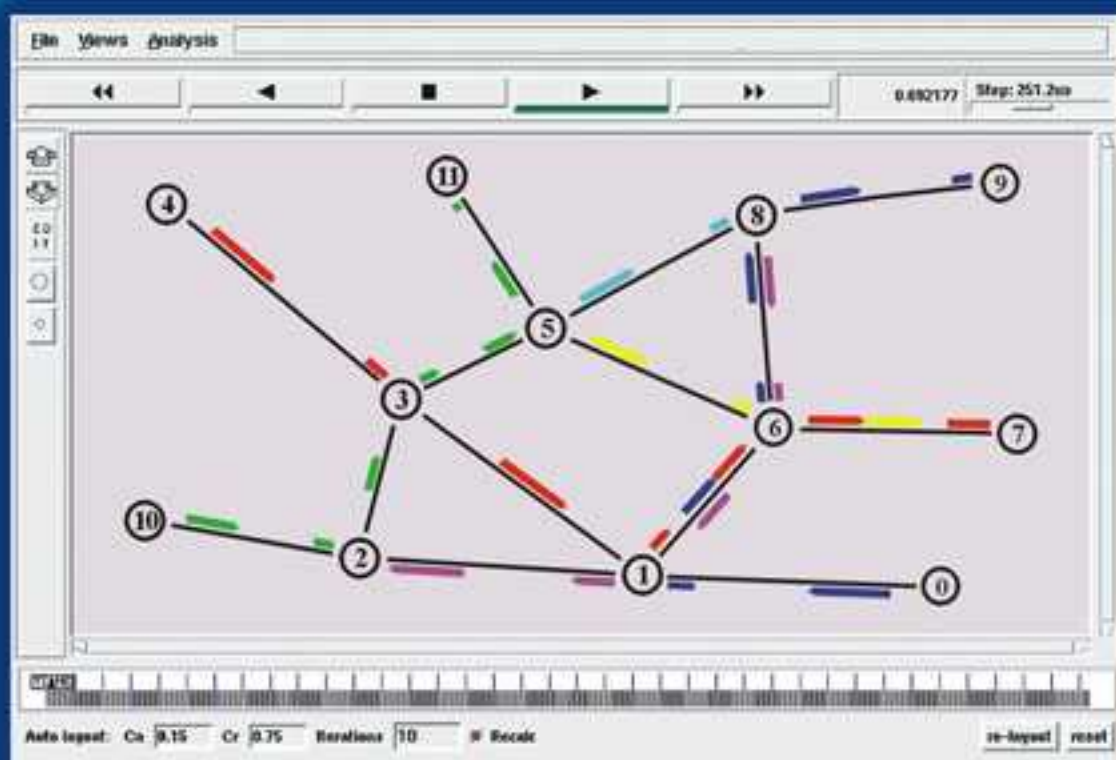


Εργαστηριακά Μαθήματα στα Δίκτυα και Διαδίκτυα Υπολογιστών



Βασίλειος Τσαουσίδης
Ελευθέριος Μαμάτας
Ιωάννης Ψαρράς
Ευστράτιος Κοσμίδης
Στυλιανός Δημητρίου

Περιεχόμενα

Εισαγωγή	7
Κεφάλαιο 1 Εισαγωγή σε Έννοιες των Δικτύων Υπολογιστών.....	11
Κεφάλαιο 2 Αξιοπιστία.....	25
Κεφάλαιο 3 Αλγόριθμοι Πολλαπλής Πρόσβασης.....	65
Κεφάλαιο 4 Μεταγωγή Δεδομένων και Δρομολόγηση.....	101
Κεφάλαιο 5 Διαχείριση Ουρών.....	141
Κεφάλαιο 6 Έλεγχος Ροής/Συμφόρησης.....	165
Κεφάλαιο 7 Δρομολογητές — Διαχείριση και Ρυθμίσεις	203
Παράρτημα	251

Εισαγωγή σε Έννοιες των Δικτύων Υπολογιστών

Θεωρήστε ένα δίκτυο υπολογιστών με 100 χρήστες και 100 παραλήπτες. Θεωρήστε ότι το δίκτυο μεταφέρει δεδομένα με ταχύτητα 100 Mbps. Θεωρήστε ότι όλοι οι χρήστες εκτελούν κάποια εφαρμογή, από μία ο καθένας. Θα πετύχουν όλοι οι χρήστες την ίδια απόδοση; Θα είναι δίκαιη η κατανομή των πόρων στους χρήστες; Θα έχουν όλοι οι χρήστες την ίδια επιβάρυνση σε διαχειριστικές πληροφορίες; Θεωρήστε τώρα ότι οι χρήστες χρησιμοποιούν διαφορετικά δίκτυα. Θα έχουν όλοι τις ίδιες συγκρούσεις; Θα έχουν τις ίδιες απώλειες λόγω σφαλμάτων;

Ο διαχειριστής δικτύου παρακολουθεί συχνά την κυκλοφορία πακέτων στο δίκτυο και ελέγχει κατά πόσον το δίκτυο πλησιάζει στη μέγιστη απόδοση. Εδώ αναφερόμαστε κυρίως στην **Ταχύτητα (Bandwidth)** και στην **Απόδοση (Throughput)**, δηλαδή στην απόδοση που έχει θεωρητικά το δίκτυο (π.χ. 10 Mbps) και την απόδοση που επιτυγχάνει τελικά (π.χ. 9 Mbps όταν το δίκτυο είναι φορτωμένο¹). Ο όρος Bandwidth συνήθως αναφέρεται ως "εύρος ζώνης", αφού έχει τις καταβολές του στο εύρος συχνοτήτων. Παρόλα αυτά, λόγω της συσχέτισης του εύρους ζώνης με τη χωρητικότητα (όπως θα εξηγήσουμε παρακάτω) και κατ' επέκταση με την ταχύτητα, συχνά αναφέρεται και ως **ταχύτητα**. Το "Throughput" (απόδοση ή διεκπεραιωτική ικανότητα) είναι αρκετά παρεξηγημένη έννοια. Μπορεί κανείς να το μετρήσει για μια συγκεκριμένη εφαρμογή (application throughput), ή μπορεί να το μετρήσει σε διαφορετικά σημεία του Διαδικτύου. Υποθέστε, για παράδειγμα, ότι ένα δίκτυο συνδέεται με κάποιο άλλο μέ-

¹ Κανονικά η Απόδοση μετριέται σε Bytes per second (Bps).

