Θεωρία Τηλεπικοινωνιακής Κίνησης

Ενότητα 4: Δίκτυα Αναμονής και Λειτουργικοί Νόμοι

Μιχαήλ Λογοθέτης

Πολυτεχνική Σχολή

Τμήμα Ηλεκτρολόγων Μηχανικών
και Τεχνολογίας Υπολογιστών

Σκοποί ενότητας

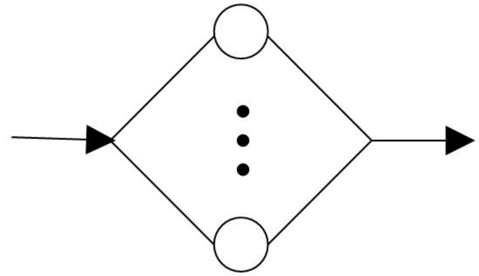
- Περιγραφή και ανάλυση (όχι μεμονωμένων συστημάτων αναμονής αλλά) δικτύων αναμονής.

Περιεχόμενα ενότητας

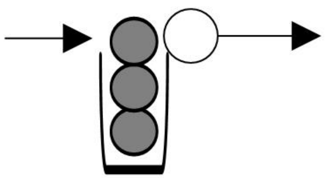
- Δίκτυα αναμονής
- Λειτουργικοί Νόμοι:
  - Ο νόμος της αξιοποίησης
  - Ο νόμος της εξαναγκασμένης ροής
  - Ο νόμος του Little
  - Ο γενικός νόμος του χρόνου απόκρισης
  - Ο νόμος της διαλογικής απόκρισης
- Ανάλυση κυκλοφοριακής συμφόρησης
- Παραδείγματα

Μεμονωμένα συστήματα εξυπηρέτησης

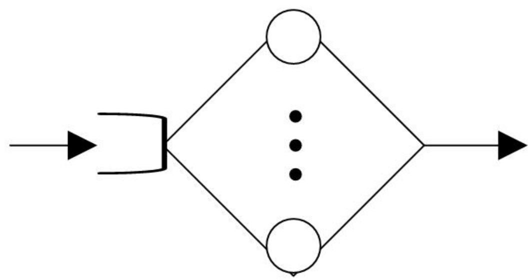
1 εξυπηρετητής
Ουρά αναμονής FIFO



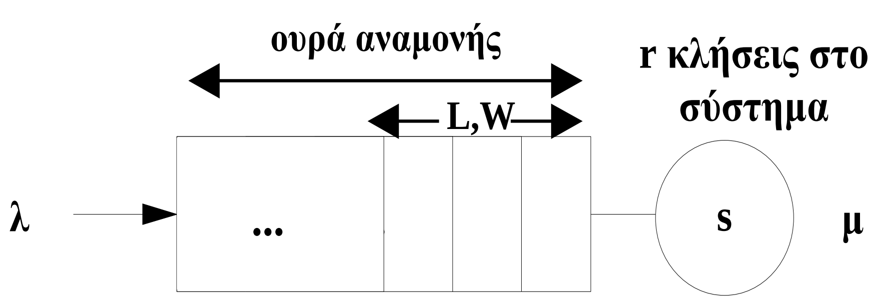
Κέντρο καθυστέρησης
Πολλοί εξυπηρετητές
χωρίς ουρά αναμονής



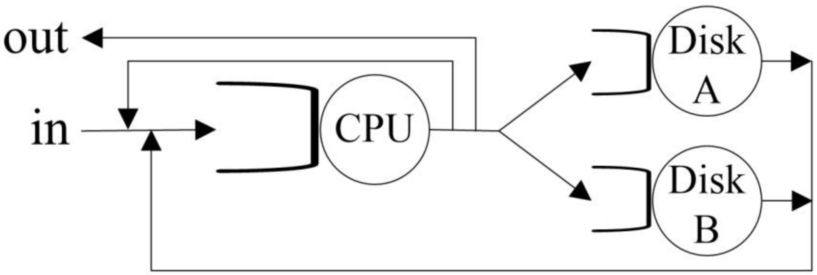
1 εξυπηρετητής
Ουρά αναμονής LIFO



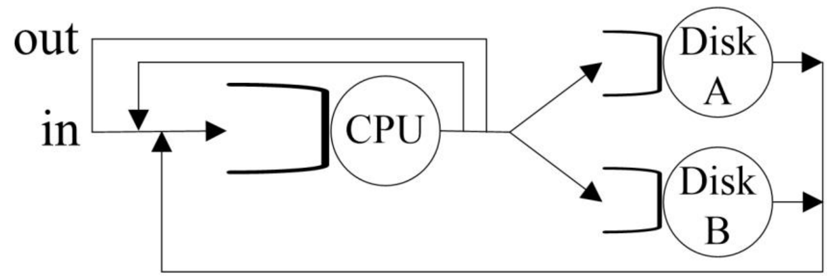
Ουρά αναμονής FIFO
με πολλαπλούς
εξυπηρετητές



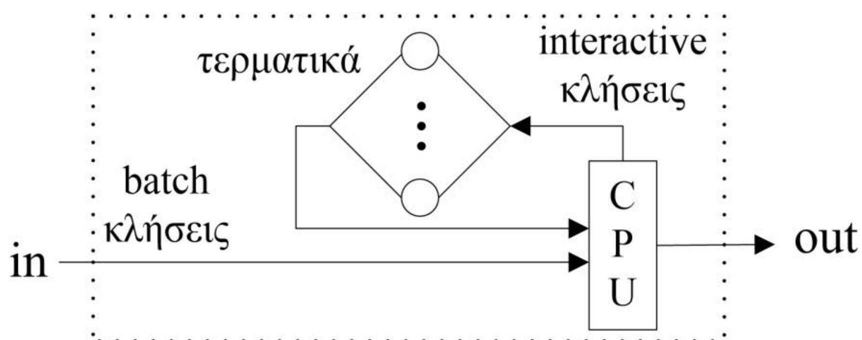
Δίκτυα αναμονής



(α) Ανοικτό Δίκτυο Αναμονής



(β) Κλειστό Δίκτυο Αναμονής



(γ) Μεικτό Δίκτυο Αναμονής

Ανοικτό για τις κλήσεις κατηγορίας *batch*

Κλειστό για τις κλήσεις κατηγορίας *interactive*

Θεώρημα του Burke

*Σε ένα σύστημα αναμονής **M/M/s** όπου s = 1, 2, …, ∞ με ρυθμό άφιξης λ:*

(i) *Η διαδικασία αναχώρησης είναι επίσης Poisson με ρυθμό λ.*

(ii) *Κάθε χρονική στιγμή t, ο αριθμός των κλήσεων στο σύστημα είναι ανεξάρτητος από τον αριθμό των αναχωρήσεων που έγιναν πριν από τον χρόνο t.*

Απόρροια:

- Δεν μπορούμε να βγάλουμε κάποιο συμπέρασμα για τον αριθμό των κλήσεων στο σύστημα, κάνοντας μετρήσεις επί του αριθμού των αναχωρήσεων.
- Αν έχομε ένα σύστημα αναμονής FIFO, M/G/*s* ή G/M/*s*, στο οποίο βρίσκουμε ότι οι αναχωρήσεις ακολουθούν την διαδικασία Poisson, τότε, G = M !

Θεώρημα του Jackson

(i) Σε ένα ανοικτό δίκτυο αναμονής, παρότι οι αφίξεις κλήσεων σε ένα (υπο-) σύστημα αναμονής μπορεί να μην είναι Poisson, το σύστημα αυτό στατιστικώς συμπεριφέρεται ωσάν σύστημα αναμονής M/M/s (δηλ. με αφίξεις Poisson).

