times n_2$
Για την αποθήκευση στον C θα έχω : $n_1 \times n_2$

Άρα $\Phi_{min} = n_1 \times n_2 + n_1 \times n_3 + n_3 \times n_2 + n_1 \times n_2$

Υπολογισμός Ω:



Αριθμός πράξεων για το AB
$(n_1 \times n_3)\ (n_3 \times n_2)$

n_3 πολ/σμοί + $n_3 - 1$ προσθέσεις για το πρώτο στοιχείο του AB
άρα για όλα τα στοιχεία του θα έχω $(n_1 \times n_2)(2n_3 - 1)$

Άρα για την πρόσθεση C + AB έχω τόσες προσθέσεις όσα και τα στοιχεία τους άρα $n_1 \times n_2$

Άρα $\Omega = (2n_3 - 1) n_1 n_2 + n_1 n_2$

άρα $$\mu_{min} = \frac{\Phi_{min}}{\Omega} = \frac{2n_1 \times n_2 + n_1 \times n_3 + n_3 \times n_2}{(2n_3 - 1) n_1 \times n_2 + n_1 \times n_2}$$

άρα

1. για $n_1 = n_2 = 1$ και $n_3 = n$ έχω:

$$\mu_{min} = \frac{2+n+n}{(2n-1)+1} = \frac{2+2n}{2n} = \frac{n+1}{n}$$

2. για $n_2 = n_3 = 1$ και $n_1 = 1$

$$\mu_{min} = \frac{2+1+1}{2} = \frac{4}{2} = 2$$

3. $n_1 = n_2 = n_3 = n$ $\quad \mu_{min} = \frac{2n^2+n^2+n^2}{(2n-1)n^2+n^2} = \frac{4n^2}{n^2(2n-1+1)} = \frac{4}{2n} = \frac{2}{n}$

Άσκηση 2.3.2

$A \in R^{n \times n}$

$x \in R^n$ $\qquad y = (A - \omega I)x$

$\omega \in R$

I ταυτοτικό μητρώο $\qquad \mu_{min} = ?$

- $y = (A - \omega I)x$

$$\Phi = n^2 + 1 + n^2 + n + n = 2n^2 + 2n + 1$$

- $y = Ax - \omega x I = Ax - \omega x$

$$\Phi = n^2 + n + 1 + n = n^2 + 2n + 1 = \Phi_{min}$$

$$\underline{\Omega} = (2n-1) \cdot n + n + n = 2n^2 - 2n + 2n = 2n^2$$

άρα $\mu_{min} = \frac{\Phi_{min}}{\underline{\Omega}} = \frac{n^2+2n-1}{2n^2}$

LOAD ω, x   φορτωνω τα ω, x

```
for i=1:n
   LOAD A(i,:)   φορτωνω τη i γραμμή του A
   y(i) = A(i,:) x - ω xi
end
STORE y
```

Ασκηση 2.33

Εστω ο βροχος:

```
for i=1:n
    y(i) = a*x(i) + y(i)
end
```

Να ξαναγραψετε το βροχο χρησιμοποιωντας ξετυλιγμα μηκους b=3.

z = mod(n,3) → το υπολοιπο της διαιρεσης.

```
for i= 1:3:n-z
    y(i) = a*(xi) + y(i)
    y(i+1) = a * x(i+1) + y(i+1)
    y(i+2) = a*x(i+2) + y(i+2)
end

if (z~=0)  % δεν ειναι ίσο
   for j=1:z
   y(i+j) = a* x(i+j) + y(i+j)
   end
end
```

Άσκηση 1η Επιστημονικού Υπολογισμού:

1. 4.

Να βρεθεί το μ_{min} του $c^{(s)} = \left[\prod_{i=1}^{s}\left(\mu B + (1-\mu)\dfrac{ee^T}{e^Te}\right)\right] b$

$B \in R^{n\times n}$

$e, b \in R^{n\times 1}$ $\qquad \mu_{min} = ?$

$\mu \in R$

$\underset{n\times1 \ \ 1\times n}{ee^T}$ $\qquad \underset{1\times n \ \ n\times 1}{e^Te}$

$\underset{n\times n}{A} \times \underset{n\times 1}{b}$

Θέτω $A = \mu B + (1-\mu)\cdot\dfrac{ee^T}{e^Te}$

i) Με συνάρτηση δύναμης: η πρ[illegible]

$$A^s b = (\overbrace{((A\cdot A)\cdot A)A)A\ldots A}^{s})\cdot b$$

$$\Phi = \underbrace{1}_{\text{για το } \mu} + \underbrace{n^2}_{\text{για το } B} + \underbrace{n}_{\text{για το } e} + \underbrace{n}_{\text{για το } b} + \underbrace{n}_{\text{για το αποτέλεσμα}} = n^2+3n+1$$

Αριθμός πράξεων:

O_1: # πράξεων για το A.