σω ενός δρομολογητή. Σε κάποια φάση της λειτουργίας του, ο δρομολογητής γεμίζει τον καταχωρητή του και αναγκάζεται να 'ρίξει' (να απορρίψει) μερικά πακέτα. Αν μετρήσουμε την κυκλοφορία σε ένα σημείο του Διαδικτύου που βρίσκεται μετά το δρομολογητή, θα διαπιστώσουμε μικρότερη κυκλοφορία απ' ό,τι αν τη μετρούσαμε στον αποστολέα.

Εκτός από αυτή τη διάκριση, πρέπει να λάβουμε υπόψη μας και το γεγονός ότι πολλά πακέτα χάνονται και ξαναστέλλονται από τους αποστολείς. Αυτό σημαίνει ότι μόνο ένα μέρος από τα πακέτα που κυκλοφορούν στο Διαδίκτυο είναι 'χρήσιμα' πακέτα για τη δικτυακή εφαρμογή. Και όμως, αυτό που μας ενδιαφέρει τελικά ως χρήστες είναι πόσο καλά 'τρέχει' η εφαρμογή μας, δηλαδή τι ροή χρήσιμων δεδομένων υπάρχει. Προκειμένου να προσδιορίσουμε με σαφήνεια αυτή τη λειτουργία/κατάσταση, εισάγουμε μια καινούργια μονάδα μέτρησης που την ονομάζουμε **Πραγματική Απόδοση (Goodput)**, η οποία μετράει ακριβώς την ίδια ροή δεδομένων που μετράει και η Απόδοση, πλην των επιπλέον Bytes της επικεφαλίδας (header overhead) και της αναμετάδοσης δεδομένων (retransmission overhead).

Συνολικά ορίζουμε τις παρακάτω μονάδες:

- ✱ **Ταχύτητα (Bandwidth):** (bps) Η θεωρητική ταχύτητα ροής δεδομένων, ή διαφορετικά η 'χωρητικότητα' μιας τομής του καναλιού σε bit. Μετριέται σε bit ανά δευτερόλεπτο (bit per second, bps).
- ✱ **Απόδοση (Throughput):** (Bps) Η μετρήσιμη ταχύτητα ροής, η μέγιστη ροή, όταν πάντα υπάρχουν δεδομένα προς αποστολή κατά το διάστημα της μέτρησης. Περιλαμβάνει και την **επιβάρυνση (overhead)** και μετριέται σε Byte ανά δευτερόλεπτο (Byte per second, Bps)

$$\text{Απόδοση} = (\text{Σταλθέντα δεδομένα σε Byte}) / \text{Χρόνος}$$

$$\text{Απόδοση Εφαρμογής} = (\text{Ληφθέντα δεδομένα σε Byte}) / \text{Χρόνος}$$

- ✱ **Πραγματική Απόδοση (Goodput):** (Bps) Η Απόδοση (Throughput) χωρίς τις κεφαλίδες και την επιβάρυνση (overhead) λόγω αναμετάδοσης δεδομένων. Πρόκειται για τα 'καθαρά' δεδομένα που παραλαμβάνει η εφαρμογή.

$$\text{Πραγματική Απόδοση} = \text{Απόδοση} - (\text{Επιβάρυνση} / \text{Χρόνος})$$

- ✱ **Επιβάρυνση (Overhead):** (B) Οι επιπλέον διαχειριστικές πληροφορίες, καθώς και τα δεδομένα που ξαναστέλλονται από τους αποστολείς.

$$\text{Επιβάρυνση} = (\text{Επιβάρυνση Κεφαλίδας}) + (\text{Επιβάρυνση Αναμετάδοσης})$$

Σχετική επίσης με την απόδοση ενός πρωτοκόλλου είναι και η **καθυστέρηση (Delay)**. Η καθυστέρηση απαρτίζεται από επιμέρους μονάδες.

- * **Καθυστέρηση Διάδοσης (Propagation Delay)** (ms) είναι η καθυστέρηση του σήματος μέχρι να φτάσει στον παραλήπτη, και έχει να κάνει με την απόσταση και την ταχύτητα του φωτός πάνω στο συγκεκριμένο μέσο μεταφοράς.
- * **Καθυστέρηση Μετάδοσης / Μεταβίβασης (Transmission Delay)** (ms) είναι η καθυστέρηση που οφείλεται στην ταχύτητα μετάδοσης δεδομένων πάνω από το κανάλι, και σχετίζεται με την ποσότητα των δεδομένων και την ταχύτητα μετάδοσης του καναλιού.

Καθυστέρηση Μετάδοσης = (Μεταδιδόμενα Δεδομένα)/(Εύρος Ζώνης)

- * **Καθυστέρηση Αναμονής (Queuing Delay)** (ms) είναι η καθυστέρηση που οφείλεται στην αναμονή στους **καταχωρητές ή περιοχές προσωρινής αποθήκευσης (buffers)** των δρομολογητών λόγω της πολύπλεξης.
- * **Καθυστέρηση Επεξεργασίας (Processing Delay)** (ms) είναι η καθυστέρηση που οφείλεται στην προσθήκη των κεφαλίδων, στην υλοποίηση του δικτύου, και στην ενδοεπικοινωνία διαφορετικών πρωτοκόλλων μέχρι να διοχετεύσουν τα δεδομένα στο δίκτυο. Τελικά

Καθυστέρηση = (Καθυστέρηση Διάδοσης) + (Καθυστέρηση Μετάδοσης)
+ (Καθυστέρηση Αναμονής) + (Καθυστέρηση Επεξεργασίας)

Από τη συζήτησή μας προκύπτουν δύο εύλογα ερωτήματα.

1. Ποιο είναι το πιο σημαντικό κομμάτι της καθυστέρησης;
2. Ποια μονάδα εκφράζει καλύτερα τη δυναμικότητα ενός δικτύου (ποια έχει μεγαλύτερη σημαντικότητα), η ταχύτητα (Bandwidth) ή η καθυστέρηση (Delay);

Όπως αναφέραμε προηγουμένως, η καθυστέρηση διάδοσης δεν σχετίζεται με την ποσότητα δεδομένων, και η καθυστέρηση μετάδοσης δεν σχετίζεται με την απόσταση. Η καθυστέρηση διάδοσης μπορεί να είναι της τάξεως των 5-50 ms (στα δορυφορικά δίκτυα μπορεί να φτάσει και τα 500 ms). Συγκριτικά, μπορούμε να υπολογίσουμε ότι η καθυστέρηση μετάδοσης για 1 KB πάνω από ένα κανάλι 1 Mbps είναι $(1 \cdot 1024 \cdot 8 / 1000000)$, δηλαδή περίπου 8 ms. Επομένως, η σημαντικότητα της καθυστέρησης μετάδοσης αυξάνεται όσο αυξάνεται η ποσότητα δεδομένων ή όσο μειώνεται η ταχύτητα μετάδοσης. Με παρόμοιο συλλογισμό μπορούμε να απαντήσουμε και στο δεύτερο ερώτημα.

