Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών
και Μηχανικών Υπολογιστών
Τομέας Συστημάτων Μετάδοσης Πληροφορίας
και Τεχνολογίας Υλικών

Ανάλυση Επίδοσης των Πρωτοκόλλων QUIC και TCP σε Δίκτυο LTE

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Απόστολος Κυρατζής

Επιβλέπων : Παναγιώτης Κωττής

Καθηγητής Ε.Μ.Π

Αθήνα, Σεπτέμβριος 2020

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών
και Μηχανικών Υπολογιστών
Τομέας Συστημάτων Μετάδοσης Πληροφορίας
και Τεχνολογίας Υλικών

Ανάλυση Επίδοσης των Πρωτοκόλλων QUIC και TCP σε Δίκτυο LTE

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Απόστολος Κυρατζής

Επιβλέπων : Παναγιώτης Κωττής

Καθηγητής Ε.Μ.Π

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 8η Σεπτεμβρίου 2020

.....	.....	.....
Παναγιώτης Κωττής	Χρήστος Καψάλης	Γεώργιος Φικιώρης

Αθήνα, Σεπτέμβριος 2020

.....

Απόστολος Κυρατζής

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Περίληψη

Η ανάπτυξη των ασύρματων δικτύων επικοινωνιών και η εξάπλωση της χρήσης των έξυπνων κινητών συσκευών επέφερε τα τελευταία χρόνια ραγδαία αύξηση στην κίνηση από Web Browsing και Video Streaming εφαρμογές. Η τάση αυτή αναμένεται να ενταθεί περισσότερο, καθώς βρισκόμαστε στην αυγή νέων τεχνολογιών όπως το Διαδίκτυο των Πραμάτων (Internet of Things) και τα ασύρματα δίκτυα 5ης γενιάς (5G). Ωστόσο, η στοίβα πρωτοκόλλων στην οποία βασίζεται η λειτουργία του Διαδικτύου και συγκεκριμένα το Πρωτόκολλο Ελέγχου Μετάδοσης (Transmission Control Protocol - TCP), παρά την ανάπτυξη ορισμένων βελτιώσεων, δεν έχει αλλάξει στον πυρήνα της από όταν σχεδιάστηκε για τα πρώτα ενσύρματα δίκτυα. Το TCP παρουσιάζει σήμερα ορισμένα γνωστά προβλήματα όπως την έλλειψη πολυπλεξίας στο επίπεδο μεταφοράς, προκαλώντας το επονομαζόμενο Head of Line Blocking(HOL Blocking). Επομένως, το ενδιαφέρον στράφηκε στην ανάπτυξη ενός νέου πρωτοκόλλου μεταφοράς. Το 2013 προτάθηκε από τη Google το πρωτόκολλο QUIC (Quick UDP Internet Connections), το οποίο λειτουργεί πάνω από το υπάρχον πρωτόκολλο UDP (User Datagram Protocol). Το QUIC βασίζεται στις αρχές λειτουργίας του TCP σχετικά με ασφαλή και αξιόπιστη μεταφορά δεδομένων, αλλά παρουσιάζει κα ουσιώδεις διαφορές.

Στην παρούσα διπλωματική εργασία μελετάται η επίδοση του πρωτοκόλλου QUIC συγκριτικά με το TCP σε ασύρματα δίκτυα 4ης γενιάς (Long Term Evolution - LTE). Στόχος είναι να εξεταστούν διάφορα σενάρια χρήσης, προκειμένου να διερευνηθούν τα αναμενόμενα οφέλη του QUIC στα σύγχρονα ασύρματα δίκτυα.

Στο πρώτο κεφάλαιο γίνεται μία εισαγωγή στη στοίβα πρωτοκόλλων TCP/IP και παρουσιάζονται στατιστικά στοιχεία για το επίπεδο της δικτυακής κίνησης στο Internet.

Στο δεύτερο κεφάλαιο παρουσιάζεται το πρωτόκολλο TCP. Περιγράφονται οι βασικές υπηρεσίες που προσφέρει το TCP, η δομή της επικεφαλίδας και οι μηχανισμοί λειτουργίας του.

Στο τρίτο κεφάλαιο γίνεται εκτενής αναφορά στο πρωτόκολλο QUIC, με έμφαση στις διαφορές με το TCP. Αναλύονται οι νέοι τύποι ροών (streams), πλαισίων και πακέτων, καθώς και οι μηχανισμοί για τον έλεγχο ροής και συμφόρησης.

Στο τέταρτο κεφάλαιο παρουσιάζονται τα συστήματα LTE. Αρχικά περιγράφεται το δίκτυο πρόσβασης και κορμού και στη συνέχεια η στοίβα πρωτοκόλλων και οι λειτουργίες του φυσικού στρώματος.

Στο πέμπτο κεφάλαιο παρουσιάζονται οι υπάρχουσες τεχνικές Mobile Acceleration, δηλαδή της βελτιστοποίησης των TCP συνδέσεων ενός ασύρματου δικτύου, και η πιθανότητα εφαρμογής τους στο QUIC.

Στο έκτο κεφάλαιο γίνεται η μελέτη της επίδοσης των πρωτοκόλλων TCP και QUIC σε LTE δίκτυο με χρήση προσομοιώσεων για διάφορα σενάρια εφαρμογών.

Λέξεις Κλειδιά: Πρωτόκολλο Μεταφοράς, TCP, QUIC, LTE, Mobile Acceleration, TCP Optimization, προσομοίωση, ns-3

Abstract

The deployment of wireless networks and the widespread use of smart mobile devices have led to a rapid increase in data traffic from Web Browsing and Video Streaming applications in the recent years. This trend is expected to magnify with the emergence of new technologies, such as the Internet of Things (IoT) and the 5th generation of wireless networks (5G). However, the protocol stack on which the Internet's functionality is based and specifically the Transmission Control Protocol (TCP), despite the development of some improvements, has not changed in its core from when it was originally designed for wired networks. Today, TCP experiences certain well know issues, such as the lack of multiplexing capability in the transport layer, which causes the problem identified as Head of Line Blocking (HOL Blocking). Therefore, scientific interest was focused on the design of a new transport protocol. For this reason, the QUIC (Quick UDP Internet Connections) protocol was proposed by Google in 2013, which is implemented on top of the existing UDP (User Datagram Protocol). QUIC's design builds on that of TCP for secure and reliable data transmission, but the two also present significant differences.

This diploma thesis evaluates the performance of QUIC compared to TCP when operating over 4G networks (Long Term Evolution - LTE). The main goal is to examine different use cases and scenarios, in order to investigate the possible performance improvements of QUIC for modern wireless networks.

The first chapter presents an overview of today's Internet traffic in numbers and gives a brief introduction to the TCP/IP protocol stack.

The second chapter presents the TCP protocol. The core functionalities that TCP offers are described, along with header formats and operational mechanisms.

The third chapter is an extensive reference to QUIC protocol, with emphasis on the differences with TCP. The new transport streams, frame and packet formats, as well as flow and congestion control mechanisms are analyzed.

The fourth chapter describes the LTE systems. Firstly, the Radio Access and Core Network architecture is presented and afterwards the design of the protocol stack and the operation of the physical layer.

The fifth chapter focuses on the existing techniques for Mobile Acceleration, which is the optimization of TCP connections inside a wireless network and its possible applications on future QUIC connections.

The sixth chapter studies the performance of QUIC and TCP over an LTE network, based on simulations of different use case scenarios.

Keywords: Transport Protocol, TCP, QUIC, LTE, Mobile Acceleration, TCP Optimization, simulation, ns-3

Ευχαριστίες

Θα ήθελα να ευχαριστήσω θερμά τον επιβλέποντα καθηγητή μου, κ. Παναγιώτη Κωττή, για την εξαιρετική συνεργασία μας κατά τη διάρκεια της εκπόνησης της παρούσας διπλωματικής εργασίας. Η βοήθειά του υπήρξε πολύτιμη για την οργάνωση και σύνταξη αυτού του τόμου. Θα ήθελα, επίσης, να τον ευχαριστήσω τόσο για τις επιστημονικές γνώσεις που μου παρείχε μέσω της διδασκαλίας του, όσο και για τον επαγγελματισμό και ήθος που μου μετέδωσε.

Καθώς ολοκληρώνεται ο κύκλος των προπτυχιακών σπουδών μου, θα ήθελα να ευχαριστήσω όλους τους φίλους που είχα την τύχη και τιμή να γνωρίσω όλα αυτά τα χρόνια. Η συντροφιά και οι ξεχωριστές προσωπικότητές τους έκαναν πάντα πιο εύκολες και ευχάριστες τις απαιτήσεις των σπουδών.

Τέλος, η συγκεκριμένη διπλωματική εργασία αφιερώνεται στους γονείς μου, Ιωάννη και Άννα, όπως επίσης και στην αγαπημένη μου αδερφή Κατερίνα. Η απλόχερη αγάπη και αδιάκοπη στήριξή τους μου δίνει κίνητρο και θέληση ώστε να βελτιώνομαι συνεχώς ως άνθρωπος. Από τα βάθη της καρδιάς μου, τους ευχαριστώ.

Απόστολος Κυρατζής
Αθήνα, 8η Σεπτεμβρίου 2020

If we lose our Money, it gives us some Concern.
If we are cheated or robb'd of it, we are angry:
But Money lost may be found;
What we are robb'd of may be restored:
The Treasure of Time once lost, can never be recovered;
Yet we squander it as tho' 'twere nothing worth,
Or we had no Use for it.
— Benjamin Franklin

Περιεχόμενα

Κατάλογος Σχημάτων **15**
Κατάλογος Πινάκων **19**
Κατάλογος Συντμήσεων **20**
Γλωσσάριο Απόδοσης Αγγλικών Όρων **24**
Κεφάλαιο 1. Εισαγωγή **25**
1.1 Το Σημερινό Διαδίκτυο σε Αριθμούς 25
1.2 Το Διαδίκτυο ως Οντότητα 25
1.2.1 Το Μοντέλο Παροχής Υπηρεσίας 26
1.2.2 Το Μοντέλο Δικτυακών Υποδομών 26
1.3 Η Στοίβα Πρωτοκόλλων TCP/IP 28
1.3.1 Επίπεδο Εφαρμογής 29
1.3.2 Επίπεδο Μεταφοράς 31
1.3.3 Επίπεδο Δικτύου 32
1.3.4 Επίπεδο Ζεύξης 32
1.3.5 Φυσικό Επίπεδο 33
Κεφάλαιο 2. Transmission Control Protocol (TCP) **35**
2.1 Το Μοντέλο Υπηρεσίας του TCP 35
2.2 Επικεφαλίδα TCP 36
2.3 Η Σύνδεση TCP 38
2.3.1 Εγκατάσταση Σύνδεσης - Η Τριμερής Χειραψία 39
2.3.2 Τερματισμός Σύνδεσης 40
2.4 Αξιόπιστη Μεταφορά Δεδομένων 41
2.4.1 Αριθμός Ακολουθίας 42
2.4.2 Αριθμός Επιβεβαίωσης 42
2.4.3 Ανίχνευση Απωλειών 43
2.5 Έλεγχος Ροής 47
2.6 Έλεγχος Συμφόρησης 49
2.6.1 Παράθυρο Συμφόρησης 49
2.6.2 Βασικές Αρχές 50
2.6.3 Ο Αλγόριθμος Ελέγχου Συμφόρησης 50
2.6.4 Δικαιοσύνη 54
2.6.5 Δημοφιλείς Αλγόριθμοι Συμφόρησης 55