(ii) Οποιοδήποτε ανοικτό δίκτυο αποτελούμενο από M υποσυστήματα με εκθετική κατανομή του χρόνου εξυπηρέτησης έχει λύση μορφής γινομένου (product form).

- Αν το δίκτυο περιλαμβάνει M υποσυστήματα (ουρές) και n_1, n_2, …, n_M είναι οι καταστάσεις των M υποσυστημάτων (αριθμός κλήσεων σε κάθε ουρά), η από κοινού πιθανότητα των κλήσεων αναμονής στις M ουρές, $P(n_1, n_2, \ldots, n_M)$, μπορεί να υπολογιστεί απλά πολλαπλασιάζοντας τις «ατομικές» πιθανότητες:

$$P(n_1, n_2, \ldots, n_M) = P_1(n_1) P_2(n_2) \ldots P_M(n_M)$$

Λειτουργικές Ποσότητες - Λειτουργικοί Νόμοι

Λειτουργική ποσότητα = απευθείας μετρήσιμη ποσότητα

Εντός χρόνου *T*, απευθείας μετρήσιμος είναι:

Ο αριθμός κλήσεων *A*, που φτάνουν στο σύστημα

$\Rightarrow$ Ρυθμός άφιξης κλήσεων, $\boldsymbol{\lambda = A / T}$ $(\lambda \sim c)$

Ο αριθμός κλήσεων *C*, που αναχωρούν από το σύστημα

$\Rightarrow$ Διεκπεραιωτική ικανότητα, $\boldsymbol{X = C / T}$ $(X \sim \mu)$

Το συνολικό χρονικό διάστημα *B*, που το σύστημα είναι κατειλημμένο

$\Rightarrow$ Αξιοποίηση του συστήματος, $\boldsymbol{U = B / T}$ $(U \sim \alpha_c)$

$\Rightarrow$ Μέσος χρόνος εξυπηρέτησης ανά κλήση, $\boldsymbol{S = B / C}$ $(S \sim h)$

Ο Λειτουργικός Νόμος της Αξιοποίησης U

- $\boldsymbol{U = B / T}$ $\Rightarrow$ $U = (C\,B) / (T\,C) = (C / T)\,(B / C)$ $\Rightarrow$ $\boldsymbol{U = X\,S}$ ***(utilization law)***

  Παράδειγμα: Σε ένα σύστημα *i* οι μετρήσεις έδειξαν ότι οι κλήσεις φτάνουν σ' αυτό με μέση τιμή 200 κλήσεις/s, ενώ το σύστημα χρειάζεται 4 ms/κλήση προκειμένου να τις εξυπηρετήσει. Χρησιμοποιώντας αρχικά το γνωστό μοντέλο αναμονής M/M/1 και στην συνέχεια τον νόμο της αξιοποίησης, να ευρεθεί η αξιοποίηση του συστήματος.

  – Βάσει του M/M/1: Η αξιοποίηση του συστήματος U ισούται με την μέση τιμή της διεκπεραιουμένης κίνησης. Δεδομένου ότι δεν υπάρχουν απώλειες κίνησης, $U = \alpha_c = \alpha = \lambda\,h = 200 \times 0.004 = 0.8$ $\Rightarrow$ $\boldsymbol{U = 80\,\%}$.

  – Βάσει του Νόμου της Αξιοποίησης: Αφού όλες οι κλήσεις θα εξυπηρετηθούν (μετά από αναμονή), $X = \lambda$. Οπότε: $U = X\,S = 200 \times 0.004 = 0.8$ $\Rightarrow$ $\boldsymbol{U = 80\,\%}$.

- Με το M/M/1 υποθέτουμε ότι οι χρόνοι άφιξης και εξυπηρέτησης των κλήσεων είναι τυχαίες μεταβλητές που ακολουθούν την εκθετική κατανομή.

- Ενώ, με τον Νόμο της Αξιοποίησης το αποτέλεσμα (*βάσει των μετρήσεων*) ισχύει προσεγγιστικά για **κάθε** διαδικασία άφιξης ή εξυπηρέτησης κλήσεων.

Ο Λειτουργικός Νόμος της Εξαναγκασμένης Ροής

- Διεκπεραιωτική ικανότητα ανοικτού δικτύου αναμονής ≡ Αριθμός αναχωρήσεων C στην μονάδα του χρόνου.
- Διεκπεραιωτική ικανότητα κλειστού δικτύου αναμονής ≡ Αριθμός των κλήσεων που διέρχονται από την εξωτερική ζεύξη στην μονάδα του χρόνου.
- Αν εντός ενός χρονικού διαστήματος T σε κάθε υποσύστημα i ισχύει $\boldsymbol{A_\mathrm{i} = C_i}$ (αφίξεις = αναχωρήσεις), τότε έχομε **εξισορροπημένη ροή**. Τότε ορίζουμε ως:
- **Επισκεψιμότητα (ρυθμό επισκέψεων – visit ratio**) $\boldsymbol{V_\mathrm{i}}$ **ενός υποσυστήματος** $\boldsymbol{i}$ ≡ το ποσοστό των εξυπηρετήσεων C_i που προσέφερε το υποσύστημα i ως προς τον αριθμό των εξυπηρετήσεων C_0 όλου του συστήματος ≡ την μέση τιμή των επισκέψεων που πραγματοποιεί μία κλήση στο υποσύστημα αυτό:

$$\boldsymbol{V_\mathrm{i} = C_i \,/\, C_0} \,.$$

- **Λειτουργικός νόμος της εξαναγκασμένης ροής**: Η διεκπεραιωτική ικανότητα του υποσυστήματος i, $\boldsymbol{X_\mathrm{i} = C_\mathrm{i} / T}$, και η συνολική διεκπεραιωτική ικανότητα όλου του συστήματος, $\boldsymbol{X = C_0 / T}$, συνδέονται με την επισκεψιμότητα $\boldsymbol{V_\mathrm{i}}$:

$$X_\mathrm{i} = C_\mathrm{i} / T \Rightarrow X_\mathrm{i} = C_\mathrm{i}\, C_0 / (C_0\, T) \Rightarrow \boldsymbol{X_\mathrm{i} = X\, V_\mathrm{i}}$$

Υποσύστημα κυκλοφοριακής συμφόρησης (bottleneck)

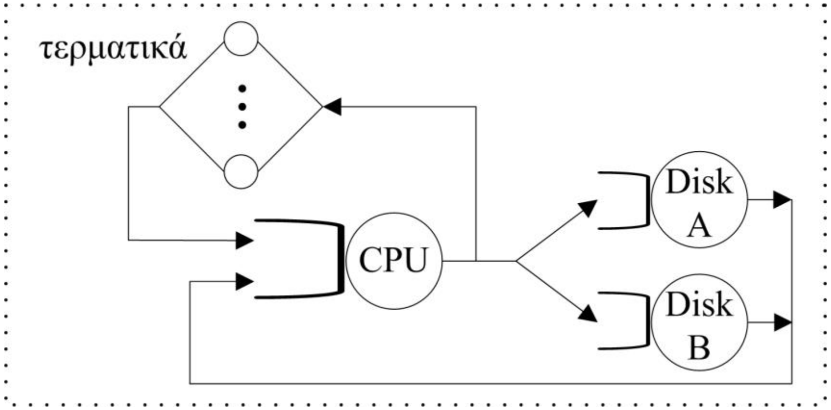
- Νόμος της Αξιοποίησης υποσυστήματος *i*: $U_i = X_i S_i$
- Νόμος της εξαναγκασμένης ροής στο υποσύστημα *i*: $X_i = X V_i$
- $U_i = X_i S_i \Rightarrow U_i = X V_i S_i \Rightarrow U_i = X D_i$, όπου:
- $D_i = V_i S_i$ είναι η <u>συνολική ζήτηση εξυπηρέτησης από το υποσύστημα *i*</u>

  Δηλ. ο συνολικός χρόνος εξυπηρέτησης (Demand), είναι πολλαπλάσιος του χρόνου εξυπηρέτησης μιας κλήσης, λόγω της επισκεψιμότητας.