Συχνά, στην πράξη αναφερόμαστε σε ένα συνδυασμό της καθυστέρησης και της ταχύτητας, το λεγόμενο "γινόμενο καθυστέρησης και εύρους ζώνης" (Delay x Bandwidth Product). Αυτό εκφράζει τη **χωρητικότητα (Capacity)** του καναλιού, δηλαδή την ποσότητα δεδομένων που μπορούμε να στείλουμε συνεχόμενα μέχρι να φτάσουν τα δεδομένα στον παραλήπτη, ή μέχρι να έχουμε επιβεβαίωση στον αποστολέα. Συνήθως, αντί για την καθυστέρηση χρησιμοποιούμε μια άλλη μονάδα, το **RTT (Round Trip Time)**.



Σχήμα 1-1 ► Χωρητικότητα ενός καναλιού.

ΑΣΚΗΣΗ 1: Σε ένα διαδίκτυο με ελάχιστη ταχύτητα (Bandwidth) 10Mbps, ένας αποστολέας στέλνει δεδομένα του 1 KB ξεκινώντας από 1 KB και με ρυθμό που αυξάνεται εκθετικά κάθε φορά που παίρνει επιβεβαίωση λήψης (2 KB, 4 KB, 8 KB, κ.λπ). Το διαδικτυακό πρωτόκολλο προσθέτει 40 Byte κεφαλίδα για κάθε KB. Ο αποστολέας έχει να στείλει συνολικά 20 KB και η καθυστέρηση διάδοσης (Propagation Delay) είναι 10 ms. Υπολογίστε την πραγματική απόδοση (Goodput) και την απόδοση (Throughput) στο τέλος της εφαρμογής.

ΛΥΣΗ: Ο αποστολέας ξεκινάει τη μετάδοση δεδομένων στέλνοντας 1 πακέτο του 1 KB. Το πακέτο αυτό φτάνει στον παραλήπτη και αυτός στέλνει επιβεβαίωση λήψης. Ο αποστολέας λαμβάνει την επιβεβαίωση και αυξάνει το ρυθμό του εκθετικά, δηλαδή αυτή τη φορά στέλνει 2 πακέτα του 1 KB το καθένα. Τα πακέτα αυτά στέλνονται συνεχόμενα (back-to-back), επομένως η καθυστέρηση διάδοσης (propagation delay) μετριέται μία φορά. Τα δεδομένα φτάνουν στον παραλήπτη, αυτός στέλνει πίσω την επιβεβαίωση λήψης, και ο αποστολέας αυξάνει το ρυθμό αποστολής του στέλνοντας αυτή τη φορά 4 πακέτα του 1 KB κ.ο.κ.. Στην άσκηση αυτή θεωρούμε ότι η κατάσταση του δικτύου είναι τέτοια ώστε δε χάνεται κανένα πακέτο δεδομένων ούτε λόγω συμφόρησης, ούτε λόγω σφάλματος κατά τη μετάδοση. Κάθε πακέτο που μπαίνει στο δίκτυο καθυστερεί κατά ένα χρονικό διάστημα ίσο με την καθυστέρηση μετάδοσης (transmission delay) και κατά ένα

χρονικό διάστημα ίσο με την καθυστέρηση διάδοσης (propagation delay). Ο ρυθμός μετάδοσης των πακέτων στο δίκτυο, καθώς και η καθυστέρηση που εισάγεται κάθε φορά, φαίνονται αναλυτικά στους ακόλουθους πίνακες.



Μετάδοση	Sender A	Transmission Link A-B	Receiver B
1η	P1 (1 KB) → ACK1	$0.8512+10=10.8512\text{ms}$ 10ms	P1 (1 KB) ← ACK1

2η	P2 (2 KB) → ACK2	$2 \cdot 0.8512+10=11.7024\text{ms}$ 10ms	P2 (2 KB) ← ACK2
----	---------------------	--	---------------------

3η	P3 (4 KB) → ACK3	$4 \cdot 0.8512+10=13.4048\text{ms}$ 10 ms	P3 (3 KB) ← ACK3
----	---------------------	---	---------------------

4η	P4 (8 KB) → ACK4	$8 \cdot 0.8512+10=16.8096\text{ms}$ 10ms	P4 (8 KB) ← ACK4
----	---------------------	--	---------------------

5η	P5 (5 KB) → ACK5	$5 \cdot 0.8512+10=14.256 \text{ ms}$ 10 ms	P5 (5 KB) ← ACK5
----	---------------------	--	---------------------

Data	Transmission Delay	Total Delay
1 KB	$(1024+40) \cdot 8/10000000=0.8512 \text{ ms}$	$(10+10+0.8512)=20.8512 \text{ ms}$
2 KB	$2 \cdot 0.8512 \text{ ms}=1.7024\text{ms}$	21.7024 ms
4 KB	$4 \cdot 0.8512 \text{ ms}=3.4048\text{ms}$	23.4048 ms
8 KB	$8 \cdot 0.8512 \text{ ms}=6.8096\text{ms}$	26.8096 ms
5 KB	$5 \cdot 0.8512 \text{ ms}=4.256\text{ms}$	24.256 ms
20 KB	17.024 ms	117.024 ms

Όπως φαίνεται από τους πίνακες, μετά την τέταρτη προσπάθεια μετάδοσης, ο αποστολέας αντί να αυξήσει το ρυθμό μετάδοσης (16 KB), τον μειώνει στέλνοντας μόνο 5 πακέτα του 1 KB. Αυτό γίνεται επειδή δεν έχει να στείλει τόσα δεδομένα, αλλά πλέον του μένουν μόνο 5KB για να ολοκληρώσει την εφαρμογή του.