2.7 Ασφάλεια Συνδέσεων TCP 57
2.7.1 Κρυπτογραφία Συμμετρικού Κλειδιού 58
2.7.2 Κρυπτογραφία Δημόσιου Κλειδιού 59
2.7.3 Η χειραψία TLS 60
2.8 Σημερινά Προβλήματα του TCP 61
Κεφάλαιο 3. Quick UDP Internet Connections (QUIC) 63
3.1 Η Στοίβα Πρωτοκόλλων του QUIC 63
3.2 Επισκόπηση Νέων Χαρακτηριστικών 65
3.3 Πακέτα και Επικεφαλίδες QUIC 66
3.3.1 Μέγεθος Πακέτων QUIC 66
3.3.2 Long Packet Header 66
3.3.3 Short Packet Header 69
3.4 Πλαίσια QUIC 69
3.4.1 Πλαίσιο CRYPTO 70
3.4.2 Πλαίσιο STREAM 71
3.4.3 Πλαίσιο ACK 73
3.4.4 Πλαίσιο MAX_DATA 75
3.4.5 Πλαίσιο CONNECTION_CLOSE 75
3.5 Αξιόπιστη Μεταφορά Δεδομένων QUIC 76
3.5.1 Διαφορές μεταξύ QUIC και TCP 76
3.5.2 Εκτίμηση Χρόνου RTT 77
3.5.3 Ανίχνευση Απωλειών 78
3.6 Έλεγχος Συμφόρησης 80
3.7 Ασφάλεια Συνδέσεων QUIC 81
3.7.1 Εγκατάσταση Σύνδεσης QUIC 82
3.7.2 Ασφάλεια Πακέτων και Επικεφαλίδων QUIC 83
Κεφάλαιο 4. Long Term Evolution (LTE) 85
4.1 Αρχιτεκτονική Δικτύου LTE 86
4.1.1 Δίκτυο Ραδιοπρόσβασης (E-UTRAN) 86
4.1.2 Δίκτυο κορμού (EPC) 88
4.2 Στοίβα Πρωτοκόλλων LTE 90
4.2.1 Non-Access Stratum - NAS 91
4.2.2 Radio Resource Control - RRC 91
4.2.3 Packet Data Convergence Protocol - PDCP 92

4.2.4 Radio Link Control - RLC 92
4.2.5 Medium Access Control - MAC 92
4.2.6 Physical Layer - PHY 92
4.2.7 Ροή Δεδομένων στη Στοίβα Πρωτοκόλλων 93
4.3 Φυσικό Στρώμα LTE 94
4.3.1 Τεχνικές Πολλαπλής Πρόσβασης OFDMA 94
4.3.2 Δομή Πλαισίου FDD 97
4.3.3 Σχήμα μετάδοσης DL 99
4.3.4 Σχήμα μετάδοσης UL 102
4.4 Κανάλια LTE 103
4.4.1 Λογικά Κανάλια 104
4.4.2 Κανάλια Μεταφοράς 104
4.4.3 Φυσικά Κανάλια Δεδομένων 105
4.4.4 Φυσικά Κανάλια Ελέγχου 106
4.5 Φυσικά Σήματα LTE 107
Κεφάλαιο 5. Mobile Optimization 111
5.1 TCP Acceleration 111
5.1.1 Επιτάχυνση της Φάσης Αργής Εκκίνησης 112
5.1.2 Αποδοτικότερη Διαχείριση Απωλειών Πακέτων 115
5.1.3 Packet Pacing 116
5.2 Αξιοποίηση Πληροφορίας του Δικτύου RAN 119
5.3 QUIC Spin Bit Measurements 120
Κεφάλαιο 6. Επίδοση των Πρωτοκόλλων TCP και QUIC σε δίκτυο LTE 123
6.1 Σενάριο Προσομοίωσης 123
6.2 Ρυθμαπόδοση Σταθερής Κατάστασης 125
6.2.1 UE Distance = 250m 125
6.2.2 UE Distance = 750m 127
6.2.3 UE Distance = 1500m 128
6.2.4 UE Distance = 2500m 130
6.2.5 Συγκριτικά Αποτελέσματα 131
6.3 Δικαιοσύνη Ροών TCP και QUIC 132
6.3.1 2 QUIC Flows 133
6.3.2 1 QUIC vs Multiple TCP Flows 133
6.3.3 2 QUIC vs 2 TCP Flows 133

6.4 Επίδοση σε File Download 135
6.5 Επίδραση των QUIC Streams στη Ρυθμαπόδοση 137
6.6 Συμπεράσματα και Μελλοντικές Επεκτάσεις 141
Βιβλιογραφία 143

Κατάλογος Σχημάτων

Σχήμα 1.1 Αναμενόμενη αύξηση διασυνδεδεμένων συσκευών και συνδέσεων στο Διαδίκτυο 25
Σχήμα 1.2 Συστατικά του Διαδικτύου για διάφορους τύπους ζεύξεων και δικτύων 27
Σχήμα 1.3 Οργάνωση σε βαθμίδες και διασύνδεση των ISP 27
Σχήμα 1.4 Η στοίβα πρωτοκόλλων TCP/IP σε πέντε επίπεδα διαστρωμάτωσης 28
Σχήμα 1.5 Η διαδικασία της ενθυλάκωσης με χρήση επικεφαλίδων 29
Σχήμα 1.6 Μεταφορά HTTP/2 αιτημάτων και απαντήσεων σε αρχιτεκτονική client-server 30
Σχήμα 1.7 Το τμήμα επιπέδου μεταφοράς του πρωτοκόλλου UDP 31
Σχήμα 1.8 Εννοιολογική ροή δεδομένων της στοίβας TCP/IP στο Διαδίκτυο 33
Σχήμα 2.1 Η δομή του τμήματος TCP 37
Σχήμα 2.2 Η ανταλλαγή τμημάτων κατά την τριμερή χειραψία του TCP 39
Σχήμα 2.3 Τερματισμός σύνδεσης TCP 40
Σχήμα 2.4 Οι δύο βασικές προσεγγίσεις για την υλοποίηση αξιόπιστης μεταφοράς 41
Σχήμα 2.5 Τεμαχισμός αρχείου σε τμήματα TCP και ανάθεση αριθμών ακολουθίας 42
Σχήμα 2.6 Εκτίμηση του χρόνου μετ' επιστροφής RTT με δείγματα SampleRTT 44
Σχήμα 2.7 Λήψη τριών διπλότυπων ACK και ταχεία επαναμετάδοση 46
Σχήμα 2.8 Το παράθυρο λήψης (rwnd) ενός παραλήπτη TCP 47
Σχήμα 2.9 Η αργή εκκίνηση του TCP 51
Σχήμα 2.10 Η αποφυγή συμφόρησης του TCP 52
Σχήμα 2.11 Διαφορές στην εξέλιξη του cwnd μεταξύ TCP Tahoe και TCP Reno 53
Σχήμα 2.12 Διάγραμμα καταστάσεων του αλγορίθμου ελέγχου συμφόρησης TCP 54
Σχήμα 2.13 Προσθετική αύξηση-Πολλαπλασιαστική μείωση του ελέγχου συμφόρησης TCP 55
Σχήμα 2.14 Η στοίβα πρωτοκόλλων με ενσωμάτωση του TLS 58
Σχήμα 2.15 Κρυπτογραφία Συμμετρικού Κλειδιού 59
Σχήμα 2.16 Κρυπτογραφία Δημόσιου Κλειδιού 59
Σχήμα 2.17 Η χειραψία TLS με χρήση του αλγορίθμου RSA 60
Σχήμα 2.18 Head-of-Line (HOL) Blocking in TCP 62
Σχήμα 3.1 Η στοίβα πρωτοκόλλων του QUIC 64
Σχήμα 3.2 Η δομή των πακέτων QUIC 67
Σχήμα 3.3 Το payload ενός πακέτου QUIC 69
Σχήμα 3.4 Η δομή του πλαισίου CRYPTO 70

Σχήμα 3.5 Η δομή του πλαισίου STREAM....72
Σχήμα 3.6 Η δομή του πλαισίου ACK....73
Σχήμα 3.7 Η μορφή του πεδίου ACK Ranges ενός πλαισίου ACK....74
Σχήμα 3.8 Η δομή του πλαισίου MAX_DATA....75
Σχήμα 3.9 Η δομή του πλαισίου CONNECTION_CLOSE....76
Σχήμα 3.10 Περίπτωση επίμονης συμφόρησης στο QUIC....81
Σχήμα 3.11 Η χειραψία QUIC....82
Σχήμα 3.12 Χρόνος εγκατάστασης σύνδεσης QUIC vs (TCP+TLS)....83
Σχήμα 3.13 Ο μηχανισμός προστασίας των πακέτων QUIC....84
Σχήμα 4.1 Η εξέλιξη των εκδόσεων της 3GPP για τα συστήματα LTE....85
Σχήμα 4.2 Αρχιτεκτονική δικτύου LTE....86
Σχήμα 4.3 Αρχιτεκτονική E-UTRAN....87
Σχήμα 4.4 Αρχιτεκτονική EPC....88
Σχήμα 4.5 Σύνδεση του E-UTRAN με τις μονάδες MME και S-GW....89
Σχήμα 4.6 Λειτουργικός διαχωρισμός μεταξύ E-UTRAN και EPC του δικτύου LTE....90
Σχήμα 4.7 Στοίβα πρωτοκόλλων για το U-Plane....91
Σχήμα 4.8 Στοίβα πρωτοκόλλων για το C-Plane....91
Σχήμα 4.9 Λογικό διάγραμμα ροής δεδομένων στη στοίβα πρωτοκόλλων του LTE....93
Σχήμα 4.10 Δυνατοί τρόποι λειτουργίας: LTE - FDD και LTE - TDD....94
Σχήμα 4.11 Τα υποφέροντα που χρησιμοποιεί η OFDM στο πεδίο της συχνότητας....95
Σχήμα 4.12 Αναπαράσταση συμβόλου OFDM στο πεδίο του χρόνου....95
Σχήμα 4.13 Εισαγωγή Cyclic Prefix σε ένα OFDM σύμβολο....96
Σχήμα 4.14 Κατανομή εύρους ζώνης στις τεχνικές OFDMA και SC-FDMA....97
Σχήμα 4.15 Δομή πλαισίου FDD με χρήση κανονικού cyclic prefix....98
Σχήμα 4.16 Δομή OFDM συμβόλου....98
Σχήμα 4.17 LTE DL Resource Grid....99
Σχήμα 4.18 Δομικό διάγραμμα υλοποίησης OFDMA στο LTE....100
Σχήμα 4.19 Δομικό διάγραμμα υλοποίησης SC-FDMA στο LTE....102
Σχήμα 4.20 Η απεικόνιση μεταξύ των διαφορετικών κατηγοριών καναλιών του LTE 103
Σχήμα 4.21 Η δομή του φυσικού καναλιού ελέγχου PUCCH....107
Σχήμα 4.22 Φυσικά σήματα DRS και SRS της ζεύξης UL....108
Σχήμα 4.23 Σήματα συγχρονισμού PSS και SSS στη διάρκεια ενός πλαισίου LTE....109
Σχήμα 4.24 Σήματα αναφοράς (RS) σε ένα RB στην κατεύθυνση DL....109
Σχήμα 5.1 Τοποθέτηση TCP Accelerator εντός του δικτύου του παρόχου....113