  Η αξιοποίηση ενός υποσυστήματος i είναι ανάλογη της συνολικής ζήτησης εξυπηρέτησης D_i (από το υποσύστημα *i*).
- Το υποσύστημα *i* με την μεγαλύτερη ζήτηση εξυπηρέτησης D_i έχει και την μεγαλύτερη αξιοποίηση U_i και ονομάζεται **υποσύστημα κυκλοφοριακής συμφόρησης (bottleneck).**

Παράδειγμα υπολογισμού αξιοποίησης (1)

Στο **σύστημα χρονομερισμού** (time-sharing) του κατωτέρω σχήματος, έχομε τα ακόλουθα δεδομένα για τα προγράμματα των χρηστών: Κάθε πρόγραμμα απαιτεί 5 s από τον χρόνο της CPU και κάνει 80 αιτήματα εισόδου-εξόδου (I/O requests) στον δίσκο A και 100 αιτήματα I/O στον δίσκο B. Ο μέσος **χρόνος σκέψης** (think time) των χρηστών είναι 18 s. Από τις προδιαγραφές των υποσυστημάτων προσδιορίστηκε ότι ο δίσκος A χρειάζεται 50 ms για να ικανοποιήσει ένα αίτημα I/O, ενώ ο δίσκος B χρειάζεται 30 ms. Έχοντας 17 ενεργά τερματικά η διεκπεραιωτική ικανότητα του δίσκου A μετρήθηκε ότι είναι 15.70 αιτήματα I/O ανά δευτερόλεπτο. Ζητείται να ευρεθεί η διεκπεραιωτική ικανότητα όλου του συστήματος X και οι αξιοποιήσεις της CPU, U_{CPU}, και των δίσκων U_A και U_B.



Παράδειγμα υπολογισμού αξιοποίησης (2)

- Πρώτα υπολογίζουμε τις επισκεψιμότητες. Αφού 1 επίσκεψη στην CPU προκαλεί επισκέψεις στους δίσκους A και B: $V_A = 80$ και $V_B = 100 \Rightarrow$

  $V_{CPU} = V_A + V_B + 1 = 181$
- Βασικό βήμα της ανάλυσης, ο υπολογισμός της ζήτησης εξυπηρέτησης:

  $D_{CPU} = 5$ s, $D_A = V_A S_A = 80 \times 0.05 = 4$ s, $D_B = 100 \times 0.03 = 3$ s
- Από τον νόμο της εξαναγκασμένης ροής, υπολογίζουμε τις διεκπεραιωτικές ικανότητες των υποσυστημάτων:

  $X_A = X\,V_A \Rightarrow 15.70 = 80\,X \Rightarrow X = 15.7/80 \Rightarrow$ **$X = 0.1963$ κλήσεις/s**

  $X_{CPU} = X\,V_{CPU} = 0.1963 \times 181 = 35.48$ αιτήματα / s

  $X_B = X\,V_B = 0.1963 \times 100 = 19.63$ αιτήματα / s
- Από τον νόμο της αξιοποίησης υπολογίζουμε τις αξιοποιήσεις των συσκευών:

  $U_{CPU} = X\,D_{CPU} = 0.1963 \times 5 = 0.9815 \Rightarrow$ **$U_{CPU} = 98.15\%$**

  $U_A = X\,D_A = 0.1963 \times 4 = 0.7852 \Rightarrow$ **$U_A = 78.52\%$**

  $U_B = X\,D_B = 0.1963 \times 3 = 0.5889 \Rightarrow$ **$U_B = 58.89\%$**

Επισκεψιμότητες και πιθανότητες μετάβασης (visit ratio vs transition probabilities)

- Όταν γνωρίζουμε τις επισκεψιμότητες V_i των υποσυστημάτων μπορούμε να βρούμε την πιθανότητα μετάβασης p_{ij} από το υποσύστημα i στο j και αντιστρόφως.
- Αν p_{i0} δηλώνει την πιθανότητα μετάβασης από το υποσύστημα i σε έξοδο του συστήματος και ορίζοντας $V_0 = 1$ την επισκεψιμότητα της εξόδου, ισχύει (για M υποσυστήματα):

$$C_j = \sum_{i=0}^{M} C_i p_{ij} \quad (\text{διαιρώντας και τα δύο μέλη διά } C_0) \quad \Rightarrow \quad V_j = \sum_{i=0}^{M} V_i p_{ij} \qquad (\textit{σχέση } 1)$$

- **ΣΕ ΜΟΝΤΕΛΑ ΚΕΝΤΡΙΚΟΥ ΕΞΥΠΗΡΕΤΗΤΗ (1 CPU, υποσύστημα 1)**:

$p_{i1} = 1 \;\; \forall\, i \neq 1$ (οι κλήσεις από όλα τα υποσυστήματα επιστρέφουν στην CPU) και

$p_{ij} = 0 \;\; \forall\, i, j \neq 1$ (οι επισκέψεις στα υποσυστήματα γίνονται μόνο από την CPU)

Από (*σχέση* 1), αφού $p_{ij} = 0 \Rightarrow \boldsymbol{V_0 = 1 = V_1\, p_{i0}}$ (έξοδος $\equiv$ υποσύστημα 0) ή $\boldsymbol{V_1 = 1 / p_{i0}}$

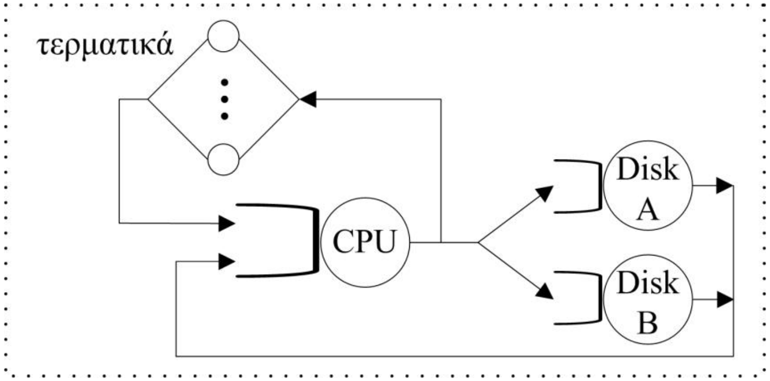
Λόγω της κεντρικής CPU $\Rightarrow \boldsymbol{V_1 = 1 + V_2 + V_3 + \ldots + V_M}$

Αφού η επίσκεψη στο υποσύστημα j γίνεται μόνο από την CPU $\Rightarrow \boldsymbol{V_j = V_1\, p_{1j} = p_{1j} / p_{i0}}$

Παράδειγμα μοντέλου κεντρικής CPU

Το ανοικτό δίκτυο του προηγουμένου παραδείγματος αποτελεί παράδειγμα με κεντρική CPU.