Τελικά η απόδοση (Throughput) και η πραγματική απόδοση (Goodput) υπολογίζονται ως εξής:

$$\text{Απόδοση} = (\text{Συνολικά Δεδομένα}) / (\text{Συνολικός χρόνος}) = \\ (20 \cdot 1064) / 117.024 = 181.843 \text{ KBps}$$

$$\text{Πραγματική Απόδοση} = (\text{Αρχικά Δεδομένα}) / (\text{Συνολικός χρόνος}) = \\ (20 \cdot 1024) / 117.024 = 175 \text{ KBps}$$

ΑΣΚΗΣΗ 2: Θεωρήστε ότι θα εκτελέσετε δύο εφαρμογές πάνω από ένα δίκτυο ταχύτητας 1 Mbps με καθυστέρηση διάδοσης μέχρι τον παραλήπτη 10 ms — πρώτα μία εφαρμογή Telnet με μέσο ρυθμό μετάδοσης 100 Bps, και κατόπιν μία εφαρμογή FTP (μεταφοράς αρχείων) με μέσο ρυθμό μετάδοσης 100 KBps. Τι θα διαλέγατε σε αυτές τις περιπτώσεις, το διπλασιασμό της ταχύτητας ή τη μείωση της καθυστέρησης διάδοσης στα 5 ms;

ΑΣΚΗΣΗ 3: Σε ένα διαδίκτυο με ελάχιστη ταχύτητα 15 Mbps, ένας αποστολέας στέλνει δεδομένα του 1 KB ξεκινώντας από 1 KB και με ρυθμό που αυξάνεται εκθετικά κάθε φορά που παίρνει επιβεβαίωση λήψης (2 KB, 4 KB, 8 KB, κ.λπ.). Το διαδικτυακό πρωτόκολλο προσθέτει 40 Byte κεφαλίδα για κάθε KB. Ο αποστολέας έχει να στείλει συνολικά 25 KB και η καθυστέρηση διάδοσης (Propagation Delay) είναι 10ms. Κατά την τέταρτη προσπάθεια μετάδοσης δεδομένων, τα δεδομένα χάνονται λόγω υπερφόρτωσης του δικτύου και το πρωτόκολλο ξεκινάει ξανά από την αρχή (1 KB). Υπολογίστε την πραγματική απόδοση (Goodput) και την απόδοση (Throughput) στο τέλος της εφαρμογής.

ΛΥΣΗ: Σύμφωνα με τα δεδομένα του προβλήματος, καταγράφουμε τη συνέχεια των γεγονότων στον επόμενο πίνακα και υπολογίζουμε την καθυστέρηση που προκαλεί το δίκτυο σε καθένα από τα πακέτα. Κατά τη λύση της άσκησης, θεωρούμε ότι για κάθε πακέτο που στέλνει ο αποστολέας περιμένει μία επιβεβαίωση λήψης από τον παραλήπτη. Η επιβεβαίωση λήψης είναι ένα πακέτο μεγέθους 40 Byte, δηλαδή είναι μόνο η κεφαλίδα του TCP. Γι' αυτόν το λόγο, στον επόμενο πίνακα υπολογίζουμε την καθυστέρηση μετάδοσης (Transmission Delay) για κάθε πακέτο. Η καθυστέρηση μετάδοσης είναι αμελητέα για τις επιβεβαιώσεις, καθώς τα 40 Byte εισάγουν καθυστέρηση μετάδοσης μικρότερη από 1 ms. Αντιθέ-

τως, όταν υπολογίζουμε την καθυστέρηση διάδοσης, υπολογίζουμε και την καθυστέρηση των επιβεβαιώσεων. Αυτό συμβαίνει επειδή κάθε πακέτο που μεταδίδεται στο δίκτυο εισάγει πάντα μια καθυστέρηση, ανεξάρτητα από το μέγεθός του

Data	Transmission Delay	Total Delay
1 KB	$(1024+40) \cdot 8 / 15.000.000 = 0.567 \text{ ms}$	$(10+10+0.567) = 20.567 \text{ ms}$
2 KB	$2 \cdot 0.567 = 1.134 \text{ ms}$	21.134 ms
4 KB	$4 \cdot 0.567 = 2.268 \text{ ms}$	22.268 ms
8 KB	$8 \cdot 0.567 = 4.536 \text{ ms}$	24.536 ms
1 KB	0.567 ms	20.567 ms
2 KB	1.134 ms	21.134 ms
4 KB	2.268 ms	22.268 ms
8 KB	4.536 ms	24.536 ms
3 KB	$3 \cdot 0.567 = 1.701 \text{ ms}$	21.701 ms
33KB	18.711ms	198.711ms

Τελικά, υπολογίζουμε τη συνολική καθυστέρηση για τη μετάδοση όλων των δεδομένων μέσω του δικτύου:

$$\text{Συνολική Καθυστέρηση Μετάδοσης} = (33 \cdot 1064 \cdot 8) / 15000000 = 18.726 \text{ ms}$$

Αφού γνωρίζουμε πλέον το συνολικό χρόνο που χρειάζεται το δίκτυο για να μεταδώσει τα δεδομένα, υπολογίζουμε την απόδοση και την πραγματική απόδοση:

$$\text{Απόδοση} = (\text{Συνολικά Δεδομένα}) / (\text{Συνολικός χρόνος}) = (33 \cdot 1064) / 198.711 = 176.698 \text{ KBps}$$

$$\text{Πραγματική Απόδοση} = (\text{Αρχικά Δεδομένα}) / (\text{Συνολικός Χρόνος}) = (25 \cdot 1024) / 198.711 = 128.83 \text{ KBps}$$

ΑΣΚΗΣΗ 4: Υπολογίστε το συνολικό χρόνο που απαιτείται για να μεταφέρουμε ένα αρχείο 1000 KB, σε κάθε μία από τις παρακάτω περιπτώσεις, υποθέτοντας για λόγους απλοποίησης ότι το RTT είναι σταθερό και ίσο με 100 ms, το μέγεθος του πακέτου είναι 1 KB δεδομένων, και απαιτείται αρχική εγκατάσταση σύνδεσης ίση με 2 RTT πριν αποσταλούν δεδομένα.