Σχήμα 5.2 Επιτάχυνση της φάσης της αργής εκκίνησης του TCP 113
Σχήμα 5.3 Σύγκριση της φάσης αργής εκκίνησης με και χωρίς χρήση TCP Accelerator 114
Σχήμα 5.4 Συμβολή TCP Accelerator στην αύξηση της ρυθμαπόδοσης στα ασύρματα δίκτυα 115
Σχήμα 5.5 Διαχείριση απωλειών πακέτων από τον TCP Accelerator 116
Σχήμα 5.6 Βελτίωση του throughput που επιτυγχάνει η τεχνική TCP Acceleration 117
Σχήμα 5.7 Μέσοι χρόνοι RTT με ενεργοποιημένο (μπλε) και απενεργοποιημένο (κόκκινο) TCP Accelerator 118
Σχήμα 5.8 Ποσοστό αναμεταδόσεων με ενεργοποιημένο (μπλε) και απενεργοποιημένο (κόκκινο) TCP Accelerator 118
Σχήμα 5.9 Μέτρηση του χρόνου RTT από έναν ενδιάμεσο παρατηρητή 120
Σχήμα 5.10 Η τετραγωνική κυματομορφή που παράγει ο μηχανισμός του Spin Bit 121
Σχήμα 6.1 Μοντέλο προσομοίωσης για τη μελέτη των TCP και QUIC σε δίκτυο LTE 123
Σχήμα 6.2 Απώλεια σε dB που θα υποστεί, λόγω διαλείψεων, η μετάδοση των physical frames του LTE κατά τη διάρκεια της προσομοίωσης 124
Σχήμα 6.3 Ρυθμαπόδοση των πρωτοκόλλων TCP και QUIC (UE distance = 250m) 125
Σχήμα 6.4 Μεταβολή του παραθύρου συμφόρησης (cwnd) για τα πρωτόκολλα TCP και QUIC (UE distance =250m) 126
Σχήμα 6.5 Εκτίμηση χρόνου RTT για τα πρωτόκολλα TCP και QUIC (UE distance = 250m) 126
Σχήμα 6.6 Ρυθμαπόδοση των πρωτοκόλλων TCP και QUIC (UE distance = 750m) 127
Σχήμα 6.7 Μεταβολή του παραθύρου συμφόρησης (cwnd) για τα πρωτόκολλα TCP και QUIC (UE distance = 750m) 127
Σχήμα 6.8 Εκτίμηση χρόνου RTT για τα πρωτόκολλα TCP και QUIC (UE distance = 750m) 128
Σχήμα 6.9 Ρυθμαπόδοση των πρωτοκόλλων TCP και QUIC (UE distance = 1500m) 128
Σχήμα 6.10 Μεταβολή του παραθύρου συμφόρησης (cwnd) για τα πρωτόκολλα TCP και QUIC (UE distance = 1500m) 129
Σχήμα 6.11 Εκτίμηση χρόνου RTT για τα πρωτόκολλα TCP και QUIC (UE distance = 1500m) 129
Σχήμα 6.12 Ρυθμαπόδοση των πρωτοκόλλων TCP και QUIC (UE distance = 2500m) 130
Σχήμα 6.13 Μεταβολή του παραθύρου συμφόρησης (cwnd) για τα πρωτόκολλα TCP και QUIC (UE distance = 2500m) 130
Σχήμα 6.14 Εκτίμηση χρόνου RTT για τα πρωτόκολλα TCP και QUIC (UE distance = 2500m) 131
Σχήμα 6.15 Μέση ρυθμαπόδοση σταθερής κατάστασης για τα πρωτόκολλα TCP και QUIC συναρτήσει της ποιότητας του σήματος λήψης 132

Σχήμα 6.16 Ρυθμαπόδοση δύο ανταγωνιστικών ροών QUIC 133
Σχήμα 6.17 Ρυθμαπόδοση ανταγωνιστικών ροών: 1 QUIC vs 2 TCP 134
Σχήμα 6.18 Ρυθμαπόδοση ανταγωνιστικών ροών: 1 QUIC vs 5 TCP 134
Σχήμα 6.19 Ρυθμαπόδοση ανταγωνιστικών ροών: 2 QUIC vs 2 TCP 135
Σχήμα 6.20 Χρόνοι λήψης αρχείων για τα πρωτόκολλα TCP και QUIC συναρτήσει του μεγέθους αρχείου 136
Σχήμα 6.21 Goodput λήψης αρχείων για τα πρωτόκολλα TCP και QUIC συναρτήσει του μεγέθους αρχείου 137
Σχήμα 6.22 Η χρήση των QUIC streams για την επίλυση του Head of Line Blocking 138
Σχήμα 6.23 Ρυθμαπόδοση του πρωτοκόλλου QUIC συναρτήσει του πλήθους των QUIC streams (UE distance = 250m) 139
Σχήμα 6.24 Ρυθμαπόδοση του πρωτοκόλλου QUIC συναρτήσει του πλήθους των QUIC streams (UE distance = 750m) 139
Σχήμα 6.25 Ρυθμαπόδοση του πρωτοκόλλου QUIC συναρτήσει του πλήθους των QUIC streams (UE distance = 1500m) 140
Σχήμα 6.26 Ρυθμαπόδοση του πρωτοκόλλου QUIC συναρτήσει του πλήθους των QUIC streams (UE distance = 2500m) 140

Κατάλογος Πινάκων

Πίνακας 3.1 Διαφορετικοί τύποι πακέτων QUIC για την περίπτωση των long headers.67
Πίνακας 3.2 Διαφορετικοί τύποι πλαισίων QUIC....70
Πίνακας 3.3 Σχεδιαστικές διαφορές των πρωτοκόλλων TCP και QUIC....84
Πίνακας 4.1 Παράμετροι φυσικού στρώματος LTE....101
Πίνακας 4.2 Είδη λογικών καναλιών LTE....104
Πίνακας 4.3 Είδη καναλιών μεταφοράς LTE....104
Πίνακας 4.4 Είδη φυσικών καναλιών δεδομένων LTE....105
Πίνακας 4.5 Πεδία ελέγχου φυσικού στρώματος LTE....106
Πίνακας 4.6 Είδη φυσικών καναλιών ελέγχου LTE....107
Πίνακας 4.7 Φυσικά σήματα LTE....108
Πίνακας 6.1 Παράμετροι PHY και MAC της προσομοίωσης....124
Πίνακας 6.2 Μέση ρυθμαπόδοση σταθερής κατάστασης για τα πρωτόκολλα TCP και QUIC....131
Πίνακας 6.3 Ρυθμαπόδοση ανταγωνιστικών ροών TCP και QUIC....135
Πίνακας 6.4 Χρόνοι λήψης αρχείων για τα πρωτόκολλα TCP και QUIC....136
Πίνακας 6.5 Goodput λήψης αρχείων για τα πρωτόκολλα TCP και QUIC....136
Πίνακας 6.6 Επίδραση των QUIC streams στη ρυθμαπόδοση....138

Κατάλογος Συντμήσεων

3G/4G/5G	3rd/4th/5th Generation
3GPP	Third Generation Partnership Project
ACK	Acknowledgement
ACM	Adaptive Coding and Modulation
ADC	Analog to Digital Converter
AES	Advanced Encryption Standard
AIMD	Additive Increase - Multiplicative Decrease
AM	Acknowledged Mode
API	Application Programming Interface
ARQ	Automatic Repeat Request
BCCH	Broadcast Control Channel
BCH	Broadcast Channel
BDP	Bandwidth-Delay Product
BPSK	Binary Phase Shift Keying
CCCH	Common Control Channel
CFI	Control Format Indicator
CID	Connection ID
CN	Core Network
CP	Cyclic Prefix
CQI	Channel Quality Indicator
CWND	Congestion Window
DAC	Digital to Analog Converter
DCCH	Dedicated Control Channel
DCI	Downlink Control Information
DES	Data Encryption Standard
DFT	Discrete Fourier Transform
DL	Downlink
DL-SCH	Downlink Shared Channel
DNS	Domain Name System
DRS	Demodulation Reference Signal
DTCH	Dedicated Traffic Channel
DVB	Digital Video Broadcasting
eNB	Evolved Node B
EPC	Evolved Packet Core
ETSI	European Telecommunications Standards Institute
E-UTRAN	Evolved Universal Terrestrial Radio Access Network
EWMA	Exponential Weighted Moving Average
FDD	Frequency Division Duplexing
FEC	Forward Error Correction
FFT	Fast Fourier Transform
FTP	File Transfer Protocol
GBR	Guaranteed Bit Rate
GTP	GPRS Tunneling Protocol

HARQ	Hybrid Automatic Repeat Request
HI	Hybrid ARQ Indicator
HOL	Head of Line
HSS	Home Subscriber Server
HTTP	Hyper Text Transfer Protocol
ICI	Inter Carrier Interference
IDFT	Inverse Discrete Fourier Transform
IETF	Internet Engineering Task Force
IFFT	Inverse Fast Fourier Transform
IoT	Internet of Things
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
ISI	Inter Symbol Interference
ISP	Internet Service Provider
IXP	Internet Exchange Point
LFN	Long Fat Networks
LTE	Long Term Evolution
LTE-A	Long Term Evolution - Advanced
M2M	Machine to Machine
MAC	Medium Access Control
MEC	Mobile Edge Computing
MIB	Master Information Block
MME	Mobility Management Entity
MSS	Maximum Segment Size
MTU	Maximum Transmission Unit
NAS	Non-Access Stratum
OFDM	Orthogonal Frequency Division Multiplexing
OFDMA	Orthogonal Frequency Division Multiple Access
OS	Operating System
P/S	Parallel to Serial
PAPR	Peak to Average Power Ratio
PBCH	Physical Broadcast Channel
PCCH	Paging Control Channel
PCFICH	Physical Control Format Indicator Channel
PCH	Paging Channel
PCI	Physical Cell Identity
PCRF	Policy Control and Charging Function
PDCCH	Physical Downlink Control Channel
PDCP	Packet Data Convergence Protocol
PDN	Packet Data Network
PDSCH	Physical Downlink Shared Channel
PDU	Protocol Data Unit
P-GW	Packet Gateway
PHICH	Physical Hybrid ARQ Indicator Channel
PHY	Physical Layer
PLT	Page Load Time

PRACH	Physical Random-Access Channel
PSS	Primary Synchronization Signal
PUCCH	Physical Uplink Control Channel
PUSCH	Physical Uplink Shared Channel
QAM	Quadrature Amplitude Modulation
QCI	QoS Class Identifier
QoE	Quality of Experience
QoS	Quality of Service
QPSK	Quadrature Phase Shift Keying
QUIC	Quick UDP Internet Connections
RACH	Random Access Channel
RADIUS	Remote Authentication Dial-In User Service
RAN	Radio Access Network
RB	Resource Block
RE	Resource Element
RFC	Request for Comments
RLC	Radio Link Control
RND	Research and Development
ROI	Return on Investment
RRC	Radio Resource Control
RRM	Radio Resource Management
RS	Cell-Specific Reference Signal
RTO	Retransmission Timeout
RTT	Round Trip Time
RWND	Receive Window
S/P	Serial to Parallel
S1-AP	S1 Application Protocol
SACK	Selective Acknowledgment
SC-FDMA	Single Carrier Frequency Division Multiple Access
SCTP	Stream Control Transmission Protocol
SDU	Service Data Unit
S-GW	Serving Gateway
SIB	System Information Block
SMTP	Simple Mail Transfer Protocol
SR	Scheduling Request
SRS	Sounding Reference Signal
SSS	Secondary Synchronization Signal
TAC	Tracking Area Code
TB	Transport Block
TCP	Transmission Control Protocol
TDD	Time Division Duplexing
TLS	Transport Layer Security
TM	Transparent Mode
UCI	Uplink Control Information
UDP	User Datagram Protocol
UE	User Equipment