Να ευρεθούν οι πιθανότητες μετάβασης από την CPU στους δίσκους A , B και στην έξοδο (στα τερματικά).



Για V_A= 80 αιτήματα/s, V_B= 100 αιτήματα/s $\Rightarrow V_{CPU} = 1+V_A+V_B$= 181 αιτήματα/s

Η πιθανότητα μία κλήση να επισκεφθεί τον Disk A είναι: 80/181 = **0.4420**

Η πιθανότητα μία κλήση να επισκεφθεί τον Disk B είναι: 100/181 = **0.5525**

Η πιθανότητα μία κλήση να επισκεφθεί τα τερματικά (έξοδο) είναι: 1/181 = **0.0055**

Αν γνωρίζαμε τις πιθανότητες p_{1A}= 0.4420, p_{1B}= 0.5525 και την πιθανότητα εξόδου p_{10}= 0.0055, τότε: $V_A = p_{1A} / p_{10} = 80$, $V_B = p_{1B} / p_{10} = 100$ και $V_{CPU} = 1+V_A+V_B = 181$.

Ο Λειτουργικός Νόμος του Little

- Εννοούμε την επέκταση του νόμου του Little $N = \lambda\ T$

  Αν ισχύει η εξισορροπημένη ροή των κλήσεων (ο αριθμός των αφίξεων ισούται με τον αριθμό των αναχωρήσεων (εξυπηρετήσεων) από το (υπο-)σύστημα) τότε για το υποσύστημα i:

$$\boldsymbol{Q_{\mathrm{i}} = X_{\mathrm{i}}\, R_{\mathrm{i}}} \qquad (Q \sim N,\ X_{\mathrm{i}} \sim \lambda,\ T \sim R)$$

- Αν (στο προηγούμενο παράδειγμα) έχουμε Q_{CPU}=8.88 κλήσεις στην CPU, Q_{A}=3.19 κλήσεις στον Disk A και Q_{B}=1.4 κλήσεις στον Disk B κατά μέσον όρον και X_{CPU}=35.48 αιτήματα/s, X_{A}=15.70 αιτήματα/s και X_{B}=19.63 αιτήματα/s, να υπολογισθούν οι χρόνοι απόκρισης αυτών των υποσυστημάτων (R_{CPU}, R_{A} και R_B).

  Από τον Νόμος του Little $\Rightarrow$

$$R_{\mathrm{CPU}} = Q_{\mathrm{CPU}} / X_{\mathrm{CPU}} = 8.88 / 35.48 = \mathbf{0.250\ s}$$

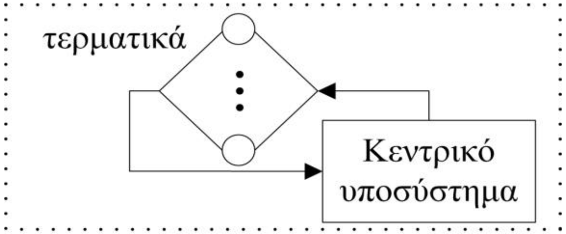
$$R_{\mathrm{A}} = Q_{\mathrm{A}} / X_{\mathrm{A}} = 3.19 / 15.70 = \mathbf{0.203\ s}$$

$$R_{\mathrm{B}} = Q_{\mathrm{B}} / X_{\mathrm{B}} = 1.4 / 19.63 = \mathbf{0.071\ s}$$

Ο γενικός νόμος του χρόνου απόκρισης

- Συστατικά μέρη συστημάτων χρονομερισμού:

Τερματικά και
Κεντρικό Υποσύστημα



- Νόμος του Little για όλο το σύστημα: $\boldsymbol{Q = X\,R}$
- Μετρώντας τα Q_i σε κάθε υποσύστημα i: $\boldsymbol{Q = Q_1 + Q_2 + \ldots + Q_M} \Rightarrow$

$$X\,R = X_1 R_1 + X_2 R_2 + \ldots + X_M R_M \Rightarrow$$

$$\boldsymbol{R = V_1 R_1 + V_2 R_2 + \ldots + V_M R_M}$$ **(γενικός νόμος του χρόνου απόκρισης)**

- Ισχύει και στην περίπτωση μη εξισορροπημένης ροής.
- Παράδειγμα: Στο σύστημα με την κεντρική CPU, όπου R_{CPU}= 0.250 s, R_{A}= 0.203 s, R_{B}= 0.071 s και V_{CPU}= 181 αιτήματα/s, R_{A}= 80 αιτήματα/s, R_{B}= 100 αιτήματα/s $\Rightarrow R = V_{CPU} R_{\mathrm{CPU}} + V_{\mathrm{A}} R_{\mathrm{A}} + V_{\mathrm{B}} R_{\mathrm{B}}$ = 45.25+16.24+7.1 $\Rightarrow$ $\boldsymbol{R}$**= 68.59 s**

Ο λειτουργικός νόμος της διαλογικής απόκρισης

- Σε ένα **διαλογικό σύστημα** (interactive system), οι χρήστες παράγουν αιτήματα τα οποία εξυπηρετούνται από το κεντρικό υποσύστημα και οι απαντήσεις γυρίζουν πίσω στο τερματικό. Μετά από ένα «χρονικό διάστημα σκέψης», *Z*, οι χρήστες υποβάλλουν ένα νέο **αίτημα** (request).
- Αν ο χρόνος απόκρισης του συστήματος είναι *R*, τότε ο συνολικός χρόνος του κύκλου αυτού των αιτημάτων και των απαντήσεων (cycle time of requests) είναι $R + Z$. Με δεδομένο ότι κάθε χρήστης παράγει $T / (R+Z)$ αιτήματα σε ένα χρονικό διάστημα *T*, τότε η διεκπεραιωτική ικανότητα του συστήματος για *N* χρήστες είναι:

$$X = \frac{\text{σύνολον αιτημάτων}}{\text{συνολικός χρόνος}} = \frac{N\left[\frac{T}{R+Z}\right]}{T} = \frac{N}{R+Z} \Rightarrow$$

$$R = \frac{N}{X} - Z$$

- *Η σχέση αυτή που συσχετίζει τον χρόνο R με τον αριθμό των χρηστών N και τον χρόνο σκέψης αποτελεί τον νόμο της διαλογικής απόκρισης. Ισχύει βέβαια και ότι:*

$$R = \sum_{i=1}^{M} R_i V_i$$

- Στο προηγούμενο παράδειγμα όπου X=0.1963, N=17, Z=18 ⇒

Ανάλυση κυκλοφοριακής συμφόρησης (1)

- Το πρώτο βήμα για να βελτιωθεί η απόδοση ενός συστήματος είναι να εντοπίσουμε το υποσύστημα κυκλοφοριακής συμφόρησης (bottleneck). Έστω ότι είναι το *b*. Τότε:

$$D_b = D_{\max} \quad \text{και} \quad U_b = X\, D_{\max}$$

- Η διεκπεραιωτική ικανότητα X και οι χρόνοι απόκρισης R του συστήματος συναρτήσει του αριθμού των χρηστών N περιορίζονται βάσει των εξής σχέσεων:

Ασυμπτωτικά όρια (asymptotic bounds)

$$X(N) \le \min\left\{\frac{1}{D_{\max}}, \frac{N}{D+Z}\right\}$$

$$R(N) \ge \max\left\{D, N\, D_{\max} - Z\right\} \quad \text{όπου } D = \sum D_i$$

- Η αξιοποίηση οποιουδήποτε υποσυστήματος δεν μπορεί να υπερβαίνει την μονάδα. Αυτή η παρατήρηση θέτει ένα όριο στη μέγιστη διεκπεραιωτική ικανότητα.