$e^Te \to$ βαθμωτό

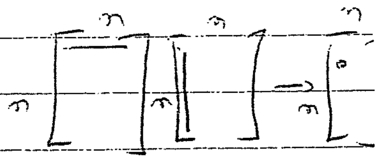
$$O_1 = \underbrace{n^2}_{\text{για το } \mu B} + \underbrace{1}_{\text{για το } (1-\mu)} + \underbrace{n^2}_{\text{για το } ee^T} + (n+n-1) + \underbrace{1}_{\text{πράξη } \frac{1-\mu}{e^Te}} + \underbrace{n^2}_{\text{πολ/σμός βαθμωτού}}$$

$\dfrac{(1-\mu)}{e^Te}ee^T \to n^2$

$$+ \underbrace{n^2}_{\text{πρόσθεση 2 πινάκων } n\times n}$$

Άρα $O_1 = 4n^2 + 2n + 1$

O_2: # πράξεων για το A^s



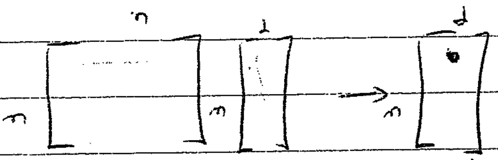
Ο A θα προκύπτει πίνακας $n \times n$.

$$O_2 = \Big[\underbrace{(n+n-1)}_{\text{για ένα στοιχείο του αποτελέσματος } A\times A} \cdot \underbrace{n^2}_{\text{επί } n^2 \text{ για όλα τα στοιχεία}}\Big]\underbrace{(s-1)}_{\text{επί τον αριθμό των πολ/σμών}}$$

Άρα $O_2 = n^2(2n-1)\cdot(s-1)$

O_3: # πράξεων για το $A^s \cdot b$

Το A^s προκύπτει πίνακας $n\times n$. Άρα $A^s\cdot b$ είναι



Άρα $O_3 = \Big[\underbrace{n}_{\text{πολ/μοί}} + \underbrace{n-1}_{\text{προσθ.}}\Big]\cdot \underbrace{n}_{\text{στοιχεία}} = (2n-1)\,n = 2n^2 - n$

Άρα $O_{ολ} = O_1 + O_2 + O_3 = 4n^2 + 2n + 1 + n^2(2n-1)(s-1) + 2n^2 - n$

άρα $\psi_{min} = \dfrac{\phi_{min}}{O_{ολ}}$

ii) Προς αριστερά $(A(A(A(A(A(A\ldots(A(A\cdot b))))))))$

Θέτω $A b = d$

$$d = A\cdot b = \mu B\cdot b + \frac{(1-\mu)}{e^T e}\, e e^T b$$

ϱ_1 : # πράξεων για το θ

$$\varrho_1 = \underset{B\cdot b}{(2n-1)n} + \underset{\gamma \times Bb}{n} + \underset{e^T e}{n} + \underset{(1-\gamma)}{1} + \underset{\frac{1-\gamma}{e^T e}}{1} +$$

$$+ \underset{e^T b}{(2n-1)} + \underset{\substack{e(e^T b)\\ \text{διανύσματος } n\times 1}}{n} + \underset{\frac{(1-\gamma)}{e^T e}\cdot ee^T b}{n} + \underset{\substack{\text{πρόσθεση}\\ \text{διανυσμάτων } n\times 1}}{n}$$

ϱ_2 : # πράξεων $\overbrace{A\cdot A\cdot A\cdot}^{s-1}(Ad)$.

$$\begin{array}{c} \\ n \end{array}\begin{array}{c} n \\ \begin{bmatrix} & A & \end{bmatrix} \end{array} \begin{array}{c} \\ n \end{array}\begin{array}{c} 1 \\ \begin{bmatrix} d \end{bmatrix} \end{array} \rightarrow \begin{bmatrix} \ \end{bmatrix} \quad \text{άρα} \quad (2n-1)\cdot n$$

αυτό θα το πολ/σω $s-1$ φορές με $n\times n$

άρα $\varrho_2 = n(2n-1)(s-1)$

Άρα $\varrho_{ολ} = \varrho_1 + \varrho_2$

iii) Προς τα δεξιά

$\underbrace{(A\cdot A\cdot A\cdot A\cdot A\cdot\ A\cdot)}_{s\ \text{φορές}}\, b$ $\qquad \varrho_1$ = # πράξεων για το A (όπως 1))

ϱ_2: $A\cdot A\cdot A \ldots A$ $\qquad n\begin{bmatrix} \ \end{bmatrix} n\begin{bmatrix} \ \end{bmatrix} \rightarrow n\begin{bmatrix} \ \end{bmatrix}$

ϱ_2: $(2n-1)\,n^2\cdot s$ $\qquad n\begin{bmatrix} \ \end{bmatrix}\cdot\begin{bmatrix} \ \end{bmatrix} \rightarrow \begin{bmatrix} \ \end{bmatrix}$

άρα $\varrho_3 = (2n-1)\cdot n \qquad \varrho_{ολ} = \varrho_1 + \varrho_2 + \varrho_3$

Υπολογιστικά Μοντέλα
- load/store
- Ιεραρχία Μνήμης $K \leq M$
- χρονος $+, -, \backslash, *$ -> $\tau_{αριθμ}$