UL	Uplink
UL-SCH	Uplink Shared Channel
UM	Unacknowledged Mode
UMTS	Universal Mobile Telecommunications System
VoIP	Voice over IP
WWW	World Wide Web

Γλωσσάριο Απόδοσης Αγγλικών Όρων

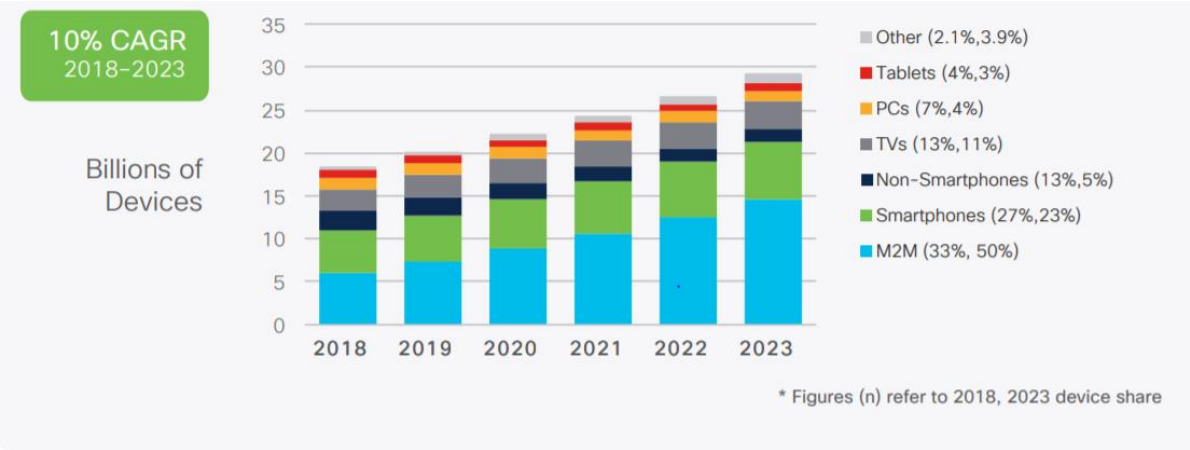
Αναζήτηση/Τηλε-ειδοποίηση	Paging
Ασυνδεσμικό Πρωτόκολλο	Connectionless Protocol
Βαθμός Χρησιμοποίησης	Utilization
Γνωστή Θύρα	Well-known Port
Διακομιστής Μεσολάβησης	Proxy Server
Διοχέτευση	Pipelining
Διπλότυπο	Duplicate
Δρομολογητής	Router
Ενδιάμεσο Κουτί	Middlebox
Ενταμιευτής	Buffer
Επίμονη Συμφόρηση	Persistent Congestion
Εύρος Ζώνης	Bandwidth
Ευρυεκπομπή	Broadcast
Ισορροπιστής Φορτίου	Load Balancer
Κρυπτοκείμενο	Ciphertext
Μοντέλο Υπηρεσίας	Service Model
Πλέγμα Ραδιοπόρων	Radio Resource Grid
Πλοήγηση Ιστού	Web Browsing
Πολύπλεξη	Multiplexing
Ραδιοκομιστής	Radio Bearer
Ραδιοπόρος	Radio Resource
Ρυθμαπόδοση	Throughput
Σημείο Πρόσβασης	Access Point
Στρώμα Ελέγχου	C-Plane
Στρώμα Χρήστη	U-Plane
Συνδεσμικό Πρωτόκολλο	Connection-oriented Protocol
Σχήμα Διαμόρφωσης	Modulation Scheme
Τείχος Προστασίας	Firewall
Τερματικό Σύστημα	End System
Τριμερής Χειραψία	Three-way Handshake
Υποφέρον	Sub-carrier
Φέρον	Carrier
Φόρτος Εργασίας	Workload
Φυλλομετρητής	Browser
Χρονοπρογραμματισμός	Scheduling
Χωρητικότητα	Capacity
Χώρος Χρήστη	User Space
Ωφέλιμο Φορτίο	Payload

Κεφάλαιο 1. Εισαγωγή

1.1 Το Σημερινό Διαδίκτυο σε Αριθμούς

Το Διαδίκτυο (Internet) αποτελεί σήμερα το ευρύτερα διαδεδομένο τεχνολογικό σύστημα, που χρησιμοποιείται καθημερινά από δισεκατομμύρια ανθρώπους, προκείμενου να έχουν πρόσβαση σε πληθώρα σύγχρονων εφαρμογών. Το 2018 υπήρχαν 3.9 δισεκατομμύρια χρήστες, ενώ προβλέπεται αύξηση του αριθμού αυτού σε 5.3 δισεκατομμύρια έως το 2023, αντιστοιχώντας στο 66% του παγκόσμιου πληθυσμού.

Ωστόσο, η μεγαλύτερη αύξηση αναμένεται στον αριθμό των διασυνδεδεμένων συσκευών στο Διαδίκτυο. Σύμφωνα με τις προβλέψεις της Cisco, ο αριθμός τους θα είναι 29.3 δισεκατομμύρια το 2023. Το μεγαλύτερος μέρος των συνδέσεων αυτών θα αφορά «επικοινωνίες μηχανής προς μηχανή» (Machine to Machine - M2M), καθώς νέες εφαρμογές του «Διαδικτύου των Πραγμάτων» (Internet of Things - IoT) θα αρχίσουν να αναπτύσσονται, όπως τα έξυπνα σπίτια και τα διασυνδεδεμένα οχήματα. Συνολικά, η κίνηση δεδομένων στο Διαδίκτυο αναμένεται να είναι 235 Exabytes ανά μήνα το 2021, τριπλάσια σε σύγκριση με την αντίστοιχη του 2016 [1].



Σχήμα 1.1 *Αναμενόμενη αύξηση διασυνδεδεμένων συσκευών και συνδέσεων στο Διαδίκτυο*
[Πηγή: Cisco Annual Internet Report, 2018-2023]

Αυτή η ανάπτυξη στη χρήση του Internet απαιτεί παράλληλα την αναβάθμιση της υπάρχουσας υποδομής των δικτύων αλλά και των πρωτοκόλλων στα οποία στηρίζεται η λειτουργία του, όπως θα φανεί αναλυτικότερα στη συνέχεια.

1.2 Το Διαδίκτυο ως Οντότητα

Η μεγάλη πολυπλοκότητα του Διαδικτύου δεν επιτρέπει τη διατύπωση συγκεκριμένης περιγραφής που να αποδίδει ακριβώς την φύση του. Σε μία πρώτη προσέγγιση, το Διαδίκτυο είναι ουσιαστικά ένα δίκτυο υπολογιστών στο οποίο συνδέονται μεταξύ τους ένας μεγάλος αριθμός από τερματικά συστήματα (end systems). Ωστόσο, το Διαδίκτυο μπορεί να κατανοηθεί καλύτερα αν εξεταστεί από δύο διαφορετικές μεταξύ τους σκοπιές, οι οποίες όμως είναι αλληλένδετες: η υποστήριξη υπηρεσιών και η παροχή δικτυακής υποδομής.

1.2.1 Το Μοντέλο Παροχής Υπηρεσίας

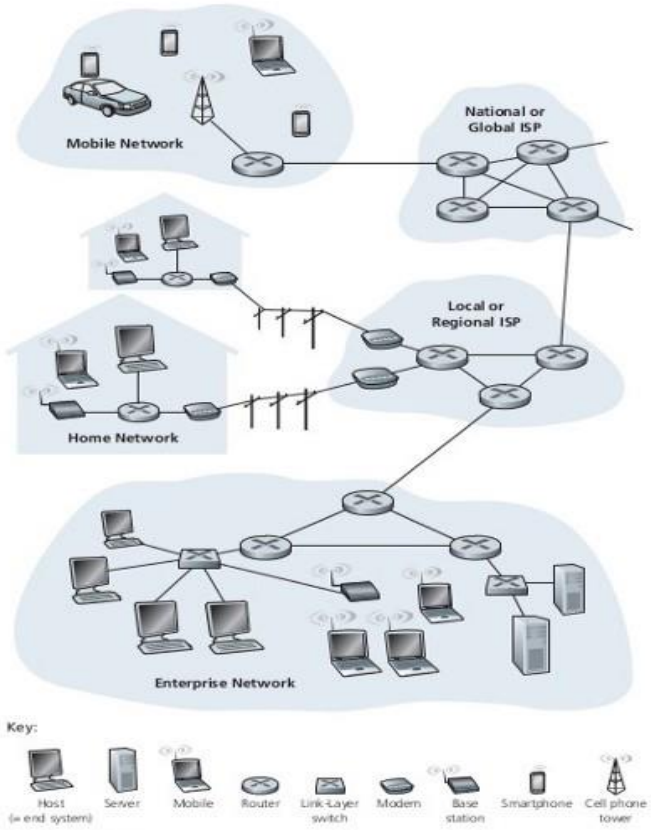
Σήμερα υπάρχει πληθώρα δικτυακών εφαρμογών οι οποίες χρησιμοποιούνται ευρέως και έχουν γίνει μέρος της καθημερινότητας. Η Πλοήγηση Ιστού (Web Browsing), τα κοινωνικά δίκτυα, το ηλεκτρονικό ταχυδρομείο και τα βίντεο συνεχούς ροής είναι μερικές από αυτές. Τέτοιες εφαρμογές εκτελούνται σε κάθε τερματική συσκευή, χρειάζεται όμως να ανταλλάσσουν δεδομένα μεταξύ τους προκειμένου να λειτουργήσουν. Αυτήν ακριβώς την υπηρεσία παρέχει το Διαδίκτυο στο λογισμικό των εφαρμογών, όταν αυτές χρειάζεται να μεταφέρουν δεδομένα από ένα τερματικό σε ένα άλλο. Η υπηρεσία αυτή καλείται από την εφαρμογή μέσω μιας Διεπαφής Προγραμματισμού Εφαρμογών (Application Programming Interface - API), η οποία ορίζει ένα σύνολο από κανόνες με τους οποίους θα ζητήσει η εφαρμογή από το Διαδίκτυο τη μεταφορά των δεδομένων της, προκειμένου αυτά να φτάσουν στην επιθυμητή εφαρμογή προορισμού. Όπως γίνεται αντιληπτό, οι κανόνες αυτοί θα είναι πολυάριθμοι και θα αφορούν διάφορες λειτουργίες. Επομένως, για να είναι αποτελεσματικοί, χρειάζονται συστηματική οργάνωση, η οποία πραγματοποιείται από την χρήση δικτυακών πρωτοκόλλων όπως θα αναλυθεί στη συνέχεια.

1.2.2 Το Μοντέλο Δικτυακών Υποδομών

Προκειμένου να είναι εφικτή η μεταφορά δεδομένων μεταξύ εφαρμογών που εκτελούνται σε διαφορετικά τερματικά, χρειάζεται κατάλληλη υποδομή μέσω της οποίας θα μεταφέρονται αυτά τα δεδομένα. Το Διαδίκτυο, επομένως, μπορεί να οριστεί ως ένα δίκτυο που αποτελείται από τερματικά συστήματα και ενδιάμεσους κόμβους, το οποίο είναι υπεύθυνο για τη φυσική μεταφορά της πληροφορίας μεταξύ των συστημάτων που συνδέονται σε αυτό. Ένα τερματικό που επιθυμεί να χρησιμοποιήσει την υπηρεσία του Διαδικτύου για μεταφορά δεδομένων πραγματοποιεί μία διαδικασία τεμαχισμού τους σε τμήματα και προσάρτησης ορισμένων bytes επικεφαλίδων (headers). Τα τελικά τμήματα που προκύπτουν είναι γενικά γνωστά ως «πακέτα» (packets) και αποτελούν τη βασική μονάδα πληροφορίας που τελικά διαδίδεται μέσα στο Διαδίκτυο προς τον τελικό προορισμό.