- Ο χρόνος απόκρισης του συστήματος με N χρήστες δεν μπορεί να είναι μικρότερος του συστήματος με 1 χρήστη. Έτσι τίθεται ένα όριο στον ελάχιστο χρόνο απόκρισης.

- Ο νόμος της διαλογικής απόκρισης μπορεί να χρησιμοποιηθεί για να μετατρέψει το όριο της διεκπεραιωτικής ικανότητας σε όριο του χρόνου απόκρισης και αντιστρόφως.

Ανάλυση κυκλοφοριακής συμφόρησης (2)

- Απόδειξη των ασυμπτωτικών ορίων

Αφού $U_b = X\,D_{\max} \le 1 \Rightarrow \boldsymbol{X \le 1/D_{\max}}$ (*σχέση* 1)

Αν στο σύστημα έχομε 1 κλήση, δεν σχηματίζεται ουρά και: $R(1) = \sum_{i=1}^{M} D_i = D$

Αν στο σύστημα έχομε $N > 1$ κλήσεις (χρήστες), τότε: $\boldsymbol{R(N) \ge D}$ (*σχέση* 2)

Από τον νόμο της διαλογικής απόκρισης και την (*σχέση* 1) $\Rightarrow$

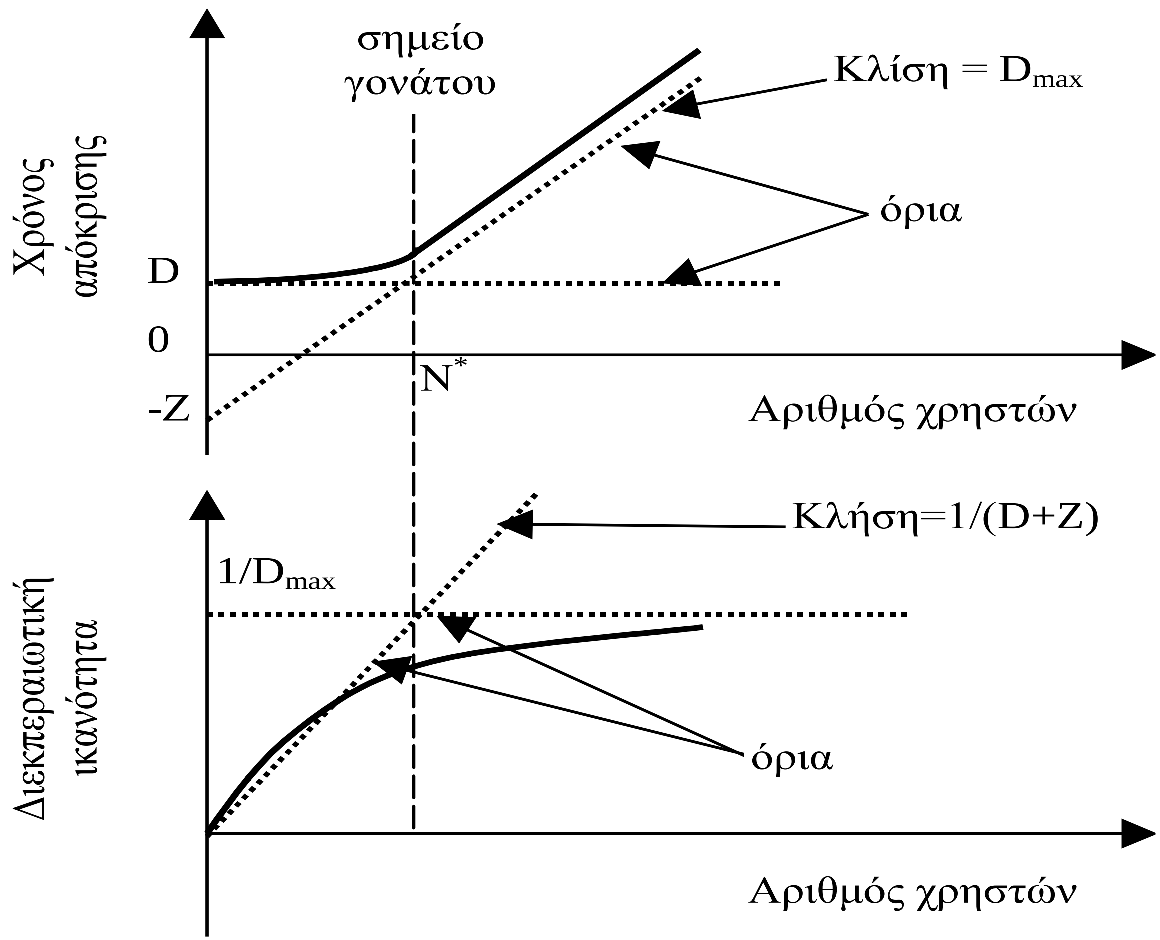
$R(N) = N/X(N) - Z \ge ND_{\max} - Z$ και από (*σχέση* 2) $\Rightarrow R(N) = \max\{D, ND_{\max} - Z\}$

Από τον νόμο της διαλογικής απόκρισης και την (*σχέση* 2) $\Rightarrow$

$X(N) = N/(R(N)+Z) \le N/(D+Z)$ και από (*σχέση* 1) $\Rightarrow X(N) = \min\{1/D_{\max}, N/(D+Z)\}$

Ανάλυση κυκλοφοριακής συμφόρησης (3)

- Γραφική παράσταση τυπικών ασυμπτωτικών ορίων



Ανάλυση κυκλοφοριακής συμφόρησης (4)

- **Παράδειγμα**

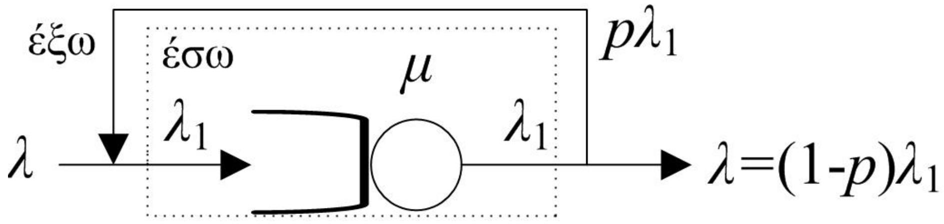
  Βάσει των προηγουμένων δεδομένων, $D_{CPU} = 5$, $D_A = 4$, $D_B = 3$ και $Z = 18$, υπολογίστε τα ασυμπτωτικά όρια και βρείτε τον αριθμό των τερματικών N που μπορούν να υποστηριχθούν από το σύστημα αυτό της κεντρικής CPU, αν απαιτείται ο χρόνος απόκρισης να διατηρηθεί μικρότερος των 100 s.