Μετρητές και ανάλυση επίδοσης

- Ω: αριθμός πράξεων κινητής υποδιαστολής (α.κ.υ)
- Φ: αριθμός μεταφορών μεταξύ κύριας μνήμης και καταχωρητών ή κρυφής μνήμης
- $Φ_{min}$: Ελάχιστος αριθμός μεταφορών Φ που μπορεί να υπολογίζει αν ~~χρησιμοποιούσαμε~~ διαθέταμε ~~ελάχιστη~~ απεριόριστη μνήμη σε όλα τα επίπεδα

Θεώρημα: Αν ο αριθμός των δεδομένων εισόδου είναι n και ο αλγόριθμος τα χρησιμοποιεί όλα για να υπολογίσει τουλάχιστον ένα αποτέλεσμα τότε $Φ_{min} \geq n+1$

(n load $+1$ store $\rightarrow$ # μεταφορών: $n+1$)

$\Rightarrow$ Κόστος εκτέλεσης μιας υλοποίησης:

$$T = T_{αριθ} + T_{μετ} = \tau_{αρ}\Omega + \tau_{μετ}\Phi = T_{αριθ}\left(1 + \frac{\tau_{μετ}}{\tau_{αριθ}}\frac{\Phi}{\Omega}\right) =$$

$$= T_{αριθ}\left(1 + \mu \frac{\tau_{μετ}}{\tau_{αριθ}}\right) \qquad \mu = \Phi/\Omega$$ αριθμός μεταφορών ανά αριθμ. πράξη

$$T = T_{αριθ}\left(1 + \mu \frac{\tau_{μετ}}{\tau_{αριθ}}\right) \geq T_{αριθ}\left(1 + \mu_{min} \frac{\tau_{μετ}}{\tau_{αριθ}}\right)$$

θεωρία

Κεφάλαιο 3.

1. Πού οφείλεται η απώλεια πληροφορίας κατά τη διαρκεια των διαδικασιών επίλυσης ενός προβλήματος
   1. Μαθηματική μοντελοποίηση & στις απλοποιήσεις
   2. Διακριτοποίηση και στα σφάλματα αποκοπής
   3. Σφάλματα στρογγύλευσης λόγω χρήσης ΑΚΥ.

✓. Η επίδρασή τους δεν είναι πάντα καταστροφική!!

Πρέπει να ελέγχουμε
- τι επίδραση έχουν
- πως τα μειώνουμε.

Προσεγγίζω τη ποσότητα x με το $\hat{x}$

2. Απόλυτο Σφάλμα $\epsilon_{abs} = |x - \hat{x}|$

Σχετικό Σφάλμα $\epsilon_{rel} = \dfrac{|x-\hat{x}|}{|x|}$

- Όταν $|x|$ είναι πολύ μικρό ή πολύ μεγάλο χρησιμοποιώ τ ϵ_{rel}

Διανύσματα $x \in C^n$ $\} \|x\|$	Μητρώα
$\|x\|_1 = \sum_{i=1}^{n} \|x_i\|$: άθροισμα απόλυτων τιμών	$\|A\|_1 = \max_{1\le i\le n} \sum \|a_{ij}\|$. το μεγαλύτερο άθροισμα στο απόλ. γραμμ.
$\|x\|_2 = \left(\sum_{i=1}^{n} \|x_i\|^2\right)^{\frac{1}{2}}$ Ευκλείδια νόρμα	$\|A\|_2 = \sqrt{\lambda_{max}(A^T A)}$
$\|x\|_p = \left(\sum_{i=1}^{n} \|x_i\|^p\right)^{\frac{1}{p}}$	$\|A\|_F = \sqrt{trace(AA^T)}$ ↳ άθροισμα διαγωνίων στοιχείων.
$\|x\|_\infty = \max_{1\le i\le n} \{\|x_i\|\}$ Η μεγαλύτερη κατά απόλυτη τιμή x_i	$\|A\|_\infty = \max_{1\le j\le n} \sum_{j=1}^{n} \|a_{ij}\|$ το μεγ. άθροισμα στο στη

$\epsilon_{abs} = \|x - \hat{x}\|$ και $\epsilon_{rel} = \dfrac{\|x - \hat{x}\|}{\|x\|}$

- Παρατηρούμε ότι:

$\|[\overbrace{1,0,0,0 \ldots, 0}^{1000 \text{ στοιχεία}}]\|_1 = 1$

$\|[10^{-3} \ldots 10^{-3}]\|_1 = 1$

} Ενώ έχω διαφορετικά στοιχεία παίρνω ίδια νόρμα

- Μπορώ λοιπόν και για νόρμες να χρησιμοποιήσω τις απόλυτες τιμές των στοιχείων:
  - έστω $x \in C^n$ τότε $y = |x| \Rightarrow y_i = |x_i|$ $i = 1:n$
  - Αν $x, y \in R^n$ τότε
    $x \le y \Rightarrow x_i \le y_i$ $i = 1:n$ $\forall y, x \in R^n$

Άρα $|x_i - \hat{x}_i| \to$ παίρνω το μέγιστο σφάλμα $\max |x_i - \hat{x}_i|$ άρα όλα τα υπόλοιπα σφάλματα θα είναι μικρότερα από αυτό.