- α) Η ταχύτητα (Bandwidth) είναι 1.5 Mbps και τα πακέτα μπορούν να στέλνονται συνεχόμενα.
- β) Η ταχύτητα είναι 1.5 Mbps, αλλά μετά την αποστολή κάθε πακέτου πρέπει να περιμένουμε την επιβεβαίωση πριν από την αποστολή του επόμενου πακέτου.
- γ) Η ταχύτητα είναι 'άπειρη' και κατά τη διάρκεια ενός RTT μπορούμε να στείλουμε μέχρι 20 πακέτα.
- δ) Η ταχύτητα είναι 'άπειρη' και κατά τη διάρκεια του πρώτου RTT μπορούμε να στείλουμε ένα πακέτο, κατά τη διάρκεια του δεύτερου RTT μπορούμε να στείλουμε δύο πακέτα, κατά τη διάρκεια του τρίτου RTT μπορούμε να στείλουμε τέσσερα πακέτα, κ.ο.κ.

ΛΥΣΗ: Σε αυτή την άσκηση θεωρούμε ότι όλες οι καθυστερήσεις (καθυστέρηση διάδοσης και μετάδοσης) συμπεριλαμβάνονται μέσα στο RTT, επομένως δεν υπολογίζουμε χωριστά την καθυστέρηση διάδοσης ή μετάδοσης ανά περίπτωση, αλλά ενδιαφερόμαστε για τον αριθμό των RTT που απαιτούνται για την ολοκλήρωση της εφαρμογής.

ΣΗΜΕΙΩΣΗ

Σε όλη τη διάρκεια της άσκησης, θεωρούμε ότι δεν υπάρχουν σφάλματα στο κανάλι μετάδοσης ή ότι αυτά τα σφάλματα δεν επηρεάζουν την εφαρμογή.

- α) Σύμφωνα με την εκφώνηση, τα πακέτα μπορούν να στέλνονται συνεχόμενα (back-to-back), άρα η ταχύτητα του καναλιού (bandwidth) δίνεται για να υπολογίσουμε πόσα πακέτα «χωράνε» μέσα στο δίκτυο σε μια συγκεκριμένη χρονική στιγμή, δηλαδή πόσα πακέτα του 1 KB μπορούν να μεταδοθούν μέσα από ένα κανάλι 1.5 Mbps σε ένα δευτερόλεπτο.

Πόσα Byte αντιστοιχούν στο 1.5Mb; $(1,5 \cdot 10^6)/8 = 0.1875 \cdot 10^6$ Byte, ή αλλιώς $0.1875 \cdot 10^6/1024 = 183.1$ KB

Κατά συνέπεια, στο συγκεκριμένο κανάλι χωράνε 183.1 πακέτα το δευτερόλεπτο. Αυτό σημαίνει ότι σε ένα RTT μπορούν να μεταδοθούν $183.1 \cdot 0.1 = 18.31$ πακέτα. Επειδή δεν είναι δυνατό να μεταδοθεί επιτυχώς το ένα τρίτο του πακέτου και το υπόλοιπο τμήμα να μεταδοθεί στο επόμενο RTT, σε ένα RTT θα μεταδοθούν 18 πακέτα (18 KB).

Σε 55 RTT θα έχουν μεταδοθεί $55 \cdot 18 = 990$ KB, ή αλλιώς 990 πακέτα. Στο επόμενο RTT θα μεταδοθούν τα υπόλοιπα 10 KB (10 πακέτα). Επομένως, χρειάζονται 56 RTT για να μεταδοθεί το αρχείο των 1000 KB. Αν συμπεριλάβουμε και τα 2 RTT που χρειάζονται για την εγκατάσταση της σύνδεσης, τότε συνολικά απαιτούνται **58 RTT** ή αλλιώς **5800 ms** για την ολοκλήρωση της εφαρμογής.

- β) Σε αυτή την περίπτωση, κάθε πακέτο πρέπει να στέλνεται ξεχωριστά από το επόμενο (όχι back-to-back), και επίσης μετά από τη μετάδοση κάθε πακέτου πρέπει να περιμένουμε 1 RTT μέχρι να επιβεβαιωθεί η άφιξη των δεδομένων στον παραλήπτη. Με λίγα λόγια, σε αυτή την περίπτωση δεν μπορούμε να στείλουμε πολλά πακέτα σε μία μετάδοση, άλλα πρέπει να τα στέλνουμε ένα-ένα. Δηλαδή ο ρυθμός μετάδοσης είναι σταθερός και ίσος με 1 πακέτο ανά RTT, και η χωρητικότητα του καναλιού αποτελεί σε αυτή την περίπτωση άχρηστη πληροφορία. Με άλλα λόγια, ανεξάρτητα από τη χωρητικότητα του καναλιού, ο αποστολέας μπορεί να εκμεταλλευτεί μόνο 1 KB κάθε φορά που μεταδίδει δεδομένα. Έτσι χρειαζόμαστε 1000 RTT, συν ακόμα 2 RTT για την εγκατάσταση της σύνδεσης, δηλαδή 1002 RTT συνολικά. Τελικά χρειάζονται $1002 \cdot 100 \text{ ms} = 100200 \text{ ms}$, ή αλλιώς 100.2 δευτερόλεπτα.
- γ) Σε αυτό το ερώτημα, όπως και στο προηγούμενο, είναι καθορισμένος εξαρχής από την εκφώνηση ο ρυθμός μετάδοσης. Η χωρητικότητα του καναλιού δεν μας ενδιαφέρει. Κάθε φορά ο αποστολέας μεταδίδει 20 πακέτα, δηλαδή 20 KB δεδομένων. Επομένως, για να μεταδοθούν 1000 KB απαιτούνται $1000/20 = 50$ RTT. Αν προσθέσουμε και τα 2 RTT για την εγκατάσταση της σύνδεσης, τότε συνολικά χρειάζονται 52 RTT, ή αλλιώς $52 \cdot 100 = 5200 \text{ ms}$, δηλαδή 5.2 δευτερόλεπτα.
- δ) Σε αυτή την περίπτωση δεν μας ενδιαφέρει η χωρητικότητα του καναλιού, αλλά η εκφώνηση καθορίζει και πάλι το ρυθμό μετάδοσης του αποστολέα. Ο αποστολέας αυξάνει το ρυθμό αποστολής του εκθετικά, δηλαδή αρχικά στέλνει 1 KB, στο επόμενο RTT στέλνει 2 KB, στο επόμενο 4 KB, στο επόμενο 8 KB, κ.ο.κ.. Τελικά στο ένατο RTT θα στείλει 256 KB και συνολικά θα έχει στείλει $2^9 - 1 = 511$ KB. Στην επόμενη προσπάθεια μετάδοσης έχει τη δυνατότητα να μεταδώσει 512 KB. Παρόλα αυτά, η εφαρμογή απαιτεί τη μετάδοση 489 KB ακόμα. Έτσι στο επόμενο RTT ο αποστολέας θα μεταδώσει τα υπόλοιπα 489 KB και θα ολοκληρώσει την εφαρμογή του. Τελικά ο αποστολέας χρειάζεται 10 RTT για τη μετάδοση δεδομένων και 2 RTT για την εγκατάσταση της σύνδεσης. Συνολικά έχουμε 12 RTT ή $12 \cdot 100 \text{ ms} = 1200 \text{ ms} = 1.2$ δευτερόλεπτα.