- $D_{max} = D_{CPU} = 5$ και $D = D_{CPU} + D_A + D_B = 5 + 4 + 3 = 12$. Οπότε:
- $X(N) = \min\{1/D_{max}, N/(D+Z)\} \Rightarrow \mathbf{X(N) = min\{1/5, N/30\}}$
- $R(N) = \max\{D, ND_{max} - Z\} \Rightarrow \mathbf{R(N) = max\{12, 5N - 18\}}$
- Από το σχήμα $\Rightarrow$ το «γόνατο» εμφανίζεται όταν $12 = 5N - 18 \Rightarrow N = 6$. Δηλ. αν έχουμε N > 6 χρήστες τότε θα εμφανιστεί ουρά αναμονής στο σύστημα.
- Θέλουμε $R(N) < 100 \Rightarrow R(N) = \max\{12, 5N - 18\} < 100 \Rightarrow 5N - 18 < 100 \Rightarrow$

  $\mathbf{N < 118/5 = 23.6}$ **χρήστες**. Δηλ. το σύστημα δεν μπορεί να υποστηρίξει περισσότερους από 23 χρήστες, αν απαιτείται ο χρόνος απόκρισης να διατηρηθεί κάτω από τα 100 s.

1ο Παράδειγμα Δικτύου Αναμονής (1)

- Γραμμή μετάδοσης δεδομένων (πακέτων) μοντελοποιείται (προσεγγιστικά) με σύστημα αναμονής M/M/1 με ανάδραση καθόσον ***p* = 30%** των πακέτων που μεταδίδονται πρέπει να ξαναμεταδοθούν λόγω σφαλμάτων. Ο εξωτερικός ρυθμός άφιξης είναι **λ = 0.5 πακέτα / sec**, ενώ η καθυστέρηση της μετάδοσης είναι ***h* = 1/*μ* = 0.75 sec**.



Να ευρεθούν:

- Ο συνολικός ρυθμός λ_1 άφιξης των πακέτων στο (έσω) σύστημα και ο συνολικός ρυθμός αναχώρησης των πακέτων από το σύστημα.
- Το διεκπεραιούμενο φορτίο κίνησης (αξιοποίηση *U* του εξυπηρετητή).
- Η επισκεψιμότητα (ρυθμός επισκέψεων) *V* στο (έσω) σύστημα.
- Η συνολική μέση καθυστέρηση *R* (Response Time) του συστήματος.

1^o Παράδειγμα Δικτύου Αναμονής (2)

- Από το Θεώρημα του Burke: ο συνολικός ρυθμός εξόδου (αναχώρησης των πακέτων) από το σύστημα είναι $\lambda = (1 - p)\, \lambda_1 = \mathbf{0.5}$ **πακέτα/s**.

  $\lambda = (1 - p)\, \lambda_1 = 0.5 \Rightarrow \lambda_1 = \lambda / (1 - p) = 0.5 / (1 - 0.3) = \mathbf{0.714}$ **πακέτα/s**.

- Το διεκπεραιούμενο φορτίο κίνησης ισούται με το προσφερόμενο:

  $A = \lambda / \mu = 0.714 \times 0.75 =$ **0.536 erl** ή $\boldsymbol{U = 5.36\ \%}$

- Σε χρόνο t έχουμε $\lambda_1 t$ επισκέψεις στο (έσω) σύστημα ενώ οι αναχωρήσεις είναι λt. Το πηλίκον του αριθμού των επισκέψεων προς τις αναχωρήσεις μας δίδει κατά μέσον όρον την επισκεψιμότητα V του συστήματος:

  $V = \lambda_1 / \lambda = \lambda_1 / (1 - p)\, \lambda_1 = 1 / (1 - p) = 1 / (1 - 0.3) \Rightarrow \boldsymbol{V = 1.4286}$

- Αν είχαμε <u>μία</u> μόνο επίσκεψη τότε η συνολική μέση καθυστέρηση R (<u>ανά επίσκεψη)</u> θα ήταν $R = T = 1 / (\mu - \lambda_1) = 1 / (1.3333 - 0.714) = 1.615$ s. Λόγω της «αυξημένης» επισκεψιμότητας όμως, η ζητουμένη R είναι:

  $R = \mathrm{V}\, T = 1.4286 \times 1.615 \Rightarrow \boldsymbol{R = 2.307}$ **s**

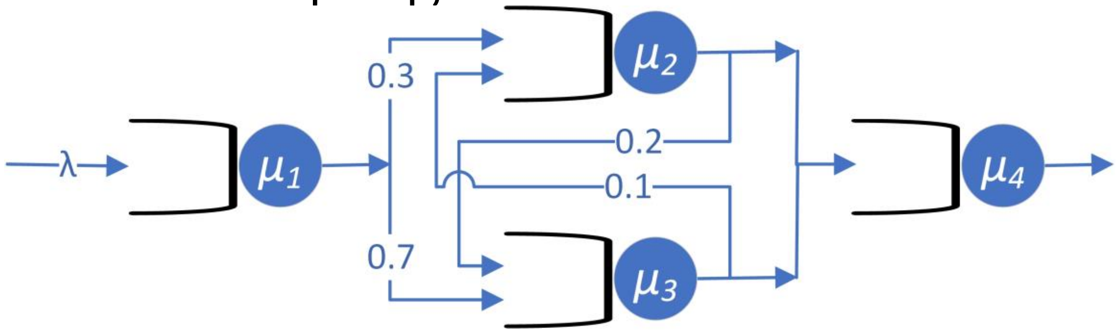
2ο Παράδειγμα Δικτύου Αναμονής (1)

Θεωρήστε την διαδικασία έκδοσης διαβατηρίου: Ο ενδιαφερόμενος να εκδώσει διαβατήριο προσέρχεται στο αρμόδιο γραφείο 1 (Αστυνομία) και λαμβάνει το έγγραφο της αίτησης που πρέπει να συμπληρώσει (service rate μ_1). Ακολούθως, αν έχει φωτογραφία (πιθανότητα 30%), προσέρχεται στο γραφείο 2 για έλεγχο της αίτησής του και των δικαιολογητικών (service rate μ_2), ενώ αν δεν έχει φωτογραφία (πιθανότητα 70%) πηγαίνει στο γραφείο 3 (φωτογραφικός θάλαμος). Αν στο γραφείο 2 κριθεί ότι η φωτογραφία δεν είναι η κατάλληλη (πιθανότητα 20%), πρέπει να βγάλει καινούργια φωτογραφία στο γραφείο 3 (service rate μ_3). Από το γραφείο 3, αν κάποιος αμφιβάλει για την ορθότητα της φωτογραφίας (πιθανότητα 10%), μπορεί να προσέλθει για έλεγχο στο γραφείο 2, ή αν δεν αμφιβάλει (πιθανότητα 90%) να προσέλθει στο γραφείο 4 για να καταθέσει όλα τα δικαιολογητικά και να λάβει το διαβατήριό του.

- Την ανωτέρω διαδικασία έκδοσης διαβατηρίου να την περιγράψετε με δίκτυο αναμονής, αν το μοντέλο εξυπηρέτησης κάθε γραφείου μπορεί διατυπωθεί με ουρά αναμονής M/M/1.
- Αν λ = **15.36** αιτήσεις ανά ώρα και ο μέσος αριθμός ενδιαφερομένων (αιτήσεων) ανά γραφείο είναι N_1= **0.0853**, N_2= **0.9665**, N_3= **0.9926** και N_4= **0.256**, να ευρεθεί ο (μέσος) χρόνος απόκρισης του κάθε γραφείου ξεχωριστά και ο (μέσος) χρόνος απόκρισης της όλης διαδικασίας έκδοσης διαβατηρίου. Επίσης, οι ρυθμοί εξυπηρέτησης μ_1, μ_2, μ_3, μ_4 και το μέσον μήκος της ουράς αναμονής σε κάθε γραφείο.