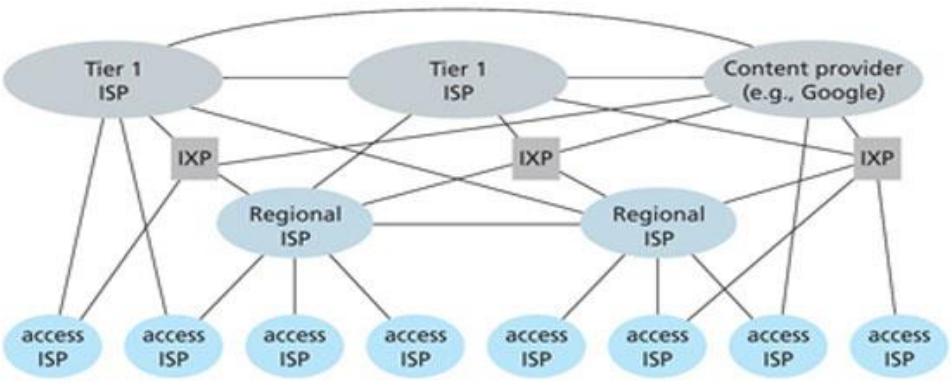
Στο Σχήμα 1.2 παρουσιάζεται ένα διάγραμμα που περιγράφει τους διάφορους τρόπους συνδεσιμότητας στο Διαδίκτυο διαφορετικού τύπου δικτύων. Αρχικά, κάθε τερματικό προκειμένου να επικοινωνήσει χρειάζεται μία ζεύξη επικοινωνίας ή σημείο πρόσβασης (access point). Σήμερα, υπάρχουν αρκετοί τύποι φυσικών ζεύξεων, όπως χάλκινα καλώδια, οπτικές ίνες και ραδιοφάσμα (radio spectrum). Το τελευταίο αφορά τα ασύρματα δίκτυα και θα αναλυθεί ιδιαίτερα στη συνέχεια. Οι δρομολογητές (routers) και οι μεταγωγείς (switches) αποτελούν τον πυρήνα του Διαδικτύου και είναι υπεύθυνοι για τη διακίνηση και προώθηση πακέτων πληροφορίας από κάποιο αποστολέα σε κάποιο παραλήπτη, με βάση τα περιεχόμενα των επικεφαλίδων των πακέτων. Κάθε τύπος δικτύου (π.χ. οικιακό, εταιρικό ή ασύρματο) πρέπει να συνδέεται με κάποιο Πάροχο Υπηρεσιών Διαδικτύου (Internet Service Provider - ISP), προκειμένου να έχει πρόσβαση στο Internet. Φυσικά, εκτός από τους τελικούς χρήστες, με τους ISP συνδέονται και όλοι οι πάροχοι διαδικτυακού περιεχομένου (π.χ. Google, YouTube, Netflix, Amazon). Οι ISP σήμερα είναι χιλιάδες, όπως για παράδειγμα εταιρικοί ISP, πανεπιστημιακοί ISP, τοπικές εταιρίες τηλεφωνίας ή μεγάλοι τηλεπικοινωνιακοί

πάροχοι. Όλοι οι ISP συνδέονται, είτε άμεσα είτε έμμεσα, μεταξύ τους, ακολουθώντας μία οργάνωση σε βαθμίδες, όπως φαίνεται στο Σχήμα 1.3.



Σχήμα 1.2 *Συστατικά του Διαδικτύου για διάφορους τύπους ζεύξεων και δικτύων* [2]

Συνοψίζοντας, το σημερινό Διαδίκτυο αποτελεί ουσιαστικά ένα «δίκτυο-δικτύων» και το μέγεθός του καθώς και η πολυπλοκότητά του αυξάνουν κατά ραγδαίο τρόπο. Προκειμένου να λειτουργήσει το Διαδίκτυο, είναι απαραίτητο κάθε υποκείμενο τμήμα του να υπακούει σε ένα κοινό σύνολο από πρωτόκολλα που έχουν σχεδιασθεί να ρυθμίζουν την αποστολή και λήψη των πληροφοριών. Το πλέον διαδεδομένο σύνολο πρωτοκόλλων σήμερα καλείται συλλογικά TCP/IP και έπαιξε καθοριστικό ρόλο στην εξέλιξη του Διαδικτύου.



Σχήμα 1.3 *Οργάνωση σε βαθμίδες και διασύνδεση των ISP* [2]

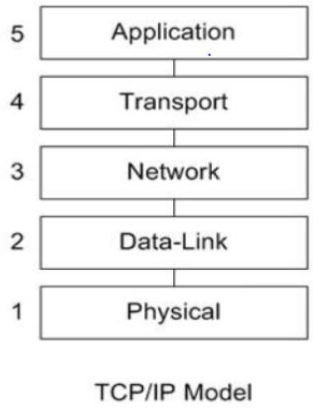
1.3 Η Στοίβα Πρωτοκόλλων TCP/IP

Έχει ήδη αναφερθεί η ανάγκη χρήσης πρωτοκόλλων. Ωστόσο, είναι χρήσιμο σε αυτό το σημείο να δοθεί σαφής ορισμός. Στη γενική μορφή του, ένα πρωτόκολλο επικοινωνίας και κατ' επέκταση ένα πρωτόκολλο δικτύωσης, ορίζει τα εξής βασικά στοιχεία:

- Τη μορφή και τη σειρά των μηνυμάτων που δύο ή περισσότερες οντότητες υποχρεούνται να ανταλλάσσουν μεταξύ τους, προκειμένου να επικοινωνήσουν.

- Όλες τις απαραίτητες ενέργειες που πρέπει να γίνουν κατά τη διάρκεια αποστολής και λήψης μηνυμάτων ή διαπίστωσης άλλου συμβάντος.

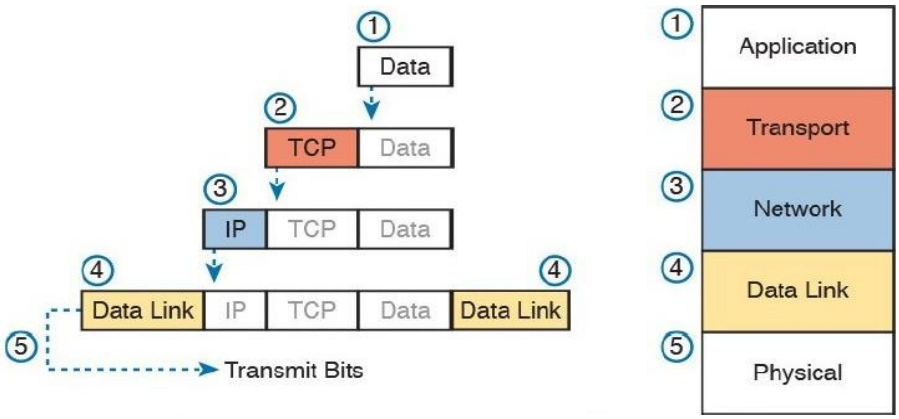
Το Internet, επομένως, έχει το δικό του σύνολο από πρωτόκολλα που καθορίζουν την επικοινωνία μεταξύ των διασυνδεδεμένων μελών του. Το όνομα Internet καθορίστηκε από τα δύο κυρίαρχα πρωτόκολλα, το TCP και το IP (Internet Protocol), ενώ ο όρος στοίβα οφείλεται στην αρχιτεκτονική υλοποίησής του. Εξαιτίας της κομβικής σημασίας τους, τα πρωτόκολλα οφείλουν να έχουν ενιαία προτυποποίηση, υπεύθυνη για την οποία είναι η οργάνωση Internet Engineering Task Force (IETF). Η IETF σήμερα παράγει την πλειοψηφία των Προτύπων Διαδικτύου (Internet Standards) μέσω των επονομαζόμενων RFC (Request for Comments). Η τέταρτη έκδοση του IP (IPv4) εκδόθηκε το 1981 με το RFC 791 και είναι η έκδοση που χρησιμοποιείται ως επί το πλείστων μέχρι σήμερα. Εξαιτίας της επικείμενης μεγάλης αύξησης των διασυνδεδεμένων συσκευών στο Internet, έχει ήδη προτυποποιηθεί το πρωτόκολλο IPv6, που σταδιακά θα αντικαταστήσει το IPv4 τα επόμενα χρόνια [3]. Η κύρια διαφορά τους είναι ότι το IPv6 χρησιμοποιεί διευθύνσεις IP μήκους 128 bits, αντί των 32 bits που χρησιμοποιεί το IPv4. Το TCP ορίστηκε επίσημα στο RFC 793 και παρά την έκδοση επόμενων RFC με βελτιώσεις της αρχικής σχεδίασης, οι βασικοί μηχανισμοί του παραμένουν οι ίδιοι.



Σχήμα 1.4 *Η στοίβα πρωτοκόλλων TCP/IP σε πέντε επίπεδα διαστρωμάτωσης* [2]

Τα πρωτόκολλα του Internet παρέχουν τους κανόνες για ένα σύνολο λειτουργιών σχετικά με αποστολή, λήψη, διευθυνσιοδότηση, δρομολόγηση και μορφοποίηση δεδομένων σε πακέτα πληροφορίας. Οι κανόνες αυτοί οργανώνονται με βάση μία αρχιτεκτονική πολλαπλών επιπέδων. Για το λόγο αυτό, καθιερώθηκε η χρήση του όρου «στοίβα», όπως φαίνεται στο Σχήμα 1.4. Κάθε επίπεδο εκτελεί ορισμένες λειτουργίες οι

οποίες συνολικά αποτελούν το μοντέλο υπηρεσίας (service model) του, η οποία προσφέρεται στο ανώτερο επίπεδο. Το πλεονέκτημα αυτής της σχεδίασης είναι ότι επιτρέπει το λογικό διαχωρισμό όλων των απαραίτητων λειτουργιών του Διαδικτύου σε επιμέρους τμήματα. Για παράδειγμα, ας υποτεθεί ότι χρειάζεται να υλοποιηθεί σε ένα επίπεδο {*n*} μία υπηρεσία αξιόπιστης παράδοσης δεδομένων μεταξύ δύο τερματικών. Ακολουθώντας την προαναφερθείσα αρχή διαστρωμάτωσης, μπορεί να χρησιμοποιηθεί κάποια αναξιόπιστη υπηρεσία παράδοσης του επιπέδου {*n-1*} και να προστεθεί στο επίπεδο {*n*} ένας μηχανισμός αναμετάδοσης, που θα εξασφαλίζει την αξιόπιστη παράδοση των δεδομένων. Με αυτό τον τρόπο, το επόμενο επίπεδο {*n+1*} έχει διαθέσιμη την αξιόπιστη παράδοση των δεδομένων του από τα δύο κατώτερα επίπεδα.