ΑΣΚΗΣΗ 5: Δυο δίκτυα μεταφέρουν δεδομένα ως εξής: μεταδίδουν ένα πακέτο ανά RTT και το ένα χάνει ένα κάθε τρία πακέτα, ενώ το άλλο χάνει ένα κάθε πέντε πακέτα. Τα πακέτα που χάνονται ξαναστέλνονται άμεσα χωρίς επιπλέον καθυστέρηση. Αν θεωρήσουμε για κάθε πακέτο κεφαλίδα 40 Byte και ότι δεν ξαναχάνονται τα πακέτα που ξαναστέλνονται, υπολογίστε την Απόδοση (Throughput) και την Πραγματική Απόδοση (Goodput) σε κάθε περίπτωση.

ΛΥΣΗ: Θεωρούμε ότι κάθε πακέτο έχει συνολικό μέγεθος ίσο με $pl+od$ (payload+overhead, ωφέλιμα δεδομένα + επικεφαλίδα). Στην πρώτη περίπτωση στέλνουμε συνολικά 4 πακέτα (3 κανονικά και 1 αναμετάδοση) και παραλαμβάνονται 3 (2 κανονικά και 1 αναμετάδοση). Η συνολική καθυστέρηση για να στείλουμε αυτά τα 4 πακέτα είναι 4 RTT.

$$\text{Απόδοση} = (\text{Συνολικά Δεδομένα}) / (\text{Συνολικός χρόνος}) = (4 \cdot (pl + od)) / (4 \cdot RTT) = (pl + od) / RTT$$

$$\text{Πραγματική Απόδοση} = (\text{Αρχικά Δεδομένα}) / (\text{Συνολικός χρόνος}) = (3 \cdot pl) / (4 \cdot RTT) = (0.75 \cdot pl) / RTT$$

Ομοίως, στη δεύτερη περίπτωση στέλνουμε συνολικά 6 πακέτα (5 κανονικά και 1 αναμετάδοση) και παραλαμβάνονται 5 πακέτα (4 κανονικά και 1 αναμετάδοση). Η συνολική καθυστέρηση είναι 6 RTT.

$$\text{Απόδοση} = (\text{Συνολικά Δεδομένα}) / (\text{Συνολικός χρόνος}) = (6 \cdot (pl + od)) / (6 \cdot RTT) = (pl + od) / RTT$$

$$\text{Πραγματική Απόδοση} = (\text{Αρχικά Δεδομένα}) / (\text{Συνολικός χρόνος}) = (5 \cdot pl) / (6 \cdot RTT) = (0.833 \cdot pl) / RTT$$

Διαπιστώνουμε ότι, παρόλο που έχουμε και στις δύο περιπτώσεις ίδια Απόδοση, η Πραγματική Απόδοση είναι μεγαλύτερη στη δεύτερη περίπτωση. Αυτό συμβαίνει επειδή η Απόδοση περιλαμβάνει μόνο την κίνηση των δεδομένων, η οποία είναι και στις δύο περιπτώσεις 1 πακέτο ανά RTT και δε διακρίνει τα ωφέλιμα δεδομένα.

ΑΣΚΗΣΗ 6: Δύο χρήστες Α και Β είναι συνδεδεμένοι σε ένα δρομολογητή Δ μέσω δύο καναλιών 10 Mbps το καθένα. Η καθυστέρηση διάδοσης για το κάθε κανάλι είναι 20 ms. Ο δρομολογητής Δ είναι τύπου αποθήκευσης και προώθησης (store-and-forward) και επεξεργάζεται για 35 ms κάθε πακέτο που λαμβάνει, αφού παραλάβει και το τελευταίο bit του πακέτου. Ο δρομολογητής μπορεί να επεξεργάζεται πολλά πακέτα ταυτόχρονα. Υπολογίστε το συνολικό χρόνο που χρειάζεται για τη μεταφορά 10000 bit από τον Α στον Β.

- α) ως ένα μόνο πακέτο.
- β) ως δύο πακέτα των 5000 bit το καθένα, που στέλνονται το ένα μετά το άλλο συνεχόμενα (back-to-back), δηλαδή χωρίς καθόλου καθυστέρηση μεταξύ των δύο μεταδόσεων.

ΛΥΣΗ: Σε αυτή την άσκηση πρέπει να βρούμε πόση είναι η καθυστέρηση που εισάγεται από το δίκτυο σε ένα πακέτο των 10 Kbit και πόση σε ένα πακέτο των 5 Kbit. Ουσιαστικά, μας ενδιαφέρει η καθυστέρηση που εισάγουν τα δύο κανάλια στα δεδομένα που μεταφέρουν, δηλαδή η καθυστέρηση του καναλιού που συνδέει τον αποστολέα Α με το δρομολογητή Δ και στη συνέχεια η καθυστέρηση του καναλιού που συνδέει το δρομολογητή Δ με τον παραλήπτη Β. Η καθυστέρηση του πρώτου καναλιού θα καθορίσει πότε ο δρομολογητής Δ θα μεταδώσει τα δεδομένα προς το δεύτερο κανάλι, και η καθυστέρηση του δεύτερου καναλιού θα καθορίσει πότε τελικά θα φτάσουν τα δεδομένα στον παραλήπτη και θα ολοκληρωθεί η εφαρμογή. Από αυτή τη χρονική διαφορά (λόγω του διαφορετικού όγκου των δεδομένων) θα καταλήξουμε τελικά στον πιο αποδοτικό τρόπο μετάδοσης των 10 Kbit.