Αντίστοιχα ορίζω και το $\max\limits_i \dfrac{|x_i - \hat{x}_i|}{|x_i|}$

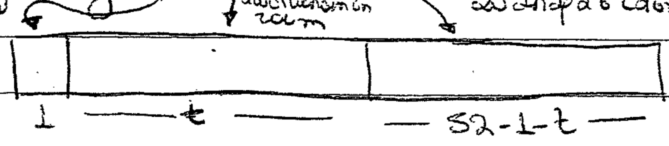
Σύστημα αριθμών κινητής υποδιαστολής.

→ Τα στοιχεία αυτού είναι της μορφής

βάση του συστήματος απλά 2

$y = \pm m \beta^{e-t}$

απεικόνιση προσήμου · απεικόνιση τιμής · αναπαράσταση τους



- $y \in F$ όπου F: ένα υποσύνολο των ρητών αριθμών

(ρητοί οι αριθμοί που μπορούν να γραφτούν και με τη μορφή κλάσματος

- Όλα τα y είναι ρητοί

- $fl : R \to F$
  ↳ στρογγύλευση ενός αριθμού στο α.κ.υ σύστημα.

- Ορίζουμε επίσης το σύνολο: $G \supset F$

$$G := \{x \in R : \mu \leq |x| \leq M\} \cup \{0\} \subset R$$

μ: ελάχιστος μη μηδενικός αυτό που μπορώ να αναπαραστήσω

M: μέγιστος

* Το G είναι ένα διάστημα του οποίου τα άκρα φράσσονται από τα μ, M και συμπεριλαμβάνεται το 0.

Απεικόνιση $fl(x)$

- $x \in F \to fl(x) = x$, το αποτέλεσμα είναι το ίδιο το x

- $x \in G$ και $x \notin F$ οπότε $fl(x) \neq x$ τότε η απεικόνιση $x \to fl(x) \in F$ είναι η στρογγυλοποίηση ή στρογγύλευση του x. Αφού $fl(x) \neq x$ προκαλείται σφάλμα.

Άρα $fl(x) = x(1+\delta)$

$$x \notin G \begin{cases} |x| > M \ : \text{υπερχείλιση} \\ |x| < m \ \& \ x \neq 0 \ : \text{υποχείλιση} \end{cases}$$ $fl(x) \notin F$ (ο αριθμός που αποθηκεύεται είναι μικρότερος ή πολύ μικρότερος)

e_1 e_2

→ στρογγυλοποίηση προς το μηδέν
→ στρογγυλοποίηση προς τον πλησιέστερο και μακρύτερο από το 0

SOS „Κανονικοποίηση και κρυμμένο bit"

Στην αναπαράσταση των α.κ.υ ένας αριθμός μπορεί να γραφεί:

$$y = 0.d_1 0000 \times b^e$$
$$y = 0.0d_1 0000 \times b^{e+1}$$
$$y = 0.00d_1 0000 \times b^{e+2}$$

Η έλλειψη μοναδικότητας

στην αναπαράσταση δημιουργεί δυσκολίες σε πράξεις π.χ. σύγκριση αριθμών

Για να επιβάλουμε την μοναδικότητα της αναπαράστασης κανονικοποιούμε το σύστημα και υποχρεώνουμε το πρώτο ψηφίο του m (στη βάση b) να είναι μη μηδενικό δηλ. $y \in RUF \Rightarrow d_1 \neq 0$

Όταν η βάση είναι $b=2$ τότε $d_1 \neq 0 \Rightarrow d_1 = 1$. Άρα στα συστήματα που χρησιμοποιούν για βάση το 2, η κανονικοποίηση σημαίνει ότι το πρώτο bit του m είναι πάντα 1. Επομένως μπορούμε να μην το αποθηκεύουμε καθόλου. Θα πρέπει όμως οι αλγόριθμοι που χρησιμοποιούνται να λαμβάνουν υπ' όψιν τους το κρυμμένο bit

Κανονικοποιημένο $y \in F$

$$y = \pm b^e \times .d_1 d_2 \ldots d_t \qquad 0 \le b_i \le b-1, \ d_1 \neq 0$$

Ιδιότητες α.κ.υ. συστήματος

- $y \in F$ και $y \neq 0$ τότε $b^{e_{min}-1} \leq |y| \leq b^{e_{max}}(1-b^{-t})$

- $z \in G$ και z_-, z_+ οι πλησιέστεροι α.κ.υ όπου $z \in [z_-, z_+]$ ώ δεν υπάρχουν άλλοι α.κ.υ που να ισχύει $z_- < z_-^* \leq z \leq z_+^* < z_+$

$z = .\overbrace{x \ldots x}^{t \text{ ψηφία}} \; n \times b^{e}$ n τα υπόλοιπα ψηφία του z.