2ο Παράδειγμα Δικτύου Αναμονής (2)

- Το ζητούμενο δίκτυο αναμονής είναι:



- Οι ρυθμοί άφιξης των ενδιαφερομένων ανά sec στα γραφεία είναι:

– $\lambda_1 = \lambda$	$\lambda_1 = \lambda$	$\lambda_1 = \lambda$	$\lambda_1 = 15.36/3600 = 0.00427$
– $\lambda_2 = 0.3\lambda + 0.1\lambda_3$	$\lambda_2 = 0.3\lambda + 0.1\lambda_3$	$\lambda_2 = 0.37755\lambda$	$\lambda_2 = 0.00161$
– $\lambda_3 = 0.7\lambda + 0.2\lambda_2$	$\lambda_3 = 0.7\lambda + 0.06\lambda + 0.02\lambda_3$	$\lambda_3 = 0.77551\lambda$	$\lambda_3 = 0.00331$
– $\lambda_4 = \lambda$	$\lambda_4 = \lambda$	$\lambda_4 = \lambda$	$\lambda_4 = 0.00427$

- Ο (μέσος) χρόνος απόκρισης του κάθε γραφείου (νόμος Little):

$N_1 = \lambda_1 T_1 \Rightarrow T_1 = 0.0853/0.00427 = \mathbf{19.977\ s}$

$N_2 = \lambda_2 T_2 \Rightarrow T_2 = 0.9665/0.00161 = \mathbf{600.311\ s}$

$N_3 = \lambda_3 T_3 \Rightarrow T_3 = 0.9626/0.00331 = \mathbf{290.816\ s}$

$N_4 = \lambda_4 T_4 \Rightarrow T_4 = 0.256/0.00427 = \mathbf{59.953\ s}$

2ο Παράδειγμα Δικτύου Αναμονής (3)

- Μέσος χρόνος απόκρισης T της όλης διαδικασίας έκδοσης διαβατηρίου:

  $N = \lambda T \;\Rightarrow\; T = N/\lambda$ όπου $N = N_1+N_2+N_3+N_4 = 0.0853+0.9665+0.9926+0.256 = 2.3$

  Άρα T = 2.3/0.00427 = **538.64 s ≈ 9 h**

- Οι ρυθμοί εξυπηρέτησης (των γραφείων) μ_i, για $i = 1, 2, 3$ και 4:

  $T_i = 1/(\mu_i - \lambda_i) \Rightarrow (\mu_i - \lambda_i)\, T_i = 1 \;\Rightarrow \mu_i = (1 + \lambda_i\, T_i\,)/\, T_i \;\Rightarrow$

  μ_1 = (1 + 0.00427*19.977)/19.977 = **0.05433 αιτήσεις/s** (> 0.00427)

  μ_2 = (1 + 0.00161*600.311)/600.311 = **0.00328 αιτήσεις/s** (> 0.00161)

  μ_3 = (1 + 0.00331*290.816)/290.816 = **0.00675 αιτήσεις/s** (> 0.00331)

  μ_4 = (1 + 0.00427*59.953)/59.953 = **0.02095 αιτήσεις/s** (> 0.00427)

- Τα μήκη L_i των ουρών αναμονής στο γραφείο μ_i, για $i = 1, 2, 3$ και 4:

  $N_i = L_i + a_i \Rightarrow L_i = N_i - a_i = N_i - \lambda_i\,/\mu_i$ *π.χ.* L_1 = 0.0853 – 0.00427/0.05433 = **0.00671**

  ή $L_i = \lambda_i\, W_i = \lambda_i\,(T_i - h_i) = \lambda_i\,(T_i - 1/\mu_i)$ για $i = 1, 2, 3$ και 4. Έτσι:

  L_1 = 0.00427(19.977 – 1/0.05433) = **0.00671**

  L_2 = 0.00161(600.311 – 1/0.00328) = **0.47565**

  L_3 = 0.00331(290.816 – 1/0.00675) = **0.47223**

  L_4 = 0.00427(59.953 – 1/0.02095) = **0.05218**

Παράδειγμα Δικτύου Αναμονής (4)

- Επαλήθευση με την βοήθεια του online tool, **Qts** – Number of Nodes: 4

γ	μ	Servers	Node	Routing Table			
0,00422	0,05433	1	1	0	0,3	0,7	0
0	0,00328	1	2	0	0	0,2	0,8
0	0,00675	1	3	0	0,1	0	0,9
0	0,02095	1	4	0	0	0	0

System Performance Measures

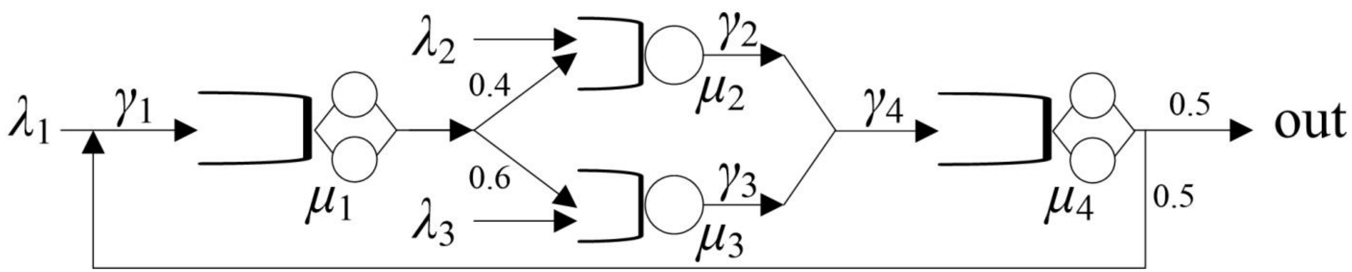
Total number in the network (L)	2,270913
Total sojourn time through the network (W)	531,8297

Node Performance Measures

Node	1	2	3	4
γ	0,00427	0	0	0
μ	0,05433	0,00328	0,00675	0,02095
Servers	1	1	1	1
λ	0,00427	0,001612	0,003311	0,00427
ρ	7,86%	49,15%	49,06%	20,38%
L	0,085298	0,966595	0,963025	0,255995
Lq	0,006704	0,475088	0,472443	0,052177
W	19,97603	599,5717	290,8184	59,95204
Wq	1,569992	294,6937	142,6703	12,21934

3ο Παράδειγμα Δικτύου Αναμονής (1)

- Στο κατωτέρω ανοικτό δίκτυο αναμονής, οι εξωτερικές αφίξεις των κλήσεων είναι Poisson με $\lambda_1 = 6$, $\lambda_2 = 3$ και $\lambda_3 = 5$ sec^{-1}. Το 1ο (υπο-)σύστημα αναμονής έχει 2 εξυπηρετητές οι οποίοι εξυπηρετούν, έκαστος, με ρυθμό $\mu_1 = 20$ κλήσεις/s. Οι εξερχόμενες απ' αυτό κλήσεις, σε ποσοστό 40% εξυπηρετούνται από το 2ο (υπο-)σύστημα αναμονής που έχει 1 εξυπηρετητή και εξυπηρετεί $\mu_2 = 13.75$ κλήσεις/s, ενώ σε ποσοστό 60% εξυπηρετούνται από το 3ο (υπο-)σύστημα αναμονής που έχει 1 εξυπηρετητή και εξυπηρετεί $\mu_3 = 34$ κλήσεις/s. Οι εξερχόμενες κλήσεις από το 2ο και 3ο υποσύστημα εισέρχονται σε 4ο, πανομοιότυπο με το 1ο υποσύστημα: $\mu_4 = \mu_1$. Μετά την εξυπηρέτηση από το 4ο υποσύστημα, οι κλήσεις κατά 50% εγκαταλείπουν το σύστημα και κατά 50% επιστρέφουν στο 1ο υποσύστημα. Βρείτε:



(a) Την αξιοποίηση του κάθε υποσυστήματος.