- α) Πιο συγκεκριμένα, μας ενδιαφέρει η καθυστέρηση μετάδοσης (transmission delay) των δεδομένων, αφού η καθυστέρηση διάδοσης (propagation delay) είναι ίδια ανεξάρτητα από το μέγεθος του πακέτου. Επίσης, σύμφωνα με την εκφώνηση, ο δρομολογητής Δ επεξεργάζεται κάθε πακέτο για 35 ms ανεξάρτητα από το μέγεθός του (αυτό δεν είναι και τόσο ρεαλιστικό), όποτε τελικά το μόνο που αλλάζει ανάμεσα στις δύο περιπτώσεις (ένα πακέτο των 10000 bit και δύο πακέτα των 5000 bit) είναι η διαφορετική καθυστέρηση μετάδοσης λόγω του διαφορετικού μεγέθους των πακέτων.

Καθυστέρηση Καναλιού Α - Δ:

Καθυστέρηση Μετάδοσης (Transmission Delay) = (Δεδομένα προς Μετάδοση)/(Χωρητικότητα Καναλιού) = (10000 bit)/(10000000 bps) = (1/1000) δευτερόλεπτα = 1 ms

Καθυστέρηση Διάδοσης (Propagation Delay) = 20 ms

Συνολική Καθυστέρηση = (Καθυστέρηση Μετάδοσης) + (Καθυστέρηση Διάδοσης) = 1 + 20 = 21 ms

Αν θεωρήσουμε ότι ο αποστολέας Α ξεκινάει τη μετάδοση στο χρόνο t_0 , τότε το τελευταίο bit του πακέτου έχει φτάσει στο δρομολογητή Δ τη χρονική στιγμή $t_1 = t_0 + 21$. Σύμφωνα με την εκφώνηση, ο δρομολογητής θα επεξερ-

γαστεί τα δεδομένα για τα επόμενα 35 ms και στη συνέχεια θα τα μεταδώσει στο κανάλι που συνδέει το δρομολογητή με τον παραλήπτη Β. Επομένως τα δεδομένα θα φύγουν από το δρομολογητή τη χρονική στιγμή $t_2 = t_1 + 35 = t_0 + 56\text{ms}$. Η καθυστέρηση στο δεύτερο κανάλι είναι ίδια με εκείνη στο πρώτο κανάλι, δηλαδή 21 ms. Τελικά τα δεδομένα θα φτάσουν στον παραλήπτη τη χρονική στιγμή $t_3 = t_2 + 21 = t_0 + 77\text{ms}$.

- β) Στην περίπτωση που μεταδίδουμε 2 πακέτα των 5 Kbit το καθένα, τότε με τον ίδιο τρόπο όπως και προηγουμένως βρίσκουμε ότι το ένα πακέτο θα φτάσει στο δρομολογητή Δ μετά από 20.5 ms.

Καθυστέρηση Καναλιού Δ - Β:

Καθυστέρηση Μετάδοσης (Transmission Delay) = (Δεδομένα προς Μετάδοση)/(Χωρητικότητα Καναλιού) = $(5000 \text{ bit})/(10000000 \text{ bps}) = (1/2000)$ δευτερόλεπτα = 0.5 ms

Καθυστέρηση Διάδοσης (Propagation Delay)=20 ms

Συνολική Καθυστέρηση=(Καθυστέρηση Μετάδοσης) + (Καθυστέρηση Διάδοσης) = $0.5 + 20 = 20.5 \text{ ms}$

Αν θεωρήσουμε, όπως και προηγουμένως, ότι η μετάδοση ξεκινάει τη χρονική στιγμή t_0 , τότε το πρώτο πακέτο θα φτάσει στο δρομολογητή τη χρονική στιγμή $t_1 = t_0 + 20.5\text{ms}$.

Σύμφωνα με την εκφώνηση, τα δύο πακέτα των 5 Kbit στέλνονται συνεχόμενα, άρα όσο «ταξιδεύει» το πρώτο πακέτο μέσα στο κανάλι μετάδοσης, συγχρόνως ακριβώς από πίσω του «ταξιδεύει» και το δεύτερο πακέτο. Με δεδομένο ότι η καθυστέρηση μετάδοσης είναι 0.5 ms για καθένα από τα πακέτα, συμπεραίνουμε ότι το δεύτερο πακέτο θα φτάσει στο δρομολογητή Δ 0.5ms μετά το πρώτο, δηλαδή τη χρονική στιγμή $t_2 = t_1 + 0.5 = t_0 + 21\text{ms}$.

Ακολουθώντας την ίδια πορεία όπως στο ερώτημα (α), βρίσκουμε ότι το πρώτο πακέτο θα φύγει από το δρομολογητή Δ τη χρονική στιγμή $t_3 = t_0 + 55.5 \text{ ms}$ και το δεύτερο πακέτο τη χρονική στιγμή $t_4 = t_0 + 56 \text{ ms}$, δηλαδή 0.5 ms μετά το πρώτο. Η καθυστέρηση των 0.5 ms προκλήθηκε από την επιπλέον καθυστέρηση μετάδοσης που προκάλεσε το κανάλι μετάδοσης από τον αποστολέα Α μέχρι το δρομολογητή Δ. Όπως και προηγουμένως, η συνολική καθυστέρηση που εισάγει το δεύτερο κανάλι είναι ίση με αυτή που εισάγει το πρώτο. Τελικά το πρώτο πακέτο θα φτάσει στον προορισμό του τη χρονική στιγμή $t_5 = t_0 + 76\text{ms}$. Το δεύτερο

πακέτο έχει και αυτό την ίδια καθυστέρηση με το πρώτο, δηλαδή 20.5 ms, και τελικά θα φτάσει στον παραλήπτη Β τη χρονική στιγμή $t_6 = t_0 + 76.5$ ms.

Όπως φάνηκε από την παραπάνω ανάλυση, στην περίπτωση που στέλνουμε δύο ξεχωριστά πακέτα η εφαρμογή ολοκληρώνεται κατά 0.5 ms ταχύτερα. Αυτό συμβαίνει επειδή το πρώτο πακέτο κερδίζει 0.5 ms για καθένα από τα δύο κανάλια που διασχίζει. Τα 0.5 ms, προφανώς, προέρχονται από την καθυστέρηση μετάδοσης που εξαρτάται από το μέγεθος του πακέτου, και είναι μικρότερη όσο λιγότερα είναι τα δεδομένα προς μετάδοση. Το δεύτερο πακέτο κερδίζει 0.5 ms μόνο από το δεύτερο κανάλι που διασχίζει. Αυτό συμβαίνει επειδή το δεύτερο πακέτο φτάνει στο δρομολογητή Δ την ίδια χρονική στιγμή που φτάνει και το πακέτο του πρώτου ερωτήματος, αλλά η καθυστέρηση που εισάγεται από το δεύτερο κανάλι είναι μικρότερη από εκείνη του πρώτου ερωτήματος, συνεπώς φτάνει στον προορισμό του νωρίτερα.