$z_- = m \times b^{e-t}$ και $z_+ = (m+1) \cdot b^{e-t}$

Το ύψος του z_- από το z είναι μέγιστο όταν το z βρίσκεται στη μέση του $[z_-^*, z_+^*]$ άρα

$$\boxed{|z - fl(z)| \leq \frac{b^{e-t}}{2}} \longrightarrow \text{απόλυτο σφάλμα}$$

$$\frac{|z - fl(z)|}{|z|} = \frac{b^{e-t}/2}{b^{e-t}} \leq \frac{b^{1-t}}{2} = u. \quad \text{Σχετικό σφάλμα}$$

$u = \dfrac{b^{1-t}}{2}$ μονάδα στρογγύλευσης

- Αν $z \in G$ τότε $fl(z) = z(1+\delta)$ για $|\delta| < u$

- Οι "αποστάσεις" μεταξύ των διαδοχικών αυξ μεγαλώνει κατά b.

$$|m b^{e-t} - (m+1) b^{e-t}| = b^{e-t}$$
$$|m b^{e+1-t} - (m+1) b^{e+1-t}| = b^{e+1-t}$$

Έψιλον της μηχανής

- $1+x = 1+x$ → Δεν ισχύει στους υπολογιστές

- για κάθε x τ.ω. $0 \le x < \varepsilon$ → $1 \tilde{+} x = 1$

eps → έψιλον μηχανής

$$\delta(x,x^+) = x^+ - x$$

$x \qquad x^+$

Ορισμός ε_u: ε_M : η απόσταση από το 1 ως τον αμέσως μεγαλύτερο α.κ.υ

$$\varepsilon_M := \delta(1, 1^+)$$

$$\varepsilon_M := \arg\min_{\varepsilon>0}\{fl(1+\varepsilon)>1\}.$$

$\varepsilon_M = 2^{1-t}$ όμως $u = 2^{-t}$ άρα $\boxed{\varepsilon_M = 2u}$

Αριθμητικές πράξεις και μετάδοση σφάλματος στρογγύλευσης

Για κάθε μια από τις πράξεις $\odot\{-,+,*,/\}$ $\exists\, x,y \in F$ τ.ω. $x \odot y \notin F$.

$\tilde{\odot}$: συμβολίζει την υλοποίηση της πράξης $\odot$

για $x,y \in F$ ισχύει ότι $x \tilde{\odot} y = Fl(x \odot y) \in F$ δηλ. το αποτέλεσμα της πράξης στο σύστημα είναι σαν να εκτελείται η πράξη ακριβώς στο R (δηλ. $x \odot y$) και μετά να στρογγυλοποιείται το αποτέλεσμα. Τότε λέμε ότι έχουμε ακριβή στρογγύλευση

Χωρίς ψηφίο προστασίας:

$$fl(x\pm y) = (1+a)\,x + (1+b)\,y \qquad ,\ |a|,|b|\le u$$

Πράξη για $x\pm y$ στο $\mathbb{R}$ με λίγο αλλοιωμένα τα x, y.

A1 $fl(x+y) = fl(y+x)$

-inf 0 +inf

$-\infty$ $+\infty$

A2. $x_1+x_2+x_3+x_4$

$$\left.\begin{array}{l} ((x_1 \tilde{+} x_2) \tilde{+} x_3) \tilde{+} x_4 \\ (x_1 \tilde{+} x_2) \tilde{+} (x_3 \tilde{+} x_4) \end{array}\right\} \text{Διαφορετικά αποτελέσματα.}$$

$\boxed{\text{Σχετικό Σφάλμα}}$ $\dfrac{|fl(x\odot y) - x\odot y|}{|x\odot y|} \le u \quad , x\odot y \ne$

$\boxed{\text{Μοντέλο μετάδοσης σφάλματος}}$

$x, y\in F$ και $x\odot y \in G$ τότε

$$fl(x\odot y) = (x\odot y)(1+\delta) \quad \text{για κάποιο } |\delta|\le u$$

και

$$fl(x\odot y) = \frac{x\odot y}{1+\delta} \quad \text{για } |\delta|\le u$$

$x\odot y \in F \Rightarrow \delta = 0$

$x\,\tilde{\odot}\, y = fl(x\odot y) = (x\odot y)(1+\delta)$ με $\tilde{x} = x(1+\delta)$ και $\tilde{y} = y(1+\delta)$

Εστω $A^{n\times n}$

$fl(a_{ij}) = a_{ij}(1+\delta_{ij})$, $|\delta_{ij}| \leq u$

1. ΝΔΟ $|fl(A)-A| \leq |A|\,u$

$|fl(A)-A| \leq |a_{ij}(1+\delta_{ij}) - a_{ij}| = |a_{ij}\delta_{ij}| = |a_{ij}||\delta_{ij}| \Rightarrow$

ξερω οτι $|\delta_{ij}| \leq u$ οποτε $|fl(A)-A| \leq |A|\,u$

2. ΝΔΟ $fl(bA) = b\cdot A + E$, $|E| \leq u\,|bA|$

$fl(b\cdot A) = b\cdot a_{ij}(1+\delta_{ij}) = b a_{ij} + b a_{ij}\delta_{ij}$ ομως

$|b a_{ij}\delta_{ij}| = |b\cdot a_{ij}|\cdot|\delta_{ij}|$ ομως $|\delta_{ij}| < u$ αρα

$|b a_{ij}\delta_{ij}| \leq u|b\cdot A|$

αρα $fl(b\cdot A) = b\,a_{ij} + E = bA + E$

3. ΝΔΟ $fl(A+B) = (A+B) + E$, $|E| \leq u\,|A+B|$

$fl(A+B) = fl(a_{ij}+b_{ij}) = (a_{ij}+b_{ij})(1+\delta_{ij}) = a_{ij}+b_{ij}+a_{ij}\delta_{ij}+b_{ij}\delta_{ij}$

$= (a_{ij}+b_{ij}) + (a_{ij}+b_{ij})\delta_{ij}$ ομως $|\delta_{ij}| \leq u$ αρα

$fl(A+B) = (A+B) + (A+B)\,u = (A+B) + E$

$z + x\psi \quad fl(z + xy) = (z + x\psi)(1+\delta)$

$fl(z + fl(x\psi)) = (z + x\psi(1+\delta))(1+\delta)$

Παραδειγμα 3.3.6.