Σχήμα 1.5 *Η διαδικασία της ενθυλάκωσης με χρήση επικεφαλίδων*
[Πηγή: https://pbs.twimg.com/media/DWdg2e6WsAAMzxw.jpg]

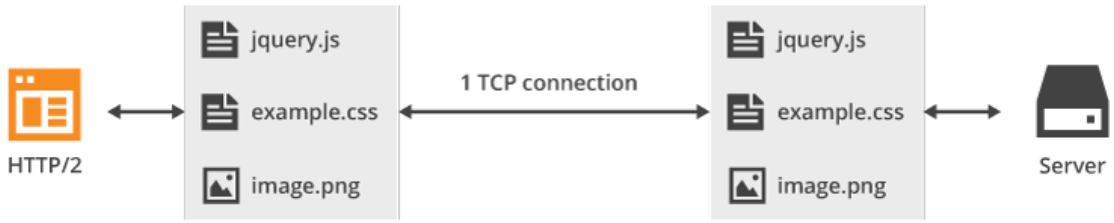
Προτού δοθεί με μία σύντομη περιγραφή των πέντε επιπέδων της στοίβας πρωτοκόλλων, είναι σημαντικό να αναφερθεί ο τρόπος με τον οποίο ένα επίπεδο προσφέρει την υπηρεσία του στην πράξη, χρησιμοποιώντας την αρχή της ενθυλάκωσης (encapsulation). Όπως φαίνεται στο Σχήμα 1.5, κάθε επίπεδο πρωτοκόλλου προσθέτει στα δεδομένα που παραλαμβάνει από το ανώτερο επίπεδο ένα πεδίο επικεφαλίδας (header field), καθώς τα δεδομένα προχωρούν προς μετάδοση στο δίκτυο. Κάθε επικεφαλίδα περιέχει συγκεκριμένα πεδία και μεταβλητές, οι οποίες αφορούν την υπηρεσία που παρέχει κάθε επίπεδο. Για παράδειγμα, το πρωτόκολλο μεταφοράς TCP προσθέτει αριθμούς ακολουθίας και επιβεβαιώσεων στα δεδομένα προκειμένου να επιτύχει την αξιόπιστη παράδοση, όπως θα αναλυθεί σε επόμενο κεφάλαιο. Όταν τα δεδομένα φθάσουν στον προορισμό τους, ακολουθείται η αντίστροφη διαδικασία αποθυλάκωσης, που κάθε επίπεδο πρωτοκόλλου αφαιρεί την αντίστοιχη επικεφαλίδα, εκτελεί ορισμένες ενέργειες με βάση την τιμή των πεδίων της και προωθεί τα δεδομένα στο αμέσως ανώτερο επίπεδο.

1.3.1 Επίπεδο Εφαρμογής

Το επίπεδο εφαρμογής διαχειρίζεται τα δεδομένα που παράγει ένας χρήστης καθώς χρησιμοποιεί μία εφαρμογή. Όλες οι σημερινές δημοφιλείς εφαρμογές Διαδικτύου τρέχουν ένα πρωτόκολλο εφαρμογής που είναι υπεύθυνο για την ανταλλαγή πακέτων πληροφορίας, που ονομάζονται «μηνύματα» (messages), μεταξύ διαδικασιών (processes) που εκτελούνται σε διαφορετικά τερματικά. Το επίπεδο εφαρμογής χρησιμοποιεί όλες τις υπηρεσίες που προσφέρουν τα κατώτερα στρώματα και ιδιαίτερα

το στρώμα μεταφοράς για αξιόπιστη παράδοση δεδομένων. Η επικοινωνία των διαδικασιών με το στρώμα μεταφοράς οργανώνεται με τη χρήση αριθμημένων θυρών (ports), που κάθε θύρα είναι συνήθως χαρακτηριστική για κάθε τύπο εφαρμογής. Στη συνέχεια αναφέρονται ορισμένα ευρέως διαδεδομένα πρωτόκολλα εφαρμογής με έμφαση στο πρωτόκολλο HTTP:

- **Hypertext Transfer Protocol (HTTP)**: Το HTTP είναι το πρωτόκολλο σύμφωνα με το οποίο λειτουργεί ο Παγκόσμιος Ιστός (World Wide Web - WWW). Ο σχεδιασμός του ακολουθεί την αρχιτεκτονική του μοντέλου πελάτη-εξυπηρετητή (client-server architecture). Όταν ένας χρήστης θέλει να συνδεθεί σε μία ιστοσελίδα, η εφαρμογή Web Browser (πελάτης) που χρησιμοποιεί αποστέλλει ένα αίτημα (request) στην εφαρμογή Web Server (εξυπηρετητής) που εκτελείται σε κάποιο άλλο τερματικό, χρησιμοποιώντας τη γνωστή θύρα 80 (well-known port). Κατόπιν, ο Web Server αποστέλλει μία απάντηση (response), που περιέχει τα δεδομένα που ζητήθηκαν. Οι σημερινές ιστοσελίδες αποτελούνται πλέον από πολλά αντικείμενα που πρέπει να σταλούν στο χρήστη, όπως για παράδειγμα αρχεία HTML, CSS και JSON. Τα αρχεία αυτά πολυπλέκονται και αποστέλλονται πάνω από την ίδια TCP (συνήθως) σύνδεση, όπως φαίνεται στο Σχήμα 1.6. Η πλέον πρόσφατη έκδοση του πρωτοκόλλου είναι η HTTP/2, ενσωματώνει τα προαναφερθέντα χαρακτηριστικά και χρησιμοποιείται σήμερα σχεδόν από το 45% του συνόλου των ιστοσελίδων [4].



Σχήμα 1.6 *Μεταφορά HTTP/2 αιτημάτων και απαντήσεων σε αρχιτεκτονική client-server*
[Πηγή: https://www.primointeractive.com/services/web-hosting]

- **Simple Mail Transfer Protocol (SMTP)**: Το SMTP είναι το πρωτόκολλο των εφαρμογών ηλεκτρονικού ταχυδρομείου (email) και αποτελεί μία από τις πλέον διαδεδομένες εφαρμογές Διαδικτύου.

- **File Transfer Protocol (FTP)**: Όπως υποδηλώνει και το όνομά του, το FTP υποστηρίζει τη μεταφορά αρχείων μεταξύ δύο τερματικών συστημάτων. Το FTP χρησιμοποιεί την θύρα 21.

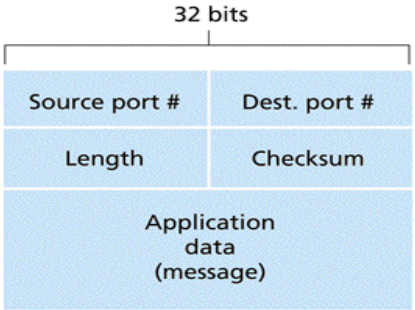
- **Domain Name System (DNS)**: Το DNS δεν σχετίζεται με κάποια γνωστή στο μέσο χρήστη εφαρμογή, όπως τα προηγούμενα πρωτόκολλα. Ωστόσο, είναι εξαιρετικά σημαντικό, καθώς μεταφράζει τα ονόματα ιστοσελίδων σε διευθύνσεις IP. Για παράδειγμα, πριν ένας χρήστης μπορέσει να δει την ιστοσελίδα της IETF στον Web Browser που χρησιμοποιεί, πρέπει το DNS να μεταφράσει το όνομα

«www.ietf.org» στη διεύθυνση IP του αντίστοιχου Web Server. Το DNS χρησιμοποιεί τη θύρα 53.

1.3.2 Επίπεδο Μεταφοράς

Το επίπεδο μεταφοράς είναι υπεύθυνο για την εγκατάσταση λογικών καναλιών επικοινωνίας που οι εφαρμογές μπορούν να χρησιμοποιήσουν για την από-άκρο-σε-άκρο ανταλλαγή των μηνυμάτων τους. Ωστόσο, η επικοινωνία σε επίπεδο μεταφοράς γίνεται μεταξύ τερματικών συστημάτων και όχι εφαρμογών. Προκειμένου τα μηνύματα από διαφορετικές εφαρμογές να μπορούν να διαχωρίζονται μεταξύ τους σε ένα τερματικό, το επίπεδο μεταφοράς χρησιμοποιεί αριθμημένες θύρες δικτύου (network ports) για κάθε εφαρμογή όπως ήδη αναφέρθηκε στην §1.3.1. Τα μηνύματα του επιπέδου εφαρμογής ενθυλακώνονται σε πακέτα πληροφορίας επιπέδου μεταφοράς που ονομάζονται «τμήματα» (segments). Οι ελάχιστες υπηρεσίες που πρέπει να παρέχει ένα πρωτόκολλο μεταφοράς είναι η παράδοση μηνυμάτων μεταξύ εφαρμογών και ο έλεγχος ακεραιότητας δεδομένων, με χρήση ειδικών πεδίων ανίχνευσης σφαλμάτων στις επικεφαλίδες. Τα δύο περισσότερο διαδεδομένα πρωτόκολλα μεταφοράς στο Διαδίκτυο είναι:

- **User Datagram Protocol (UDP)**: Το UDP αποτελεί ένα «ασυνδεσμικό» πρωτόκολλο μεταφοράς, επειδή η διαδικασία μετάδοσης δεδομένων από τον αποστολέα στον παραλήπτη ξεκινάει χωρίς την εγκατάσταση κάποιων κοινών παραμέτρων της σύνδεσης μέσω μιας «χειραψίας» των δύο τερματικών. Το UDP δεν εξασφαλίζει την αξιόπιστη παράδοση δεδομένων, προσφέρει ωστόσο ανίχνευση σφαλμάτων. Για τους λόγους αυτούς, το UDP αποκαλείται πρωτόκολλο «βέλτιστης προσπάθειας» (best effort), όρος που συναντάται και στο πρωτόκολλο IP. Ως ασυνδεσμικό πρωτόκολλο, δεν εισάγει πρόσθετες καθυστερήσεις και σε συνδυασμό με το μικρό μέγεθος επικεφαλίδας του (8 bytes), είναι χρήσιμο για πρωτόκολλα εφαρμογής που βασίζονται στη συχνή αποστολή μικρών μηνυμάτων, όπως το DNS. Τέλος, το UDP χρησιμοποιείται συχνά από εφαρμογές πραγματικού χρόνου (π.χ. τηλεδιασκέψεις) ή χρονικά ευαίσθητες (time sensitive) εφαρμογές (π.χ. VoIP), δηλαδή εφαρμογές όπου επιβάλλονται χαμηλές καθυστερήσεις μετάδοσης δεδομένων και υπάρχει ανεκτικότητα σε απώλεια πακέτων.



Σχήμα 1.7 *Το τμήμα επιπέδου μεταφοράς του πρωτοκόλλου UDP*
[Πηγή: https://networkengineering.stackexchange.com]

- **Transmission Control Protocol (TCP)**: Το TCP είναι ένα «συνδεσμικό» πρωτόκολλο, επειδή απαιτεί την αμοιβαία συμφωνία στη χρήση παραμέτρων

σύνδεσης πριν την έναρξη αποστολής δεδομένων. Παρέχει αξιόπιστη υπηρεσία παράδοσης, δηλαδή τα δεδομένα του αποστολέα φθάνουν στον παραλήπτη με τη σωστή σειρά και χωρίς λάθη. Το TCP χρησιμοποιεί αλγορίθμους προκειμένου να προσαρμόζει το ρυθμό μετάδοσης τόσο στις δυνατότητες του χρήστη για παραλαβή δεδομένων όσο και στη διαθέσιμη χωρητικότητα του υποκείμενου δικτύου κάθε χρονική στιγμή. Οι μηχανισμοί του TCP παρουσιάζονται αναλυτικά στο Κεφάλαιο 2.