ΑΣΚΗΣΗ 7: Όπως φάνηκε από την προηγούμενη άσκηση, τελικά συμφέρει να μεταδίδουμε μικρότερα πακέτα δεδομένων, αφού έτσι κερδίζουμε λόγω της μικρότερης καθυστέρησης διάδοσης που εισάγεται από το κανάλι. Να κάνετε παρόμοια ανάλυση με την παραπάνω θεωρώντας ότι μεταδίδουμε 10000 bit ως 10 πακέτα των 1000 bit. Τελικά, μέχρι ποιο σημείο συμφέρει χρονικά να μεταδίδουμε περισσότερα πακέτα μικρότερου όγκου;

ΑΣΚΗΣΗ 8: Ένα κανάλι 'χωράει' συνολικά 20 πακέτα (μαζί με τις κεφαλίδες) και έχει καθυστέρηση διάδοσης (Propagation Delay) 10 ms. Δύο χρήστες (Α και Β) θέλουν να χρησιμοποιήσουν το κανάλι και να στείλουν ο καθένας από 30 KB, στέλνοντας δεδομένα του 1 KB ξεκινώντας από 1 KB και με ρυθμό που αυξάνεται εκθετικά κάθε φορά που παίρνουν επιβεβαίωση λήψης (2 KB, 4 KB, 8 KB, κ.λπ.). Θεωρήστε ότι όταν υπερβούμε τη χωρητικότητα του δικτύου τα πακέτα χάνονται. Το δικτυακό πρωτόκολλο προσθέτει 40 Byte κεφαλίδα για κάθε KB δεδομένων. Όταν ο αποστολέας Α επιχειρήσει να στείλει δεδομένα για δεύτερη φορά (2 KB), τότε ο αποστολέας Β ξεκινάει την αποστολή δεδομένων (προφανώς με ρυθμό 1 KB) και στέλνει τα πακέτα του με διαφορά λίγων ms μετά τον πρώτο.

- α) Περιγράψτε τη διαδικασία αποστολής δεδομένων από τους δύο χρήστες.
- β) Πόσα bit πρέπει να χρησιμοποιηθούν για την αρίθμηση των πακέτων (sequence number) σε αυτή την περίπτωση (για κάθε ροή), και πόσα για να αντιμετωπιστεί οποιαδήποτε κατάσταση (λιγότερες ή περισσότερες ροές) σε αυτό το κανάλι;

Εργαστηριακά Μαθήματα στα Δίκτυα και Διαδίκτυα Υπολογιστών

Το βιβλίο αυτό αποτελεί μια προσπάθεια μεθοδευμένης παρουσίασης Εργαστηριακών Μαθημάτων στα Δίκτυα και Διαδίκτυα Υπολογιστών. Το αντικείμενο είναι εξαιρετικά ευρύ – περιλαμβάνει στοιχεία από ένα τοπικό δίκτυο μέχρι οποιοδήποτε γεωγραφικά εκτεταμένο διαδίκτυο και, επομένως, καλύπτει μεγάλο πλήθος αλγορίθμων και πρωτοκόλλων τα οποία είναι δύσκολο να εξεταστούν αναλυτικά σε όλες τους τις πτυχές. Με βάση αυτή την παραδοχή, στόχος του βιβλίου είναι να καλυφθεί η ύλη που θα μπορούσε να διδαχθεί στο διάστημα ενός ακαδημαϊκού έτους. Επιπλέον, η ύλη είναι ταξινομημένη έτσι ώστε να μπορεί να διδαχθεί σε δύο εξάμηνα και αναφέρεται σε δίκτυα και σε διαδίκτυα, αντίστοιχα.

Τα εργαλεία που χρησιμοποιούνται συστηματικά σε αυτό το βιβλίο είναι ανοιχτού κώδικα. Η χρήση του δικτυακού προσομοιωτή ns-2 είναι εκτεταμένη για δύο επιπλέον λόγους: ο ns-2, ο οποίος ξεκίνησε από το Πανεπιστήμιο του Berkeley αλλά αναπτύχθηκε χάρη στη συνεισφορά των ερευνητών παγκοσμίως, είναι ο πιο αξιόπιστος, ο πιο δοκιμασμένος, και ο πιο πλήρης προσομοιωτής, με δεδομένο ότι έχει ενσωματώσει υλοποιήσεις σεναρίων και πρωτοκόλλων που αναφέρονται σε πρότυπα αλλά και σε πειραματικούς σχεδιασμούς.

Οι συγγραφείς

Η συγγραφική ομάδα αποτελείται από μέλη της εκπαιδευτικής και ερευνητικής κοινότητας αλλά και της βιομηχανίας τηλεπικοινωνιών.

Ο Βασίλης Τσαουσιδής είναι Καθηγητής του τμήματος Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Δημοκρίτειου Πανεπιστημίου Θράκης, ερευνητικός συνεργάτης στο Πανεπιστήμιο MIT, και Διευθυντής του Εργαστηρίου Διαδικτυωμένων Συστημάτων (<http://intersys.ee.duth.gr/>).

Ο Δρ Λευτέρης Μαμάτας είναι ερευνητής στο UCL και ασχολείται με τον έλεγχο συμφόρησης.

Ο Δρ Ιωάννης Ψαρράς είναι ερευνητής στο Πανεπιστήμιο του Surrey και ασχολείται με τη συμπεριφορά των πρωτοκόλλων σε ενσύρματα και ασύρματα δίκτυα.

Ο Ευστράτιος Κοσμιδής είναι απόφοιτος του τμήματος Μηχανικών Υπολογιστών και Πληροφορικής του Πανεπιστημίου Πατρών και Προϊστάμενος Τεχνικού Τμήματος του ΟΤΕ.

Ο Στυλιανός Δημητρίου είναι διδάκτορας του τμήματος Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Δημοκρίτειου Πανεπιστημίου Θράκης με εξειδίκευση στις τεχνικές διαχείρισης ουρών.



εκδόσεις
ΚΛΕΙΔΑΡΙΘΜΟΣ

Διοικητικό 4, Σταθμός Λαρίσης, 10440 ΑΘΗΝΑ, Τηλ. 210-5237635
www.klidarithmos.gr info@klidarithmos.gr

ISBN 978-960-461-386-1