$x, y \in F \qquad z = \dfrac{x}{\sqrt{x^2+y^2}}$

$fl(\sqrt{x}) = \sqrt{x}\,(1+\delta)$

$t_1 \leftarrow fl(x*x) \longrightarrow x^2(1+\delta_1)$

$t_2 \leftarrow fl(y*y) \longrightarrow y^2(1+\delta_2)$

$t_3 \leftarrow fl(t_1+t_2) \longrightarrow (t_1+t_2)(1+\delta_3)$

$t_4 \leftarrow fl(\sqrt{t_3}) \longrightarrow \sqrt{t_3}\,(1+\delta_4)$

$t_5 \leftarrow fl(x/t_4) \longrightarrow \dfrac{x}{t_4}(1+\delta_5) \qquad \leftarrow z$

Αρα $z = \dfrac{x}{t_4}(1+\delta_5) = \dfrac{x}{\sqrt{t_3}\,(1+\delta_4)}(1+\delta_5) = \dfrac{x}{\sqrt{(t_1+t_2)(1+\delta_3)}\cdot(1+\delta$

$\cdot(1+\delta_5) = \dfrac{x}{\sqrt{[x^2(1+\delta_1)+y^2(1+\delta_2)](1+\delta_3)}\,(1+\delta_4)}(1+\delta_5)$

$\forall\ \delta_j$ ξερω $|\delta_j| \le u$

Παραδειγμα 3.3.7.

Εστω $y = \dfrac{(1+x)-1}{x} \qquad fl(y) = \dfrac{[(1+x)(1+\delta_1) - 1](1+\delta_2)}{x}(1+$

$y = 1$ και $x < \varepsilon_u$ τοτε $fl(y) = (1+\delta_1)(1+\delta_2)(1+\delta_3) + \dfrac{\delta_1(1+\delta_2)(1+\delta_3)}{x}$

αρα οταν $x \geq \varepsilon u$

$$fl(y) = 1 + (\delta_1 + \delta_2 + \delta_3) + \delta_1\delta_2 + \delta_2\delta_3 + \delta_1\delta_3 + \delta_1\delta_2\delta_3 +$$

$$+ \delta_1 \frac{(1+\delta_2)(1+\delta_3)}{x} = 1 + O(u)$$

$$|y - fl(y)| = |1 - 1 + O(u)| \simeq O(u)$$

σχετικο σφαλμα $\frac{fl(y) - y}{y} = 1$

Εξαιρέσεις & Ειδικοί Αριθμοί

$\boxed{SOS}$

invalid op → $0/0$, $0 \times \infty$, $\sqrt{-1}$ → NAN.

overflow → → $\pm$ Inf

devide by 0 → finite number/0 → $\pm$ InF

underflow → → subnormal numbers

inexact → αν $fl(x \odot y) \neq (x \odot y)$ → στρογγυλευση

Παρατηρηση 3.4.1

- $1/(1/0)$ $(1/0) \to$ inf, $(1/\text{inf}) \to 0$ } αρα 0.

- $1/1/0$ $1/1 \to 1$, $1/0 \to$ inf } αρα inf

- $0/0$ $0/0 \to$ NaN
- $1/0$ $1/0 \to$ Inf

- max(ans, 4): inf

ans: η μεταβλητη οταν οποια καταχωρουμε το αποτελεσμα πρ..

- min(ans, 3): 3

Realmin : ο μικρότερος α.κ.υ. που μπορώ να αποθηκεύσω.

Λύση: 3.3.17

~~[illegible]~~
temp2 = realmin/2;
temp = ~~2~~ temp2;
boole (temp == realmin);

Στρογγυλοποίηση	με βαθμ. κανονικοπ.	≠ βαθμ. κανον.
προς το 0	realmin/2	0
	realmin	0
	1	0

Στρογγυλοποίηση προς τον πλησιέστερο και μικρότερο από το 0 — realmin, 2realmin, 0.

ΘΕΜΑ II ΠΡΟΟΔΟΣ 2005.

2.
i) realmax + realmin = realmax
ii) realmin/0 = inf
iii) realmin/2 == 0 ? ~~[illegible]~~ με βαθμιαία κανονικοποίηση επιστρέφει realmin/2 Δεν ισχύει η συνθήκη άρα επιστρέφεται 0
Αν δεν είχαμε βαθμ. κανονικοποίηση realmin/2 → 0 και θα επέστρεφε 1
iv) realmax + realmin/2 = inf

Άσκηση 3.8.14.

$A = [100, 1; 0, -10]$

$B = [1, 0; 0, 0]$

$C = A./B$: στοιχείο προς στοιχείο διαίρεση.