(b) Την πιθανότητα να έχομε συνολικά 4 κλήσεις στο σύστημα και μάλιστα να ευρίσκονται όλες στο τελευταίο στάδιο εξυπηρέτησης (δηλ. στο 4ο υποσύστημα).

(c) Την πιθανότητα να έχομε από 1 κλήση στα υποσυστήματα 2 και 3, ενώ το 1ο και το 4ο υποσύστημα να είναι άδεια.

3^o Παράδειγμα Δικτύου Αναμονής (2)

(a) Με επισκόπηση του σχήματος σχηματίζουμε το ακόλουθο σύστημα εξισώσεων:

- $\gamma_1 = \lambda_1 + 0.5\gamma_4$
- $\gamma_2 = \lambda_2 + 0.4\gamma_1$
- $\gamma_3 = \lambda_3 + 0.6\gamma_1$
- $\gamma_4 = \gamma_2 + \gamma_3$

όπου γ_i είναι ο πραγματικός/ενεργός ρυθμός άφιξης των κλήσεων στο υποσύστημα i.

Με διαδοχικές αντικαταστάσεις επιλύουμε το ανωτέρω σύστημα:

- $\gamma_4 = \lambda_2 + 0.4\gamma_1 + \lambda_3 + 0.6\gamma_1 = 8 + \gamma_1 = 8 + 6 + 0.5\gamma_4 \Rightarrow \gamma_4 = 28$
- $\gamma_1 = 6 + 0.5\times 28 \Rightarrow \gamma_1 = 20$
- $\gamma_2 = 3 + 0.4\times 20 \Rightarrow \gamma_2 = 11$
- $\gamma_3 = 5 + 0.6\times 20 \Rightarrow \gamma_3 = 17$ (κλήσεις/s).

Οπότε, η αξιοποίηση U_i του κάθε υποσυστήματος i υπολογίζεται ως:

- $U_1 = \gamma_1/(2\mu_1) = 20/(2\times 20) = 1/2 \quad \Rightarrow U_1 = 50\%$.
- $U_2 = \gamma_2/\mu_2 = 11/13.75 = 0.8 \quad \Rightarrow U_2 = 80\%$.
- $U_3 = \gamma_3/\mu_3 = 17/34 = 1/2 \quad \Rightarrow U_3 = 50\%$.
- $U_4 = \gamma_4/(2\mu_4) = 28/(2\times 20) = 0.7 \quad \Rightarrow U_4 = 70\%$.

3ο Παράδειγμα Δικτύου Αναμονής (3)

(b) Σύμφωνα με το θεώρημα του Jackson, αναλύσουμε το κάθε υποσύστημα μεμονωμένα, και μετά συνθέτουμε την λύση για όλο το δίκτυο αναμονής, ως το γινόμενο των λύσεων των υποσυστημάτων του. Το 1ο και το 4ο υποσύστημα, τα χειριζόμαστε ως M/M/2, ενώ το 2ο και το 3ο υποσύστημα, ως M/M/1.

Η ζητουμένη πιθανότητα $P(0,0,0,4)$ θα είναι: $\boldsymbol{P(0, 0, 0, 4) = P_1(0)\, P_2(0)\, P_3(0)\, P_4(4)}$

- Για $\alpha = \gamma_1 h_1 = \gamma_1 / \mu_1 = 1$ erl και $s = 2$, έχομε:

$$P_1(0) = \left(\sum_{r=0}^{1} \frac{(1)^r}{r!} + \frac{(1)^2}{2!} \frac{2}{2-1}\right)^{-1} = \frac{1}{1+1+1} = 0.3333 = 33.33\%$$

- Για $\alpha = \gamma_2 h_2 = \gamma_2 / \mu_2 = 0.8$ erl και $s = 1$, έχομε:

$$P_2(0) = \left(\sum_{r=0}^{0} \frac{(0.8)^r}{r!} + \frac{(0.8)^1}{1!} \frac{1}{1-0.8}\right)^{-1} = \frac{1}{1+4} = 0.2 = 20\%$$

- Για $\alpha = \gamma_3 h_3 = \gamma_3 / \mu_3 = 0.5$ erl και $s = 1$, έχομε:

$$P_3(0) = \left(\sum_{r=0}^{0} \frac{(0.5)^r}{r!} + \frac{(0.5)^1}{1!} \frac{1}{1-0.5}\right)^{-1} = \frac{1}{1+1} = 0.5 = 50\%$$

3ο Παράδειγμα Δικτύου Αναμονής (4)

- Για $\alpha = \gamma_4 h_4 = \gamma_4 / \mu_4 = 28/20 = 1.4$ erl και $s = 2$, έχομε:

$$P_4(0) = \left(\sum_{r=0}^{1} \frac{(1.4\quad)^r}{r!} + \frac{(1.4\quad)^2}{2!}\frac{2}{2-1.4}\right)^{-1} = \frac{1}{1+1.4+3.2667} = 0.1765$$
$$= 17.65\%$$

και

$$P_4(4) = \frac{(1.4\quad)^2}{2!}\left(\frac{1.4}{2}\right)^{4-2} \times 0.1765 = 0.085 = 8.5\%$$

- Οπότε:

$P(0,0,0,4) = 0.3333 \times 0.2 \times 0.5 \times 0.085 = 0.0028 = 0.28\%$.

(c) Ομοίως, υπολογίζουμε: $P_2(1) = 0.8 \times 0.2 = 0.16$ και $P_3(1) = 0.5 \times 0.5 = 0.25$, οπότε η ζητουμένη πιθανότητα είναι:

$\boldsymbol{P(0, 1, 1, 0) = 0.3333 \times 0.16 \times 0.25 \times 0.1765 = 0.002353 = 0.2353\%}$.

3ο Παράδειγμα Δικτύου Αναμονής (4)

- Επαλήθευση με την βοήθεια του online tool, **Qts** – Number of Nodes: 4

γ	μ	Servers	Node	Routing	Table		
6	20	2	1	0	0,4	0,6	0
3	13,75	1	2	0	0	0	1
5	34	1	3	0	0	0	1
0	20	2	4	0,5	0	0	0

System Performance Measures

Total number in the network (L)	9.078431
Total sojourn time through the network (W)	0.648459

Node Performance Measures

Node	1	2	3	4
γ	6	3	5	0
μ	20	13.75	34	20
Servers	2	1	1	2
λ	20	11	17	28
ρ	50.00%	80.00%	50.00%	70.00%
L	1.333333	4	1	2.745098
Lq	0.333333	3.2	0.5	1.345098
W	0.066667	0.363636	0.058824	0.098039
Wq	0.016667	0.290909	0.029412	0.048039

Marginal Probabilities

0	0.333333	0.200000	0.500000	0.176471
1	0.333333	0.160000	0.250000	0.247059
2	0.166667	0.128000	0.125000	0.172941
3	0.083333	0.102400	0.062500	0.121059
4	0.041667	0.081920	0.031250	0.084741
5	0.020833	0.065536	0.015625	0.059319
6	0.010417	0.052429	0.007813	0.041523
7	0.005208	0.041943	0.003906	0.029066
8	0.002604	0.033554	0.001953	0.020346
9	0.001302	0.026844	0.000977	0.014242
10	0.000651	0.021475	0.000488	0.009970

Τέλος Ενότητας