1.3.3 Επίπεδο Δικτύου

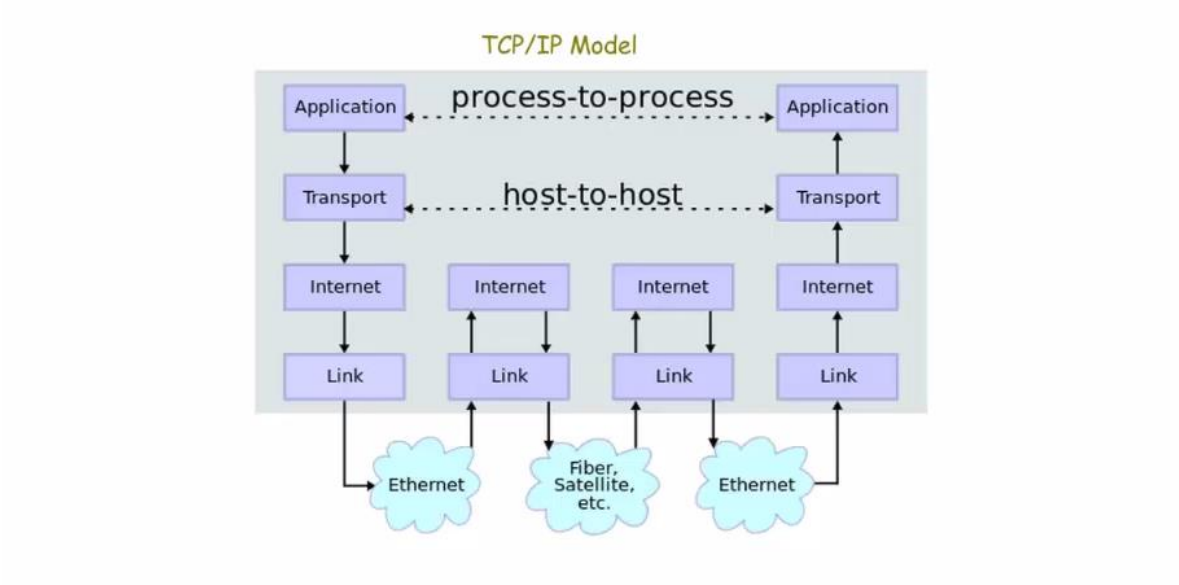
Το επίπεδο δικτύου παραλαμβάνει από το επίπεδο μεταφοράς ένα τμήμα και μία διεύθυνση προορισμού, τα συνδυάζει σε ένα «δεδομενόγραμμα» (datagram) και, στη συνέχεια, είναι υπεύθυνο για τη διακίνηση αυτού μεταξύ διαφόρων κόμβων του Διαδικτύου, μέχρις να το παραδώσει στο επίπεδο μεταφοράς του υπολογιστή προορισμού. Η μεταφορά δεδομενογραμμάτων βασίζεται σε δύο βασικά χαρακτηριστικά του επιπέδου δικτύου: την ύπαρξη συστήματος διευθυνσιοδότησης και τη δυνατότητα δρομολόγησης. Το κυρίαρχο πρωτόκολλο είναι το πρωτόκολλο IP (Internet Protocol) που ορίζει τις διευθύνσεις IP (IP addresses) και τις απαραίτητες ενέργειες που πρέπει να εφαρμόζει ένα τερματικό ή ένας δρομολογητής σε ένα δεδομενόγραμμα. Το πρωτόκολλο IP είναι μοναδικό και κάθε συσκευή που συνδέεται στο Διαδίκτυο οφείλει να το εκτελεί. Κατά τη δρομολόγηση, ένας ενδιάμεσος κόμβος του δικτύου προωθεί κάθε εισερχόμενο δεδομενόγραμμα στη διεύθυνση IP κάποιου επόμενου κόμβου, ο οποίος βρίσκεται σε πλησιέστερη απόσταση από την συσκευή που διαθέτει την διεύθυνση IP προορισμού (δηλαδή τον τελικό παραλήπτη). Η εύρεση του επόμενου κόμβου βασίζεται σε διάφορους αλγορίθμους και πρωτόκολλα δρομολόγησης, με τα οποία κάθε κόμβος διαμορφώνει έναν πίνακα δρομολόγησης (routing table) με γειτονικές διευθύνσεις IP στις οποίες μπορεί να προωθήσει δεδομενογράμματα [5]. Με βάση αυτές τις λειτουργικότητες, το πρωτόκολλο IP καθιστά δυνατή τη διαδικτύωση διαφορετικών κόμβων και δικτύων μεταξύ τους (βλ. Σχήμα 1.2) και ουσιαστικά επιτρέπει στο Internet να λειτουργεί με το γνωστό αποτελεσματικό τρόπο.

1.3.4 Επίπεδο Ζεύξης

Όπως προαναφέρθηκε, το πρωτόκολλο IP μετακινεί δεδομενογράμματα μέσα στο Διαδίκτυο από τη διεύθυνση αποστολής προς τη διεύθυνση προορισμού, χρησιμοποιώντας ενδιάμεσους δρομολογητές. Το επίπεδο ζεύξης είναι υπεύθυνο για τη μεταφορά ενός δεδομενογράμματος μεταξύ των διεπαφών δικτύου (network interfaces) δύο γειτονικών κόμβων που συνδέονται απευθείας με μία φυσική ζεύξη επικοινωνίας. Παραλαμβάνει ένα δεδομενόγραμμα από το πρωτόκολλο IP και το ενθυλακώνει σε ένα πλαίσιο (frame), το οποίο τελικά διοχετεύεται στο φυσικό επίπεδο για εκπομπή στο μέσο διάδοσης της ζεύξης. Ένα πρωτόκολλο επιπέδου ζεύξης μπορεί να παρέχει διαφορετικές υπηρεσίες, που εξαρτώνται άμεσα από τον τύπο της ζεύξης. Για παράδειγμα, μπορεί να υλοποιείται μία υπηρεσία αξιόπιστης παράδοσης που εξασφαλίζει τη σωστή μετάδοση πλαισίων αυστηρά και μόνο μεταξύ των δύο τερματικών σημείων της ζεύξης. Γνωστά πρωτόκολλα επιπέδου ζεύξης είναι τα Ethernet, Wi-Fi, Point-to-Point και κατά μία έννοια το πρωτόκολλο ραδιοζεύξης (Radio Link Protocol - RLC) των δικτύων LTE, όπως θα εξεταστεί στο Κεφάλαιο 4.

1.3.5 Φυσικό Επίπεδο

Το φυσικό επίπεδο είναι το τελευταίο στρώμα της στοίβας πρωτοκόλλων και αποτελείται από τα ηλεκτρονικά κυκλώματα και τις τεχνικές μετάδοσης ενός ηλεκτρομαγνητικού σήματος πάνω από ένα φυσικό κανάλι επικοινωνίας. Παραλαμβάνει από το επίπεδο ζεύξης ένα πλαίσιο (frame), το οποίο μεταφέρει ψηφιακή πληροφορία που περιγράφεται ως μία ακολουθία (ή ρεύμα) από bit (bit stream). Το φυσικό επίπεδο είναι υπεύθυνο για τη μετάδοση κάθε bit πάνω στη φυσική ζεύξη. Ο τρόπος μετάδοσης εξαρτάται από το είδος της ζεύξης και είναι διαφορετικός σε ένα χάλκινο σύρμα, μία οπτική ίνα ή στο ασύρματο κανάλι ενός δικτύου κινητής επικοινωνίας. Το φυσικό επίπεδο καθορίζει κρίσιμες παραμέτρους, όπως για παράδειγμα το είδος και τη στάθμη του σχήματος διαμόρφωσης, το ρυθμό κωδικοποίησης FEC (Forward Error Correction), τη συχνότητα και ισχύ εκπομπής του συστήματος και το εύρος ζώνης. Με βάση αυτές, το bitstream ενός πλαισίου μετασχηματίζεται σε ένα ηλεκτρομαγνητικό κύμα που τελικά μεταδίδεται πάνω από τη ζεύξη. Κατά τη λήψη, ακολουθείται η αντίστροφη διαδικασία, που το ηλεκτρομαγνητικό κύμα μετατρέπεται σε bitstream πλαισίου και διαβιβάζεται στο επίπεδο ζεύξης.



Σχήμα 1.8 *Εννοιολογική ροή δεδομένων της στοίβας TCP/IP στο Διαδίκτυο*
[Πηγή: https://en.wikipedia.org/wiki/Internet_protocol_suite]

Συνοψίζοντας, η λειτουργία της στοίβας πρωτοκόλλων και ο διαχωρισμός των υπηρεσιών παρουσιάζονται με τη χρήση ενός απλού μοντέλου στο Σχήμα 1.8. Ας υποτεθεί ότι δύο εφαρμογές θέλουν να επικοινωνήσουν και ότι ανάμεσα στα τερματικά παρεμβάλλονται δύο δρομολογητές. Η εφαρμογή στην πλευρά αποστολής θα παραδώσει τα δεδομένα της στο επίπεδο μεταφοράς ενώ η υπόλοιπη διαδικασία αποστολής είναι κρυμμένη από αυτή. Για το λόγο αυτό, η εμβέλεια (scope) του επιπέδου εφαρμογής είναι από διεργασία σε διεργασία (process-to-process). Το επίπεδο μεταφοράς παρέχει αξιοπιστία παράδοσης μεταξύ τερματικών (host-to-host), χωρίς να γνωρίζει την τοπολογία του υποκείμενου δικτύου. Το επίπεδο δικτύου δρομολογεί τα πακέτα δεδομένων μεταξύ γειτονικών κόμβων μέσα στο δίκτυο, αγνοώντας τις υποκείμενες ζεύξεις. Τέλος, το επίπεδο ζεύξης και το φυσικό επίπεδο ασχολούνται αποκλειστικά με

τη μεταφορά των δεδομένων πάνω από τον κατά περίπτωση τύπο καναλιού επικοινωνίας. Είναι, λοιπόν, εμφανής η αξία του διαχωρισμού επιπέδων, καθώς επιτρέπει για παράδειγμα στην υπηρεσία δρομολόγησης του πρωτοκόλλου IP να λειτουργεί με τον ίδιο τρόπο σε όλο το Internet, ανεξάρτητα από τον τύπο της ζεύξης. Η ευκολία που παρείχε αυτό το μοντέλο στο σχεδιασμό και υλοποίηση πρωτοκόλλων για κάθε επίπεδο ξεχωριστά ήταν και ο λόγος που τελικά επικράτησε για την ανάπτυξη του Διαδικτύου.

Κεφάλαιο 2. Transmission Control Protocol (TCP)

Το Πρωτόκολλο Ελέγχου Μεταφοράς (TCP) είναι το σημαντικότερο και ευρύτερα χρησιμοποιούμενο πρωτόκολλο επιπέδου μεταφοράς. Οι περισσότερες από τις εφαρμογές που παρουσιάστηκαν στην §1.3.1 βασίζονται στο TCP για την μεταφορά δεδομένων μέσω του Διαδικτύου. Το TCP ως πρωτόκολλο μεταφοράς επεκτείνει την υπηρεσία παράδοσης του πρωτοκόλλου IP μεταξύ δύο τερματικών, σε υπηρεσία παράδοσης μεταξύ των διεργασιών δύο εφαρμογών που εκτελούνται σε αυτά τα τερματικά. Στο κεφάλαιο αυτό, θα αναλυθούν οι τεχνικές και οι μηχανισμοί που έχουν υιοθετηθεί για τη λειτουργία του TCP.