$C = [100, +\text{inf}; \text{Nan}, -\text{inf}]$

Εκτίμηση σφάλματος και ποιότητας υπολογισμών.

- $x \in U$ → πραγματικά δεδομένα
- $f(x)$ → τιμές των δεδομένων χωρίς λάθη
- $x^* \in F$ → στοιχεία εισόδου που χρησιμοποιούνται στην υλοποίηση $x^* = fl(x)$ διατηρημένα
- $f(x^*)$ → τιμές του x^* (με αριθμητική ακρίβεια)
- $f_{\text{προγ.}}$ → υλοποίηση της f με α.κ.υ. πράξεις

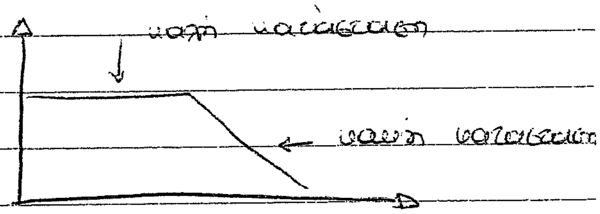
Απόλυτο Σφάλμα : $\| f_{\text{προγ}}(x^*) - f(x) \|$

Σχετικό Σφάλμα : $\dfrac{\| f_{\text{προγ}}(x^*) - f(x) \|}{\| f(x) \|}$

Αν το σφ. αλγ. είναι μικρό ο αλγόριθμος είναι ακριβής

Κατάσταση προβλήματος

Η απόλυτη και η σχετική απόσταση των $f(x^*)$ και $f(x)$ μπορεί να είναι μεγάλη. Το f είναι ευαίσθητο στις μικρές αλλαγές στις τιμές εισόδου περί το x και είναι ανεξάρτητο της στρογγυλοποίησης και της χρήσης του x^* αντί του x.
Τότε λέμε ότι η κατάσταση του προβλήματος είναι κακή κοντά στο x



Παράδειγμα 3.5.1.

1. Θεώρημα Taylor:

$$f(x+\Delta x) = f(x) + f'(x)\Delta x + \frac{f^{(2)}(x)(\Delta x)^2}{2!} + \frac{f^{(3)}(x)(\Delta x)^3}{3!} + \dots + \frac{f^{(n)}(x)(\Delta x)^n}{n!} + \frac{f^{(n+1)}(x+\theta\Delta x)(\Delta x)^{n+1}}{(n+1)!}$$

$f: R \to R$, $y=f(x)$;

στοιχείο εισόδου $x^* = x+\Delta x$ αποτέλεσμα $\tilde{y} = f(x+\Delta x)$ (R_n όπου ...)

$|\hat{y} - y|$

$$|f(x+\Delta x) - f(x)| = |\tilde{y} - y|$$

$$\tilde{y} - y = f(x+\Delta x) - f(x) = f^{(1)}(x)\Delta x + \frac{f^{(2)}(x+\theta\Delta x)}{2!}(\Delta x)^2 \qquad \theta \in (0,1)$$

Αφού $f, f^{(1)}, f^{(2)}$ υπάρχουν και είναι και συνεχείς στο διάστημα που περικλείει τα $x, x+\Delta x$

Αν $f^{(1)}(x)$ πολύ μεγάλο $\to$ σφάλμα θα είναι πολύ μεγάλο

"

$$\frac{\tilde{y} - y}{y} = \boxed{\frac{f'(x)\, x}{f(x)}}\, \frac{\Delta x}{x} + O((\Delta x)^2)$$

→ Δείκτης κατάστασης προβλήματος.

Αν $|\Delta x| << 1$ τότε ο $\left|\frac{f'(x)\, x}{f(x)}\right|$ θα έχει μικρή σχετική αλλαγή y για μικρή σχετική αλλαγή στο x

Αν για κάθε $x \in U$ υπάρχει κάποιο x^* κοντά στο x τ.ω. το $f_{prog}(x)$ να είναι κοντά στο $f(x^*)$ $f_{prog}(x) \simeq f(x^*)$ τότε οι αλγόριθμοι που, κάνο-ποιούν αυτή τη συνθήκη ονομάζονται αριθμητικά ευσταθείς

<u>Προς τα πίσω ευστάθεια</u>: Αν $\exists\, x^* \to x$ τ.ω. $f_{prog}(x) = f(x^*)$, τότε ο αλγόριθμος χαρακτηρίζεται αριθμητικά προς τα πίσω ευσταθής

ΑΝΑΛΥΣΗ ΣΦΑΛΜΑΤΟΣ "ΠΡΟΣ ΤΑ ΕΜΠΡΟΣ"

Το πρόγραμμα που εκτελείται και παράγει το $f(x)$ αποτελείται από μια σειρά από στοιχειώδεις αριθμητικές πράξεις. Κάθε πράξη παράγει ένα νέο αποτέλεσμα που εξαρτάται από τα δεδομένα εισόδου ή και από τις τιμές που έχουν ήδη υπολογιστεί. Εφόσον όμως οι πράξεις εκτελούνται σε α.κ.υ. το αποτέλεσμα σε κάθε βήμα θα είναι διαφορετικό από αυτό που θα παίρναμε με τις πράξεις στο $\mathbb{R}$ άρα το υπολογιζόμενο $\tilde{z}$ είναι προσέγγιση του z και το σφάλμα υπακούει το κανόνα μετάδοσης που αναφέραμε πριν.

πράξη που αποφεύγεται	αντικατάσταση με:
o $a^2 - b^2$	$(a-b)(a+b)$
o $\sqrt{x+\varepsilon} - \sqrt{x}$	$\dfrac{\varepsilon}{\sqrt{x+\varepsilon}+\sqrt{x}}$
o $\cos\varepsilon - 1$	$-2\sin^2\frac{\varepsilon}{2}$
o $-b \pm \sqrt{b^2-4c}$	$\lvert b\rvert + \sqrt{b^2-4c}$ + και [illegible]
o $f(x+\varepsilon) - f(x)$	$f^{(1)}(x)\varepsilon + f^{(2)}(x)\cdot\frac{\varepsilon^2}{2} + \dots +$

Στην προς τα εμπρός ανάλυση έχουμε φορές του τύπου

$$P_n = \prod_{i=1}^{n}(1+\delta_i) \quad \text{οπου } |\delta_i| \le u$$

$$(1-u)^n \le P_n \le (1+u)^n.$$

Λήμμα 3.5.1

Αν $|\delta_i| \le u$ και $\rho_i = \pm 1$ για $i=1:n$ και $nu<1$ τότε

$$\boxed{\prod_{i=1}^{n}(1+\delta_i)^{\rho_i} = 1+\theta_n} \quad \text{οπου} \quad \boxed{|\theta_n| \le \frac{nu}{1-nu} = \gamma_n}$$

ΣΟΣ

- $\prod_{i=1}^{n}(1+\delta_i)^{\rho_i} = \langle n \rangle$

- $\langle n \rangle \times \langle k \rangle = \langle n+k \rangle$ και $\langle n \rangle / \langle k \rangle = \langle n+k \rangle$

απόδειξη $\prod_{i=1}^{n}(1+\delta_i) \cdot \prod_{j=n+1}^{k+n+1}(1+\delta_j)$

$$\theta_n \le \frac{n \cdot n}{1-nn} = \gamma_n = (1+\delta_1)(1+\delta_2) \quad (1+\delta_n)(1+\delta_{n+1}) \ldots (1+\delta_{n+k+1})$$

$$= \prod_{i=n}^{n+k}(1+\delta_i) \qquad \text{Ομοίως } \langle n \rangle \langle k \rangle = \langle n-k \rangle$$

- $\prod_{i=1}^{n}(1+\delta_i) \le (1+u)^n \le e^{nu}$

βλ. παράδειγμα 3.5.2.

ΑΝΑΛΥΣΗ ΣΦΑΛΜΑΤΟΣ "ΠΡΟΣ ΤΑ ΠΙΣΩ"

Έστω ότι $\exists\, x^* \to x$ ώστε $f_{prog}(x) = f(x^*)$

$\hat{z} = f_{prog}(x)$ και $z^* = f(x^*)$

$\|z^* - z\| = \|f_{prog}(x) - f(x)\| = \|f(x^*) - f(x)\|$ το ανάγουμε σε σφάλματα που οφείλεται στα στοιχεία εισόδου

αφού $f_{prog}(x) = f(x^*)$

Παράδειγμα 3.5.3

Έστω $f(x_1, x_2, x_3) = (x_1 + x_2) + x_3$ με στοιχεία εισόδου x_1, x_2, x_3

Τότε $f_{prog}(x_1, x_2, x_3) = \left((x_1 + x_2)(1+\delta_1) + x_3\right)(1+\delta_2) =$

$= x_1(1+\delta_1)(1+\delta_2) + x_2(1+\delta_1)(1+\delta_2) + x_3(1+\delta_2)$ όπου $|\delta_j| \le u$, $j=1$

$= f(\tilde{x}_1, \tilde{x}_2, \tilde{x}_3)$

$$\frac{|\tilde{x}_1 - x_1|}{|x_1|} = \frac{|x_1(1+\delta_1)(1+\delta_2) - x_1|}{|x_1|} = |(1+\delta_1)(1+\delta_2) - 1| = |\delta_1 + \delta_2 + \delta_1\delta_2| \le 3u$$

$$\frac{|\tilde{x}_2 - x_2|}{|x_2|} = \frac{|x_2(1+\delta_1)(1+\delta_2) - x_2|}{|x_2|} = |(1+\delta_1)(1+\delta_2) - 1| \le 3u$$

$$\frac{|\tilde{x}_3 - x_3|}{|x_3|} = \frac{|x_3(1+\delta_3) - x_3|}{|x_3|} = 1 + \delta_3 - 1 = \delta_3 \le u$$

Άρα $|f_{prog}(x_1, x_2, x_3) - f(x_1, x_2, x_3)| = |f(\tilde{x}_1, \tilde{x}_2, \tilde{x}_3) - f(x_1, x_2, x_3)|$

με $\dfrac{|\tilde{x}_i - x_i|}{|x_i|} \le 3u$ για $i = 1, 2$ και $\dfrac{|\tilde{x}_3 - x_3|}{|x_3|} \le u$

Ο αλγόριθμος υπολογισμού είναι προς τα πίσω ευσταθής