2.1 Το Μοντέλο Υπηρεσίας του TCP

Είναι φανερό ότι για να λειτουργήσει σωστά η επικοινωνία μεταξύ δύο εφαρμογών μέσω του Διαδικτύου, είναι απαραίτητο τα μηνύματα που λαμβάνει ο παραλήπτης να είναι ακριβώς τα ίδια με αυτά που έστειλε ο αποστολέας. Ωστόσο, το πρωτόκολλο IP του επιπέδου δικτύου που παραδίδει δεδομενογράμματα μεταξύ τερματικών είναι ένα πρωτόκολλο «βέλτιστης προσπάθειας» όσον αφορά την παράδοση δεδομένων. Ιδιαίτερα στα κατώτερα στρώματα της στοίβας πρωτοκόλλων, πακέτα δεδομένων μπορεί να χάνονται ή να αλλοιώνονται λόγω συμφόρησης σε δρομολογητή ή κάποιας δυσμενούς κατάστασης στο δίκτυο. Το πρωτόκολλο IP δεν προσφέρει κάποια υπηρεσία που επιτρέπει την αντιμετώπιση τέτοιων περιστατικών. Την αποστολή αυτή αναλαμβάνει το TCP στο επίπεδο μεταφοράς που διαχειρίζεται όλη την επικοινωνία της διεργασίας μίας εφαρμογής με το υποκείμενο δίκτυο.

Το πρωτόκολλο TCP προσφέρει κατά βάση μία αξιόπιστη υπηρεσία παράδοσης ενός ρεύματος από byte (byte stream), μεταξύ δύο διεργασιών. Διασφαλίζει ότι το byte stream που θα μεταβιβαστεί στη διεργασία λήψης είναι ταυτόσημο με αυτό που αρχικά στάλθηκε από τη διεργασία αποστολής. Συνοπτικά, οι υπηρεσίες του TCP που θα εξεταστούν στη συνέχεια είναι οι εξής [2]:

- **Εγκατάσταση Σύνδεσης**: Μία σύνδεση TCP αποτελεί ένα «από-άκρο-σε-άκρο» λογικό κανάλι επικοινωνίας μεταξύ δύο διεργασιών που εκτελούνται σε διαφορετικά τερματικά συστήματα. Πριν ξεκινήσει η ανταλλαγή δεδομένων, είναι απαραίτητη η αρχική συμφωνία επί των τιμών ορισμένων μεταβλητών, οι οποίες θα ανανεώνονται κατά τη διάρκεια της σύνδεσης. Για το λόγο αυτό, το TCP αποτελεί ένα συνδεσμικό πρωτόκολλο (connection - oriented protocol).

- **Πολύπλεξη**: Μία εφαρμογή σε ένα τερματικό επικοινωνεί με το επίπεδο μεταφοράς μέσω ενός API που ονομάζεται socket. Ωστόσο, σε ένα τερματικό μπορούν να εκτελούνται πολλές εφαρμογές ταυτόχρονα. Επομένως, πρέπει το TCP να μπορεί να τις διαχωρίζει για να μεταβιβάζει τα δεδομένα στη σωστή socket. Για το σκοπό αυτό, οι αριθμοί θυρών (port numbers) και διευθύνσεων IP προέλευσης και προορισμού κάθε σύνδεσης TCP χρησιμοποιούνται συνδυαστικά ως μοναδικό αναγνωριστικό (unique identifier). Συνεπώς, κάθε socket μίας εφαρμογής σε ένα τερματικό είναι διαχωρίσιμη από τις υπόλοιπες με βάση αυτήν την τετράδα τιμών (4-tuple multiplexing).

- **Αξιοπιστία**: Το TCP εξασφαλίζει ότι η διεργασία λήψης θα λάβει τα δεδομένα που έστειλε η διεργασία αποστολής αναλλοίωτα και με τη σωστή σειρά. Για να το επιτύχει, χρησιμοποιεί τεχνικές θετικών επιβεβαιώσεων και αναμεταδόσεων. Ο αποστολέας διατηρεί αριθμούς ακολουθίας για τα τμήματα που έχει στείλει, ενώ ο παραλήπτης αποστέλλει θετική επιβεβαίωση όταν λαμβάνει σωστά δεδομένα. Αν κάποιο τμήμα χαθεί μέσα στο δίκτυο, ο αποστολέας δεν θα λάβει θετική επιβεβαίωση και θα αναμεταδώσει το τμήμα αυτό, με βάση έναν χρονομετρητή που διατηρεί για τους χρόνους αποστολής δεδομένων, όπως θα εξηγηθεί στη συνέχεια.

- **Έλεγχος Ροής**: Σε μία σύνδεση TCP, τα δύο άκρα χρησιμοποιούν ενταμιευτές (buffers) αποστολής και λήψης, για την προσωρινή αποθήκευση δεδομένων. Μία εφαρμογή λήψης διαβάζει δεδομένα από τον ενταμιευτή λήψης χωρίς, ωστόσο, αυτό να γίνεται απαραίτητα την ίδια στιγμή, καθώς η διεργασία μπορεί να είναι απασχολημένη. Επομένως, χωρίς κατάλληλο έλεγχο, είναι εύκολο ο αποστολέας να στείλει ταχέως πολλά δεδομένα στον παραλήπτη, με αποτέλεσμα την υπερχείλιση του ενταμιευτή λήψης. Ο έλεγχος ροής εξασφαλίζει ότι η εφαρμογή αποστολής θα αποστέλλει δεδομένα με ρυθμό που η εφαρμογή λήψης μπορεί να τα διαβάσει από τους ενταμιευτές.

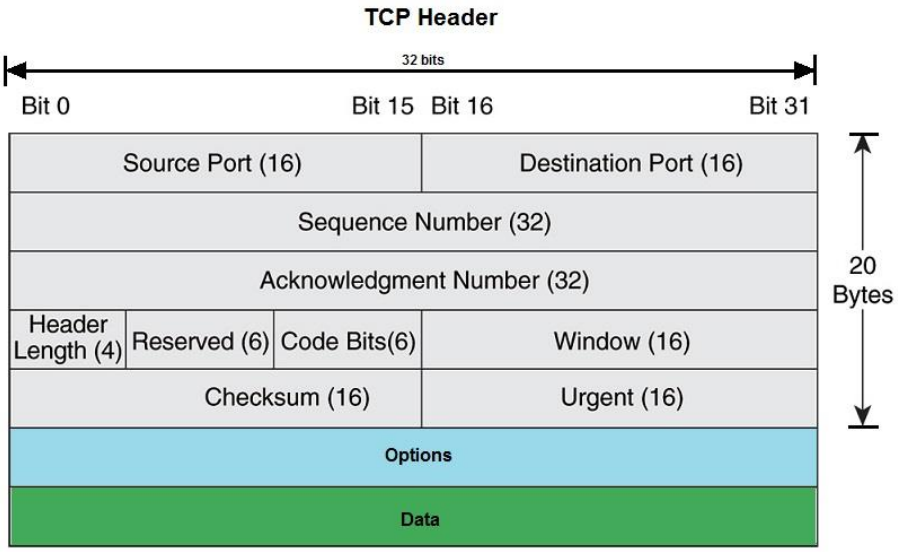
- **Έλεγχος Συμφόρησης**: Όταν ένα δίκτυο (όπως το Internet) χρησιμοποιείται ταυτόχρονα από πολλούς χρήστες, είναι φανερό ότι κάθε σύνδεση TCP δεν είναι εφικτό να αποστέλλει δεδομένα με τον υψηλότερο δυνατό ρυθμό καθώς, στην περίπτωση αυτή, θα κατέρρεαν οι ενδιάμεσες ζεύξεις και οι δρομολογητές από τον όγκο της κίνησης. Επομένως, μία σύνδεση TCP πρέπει να προσαρμόζει το ρυθμό μετάδοσής της στο εκάστοτε διαθέσιμο εύρος ζώνης του δικτύου. Αυτό επιτυγχάνεται με τη λειτουργία του ελέγχου συμφόρησης που, ίσως, αποτελεί το σημαντικότερο χαρακτηριστικό του πρωτοκόλλου TCP.

2.2 Επικεφαλίδα TCP

Η συζήτηση για το πρωτόκολλο TCP θα ξεκινήσει με την περιγραφή της δομής ενός τμήματος TCP. Όπως έχει αναφερθεί, κάθε επίπεδο της στοίβας πρωτοκόλλων προσθέτει στα δεδομένα που παραλαμβάνει από το ανώτερο επίπεδο μία επικεφαλίδα (header) πριν τα μεταβιβάσει στο κατώτερο επίπεδο, ακολουθώντας την αρχή της ενθυλάκωσης. Αντίστοιχα, το TCP στο σημείο αποστολής θα παραλάβει δεδομένα από τη διεργασία εφαρμογής και θα τους προσθέσει μία επικεφαλίδα TCP πριν τα μεταβιβάσει στο πρωτόκολλο IP για να σταλούν προς το άλλο άκρο της σύνδεσης. Είναι, επομένως, σημαντικό να παρουσιαστούν τα πεδία της επικεφαλίδας TCP, καθώς με βάση αυτήν την πληροφορία καθορίζονται οι μηχανισμοί όπου θα αναλυθούν στη συνέχεια.

Η δομή ενός τμήματος TCP (TCP segment) απεικονίζεται στο Σχήμα 2.1. Αποτελείται από την επικεφαλίδα και το πεδίο δεδομένων που μεταφέρει το μήνυμα της εφαρμογής. Το ελάχιστο μήκος της επικεφαλίδας TCP είναι 20 bytes, ενώ το μέγιστο επιτρεπτό μέγεθος είναι 60 bytes, εφόσον χρησιμοποιηθούν και προαιρετικά πεδία. Το γεγονός ότι το TCP είναι σαφώς περισσότερο περίπλοκο πρωτόκολλο σε σχέση με το UDP αντανακλάται και στο μεγαλύτερο μέγεθος επικεφαλίδας (8 bytes συγκριτικά με τουλάχιστον 20 bytes), δεδομένου ότι η λειτουργία των μηχανισμών του TCP απαιτεί

ανταλλαγή περισσότερης πληροφορίας μεταξύ των δύο τερματικών που βρίσκονται στα άκρα της σύνδεσης TCP.



Σχήμα 2.1 *Η δομή του τμήματος TCP*

[Πηγή: https://www.networkurge.com/2017/10/tcp-header-details.html]

Τα πεδία που περιέχονται σε ένα τμήμα TCP είναι τα ακόλουθα:

- **Θύρα Προέλευσης – 16 bits**: Τίθεται από τη διεργασία αποστολής του τμήματος TCP και χρησιμοποιείται για τη διαδικασία πολύπλεξης/αποπολύπλεξης δεδομένων από/προς το στρώμα εφαρμογής. Η τιμή της μπορεί να ανήκει στους γνωστούς αριθμούς θυρών, αν ο αποστολέας είναι κάποια διεργασία εξυπηρετητή (π.χ. θύρα 80 για Web Server). Σε άλλη περίπτωση, η τιμή της θύρας προέλευσης επιλέγεται αυθαίρετα μέσα στο επιτρεπτό εύρος.

- **Θύρα Προορισμού – 16 bits**: Τίθεται επίσης από τη διεργασία αποστολής και χρησιμοποιείται όπως προηγουμένως. Όταν ο αποστολέας είναι κάποια διεργασία πελάτη (client), τότε η τιμή της είναι κάποιος γνωστός αριθμός θύρας, που αντιστοιχεί στην επιθυμητή εφαρμογή (βλ. §1.3.1).

- **Αριθμός Ακολουθίας – 32 bits**: Ένας TCP αποστολέας χρησιμοποιεί το πεδίο αυτό για να αριθμεί τα πακέτα που αποστέλλει. Ουσιαστικά πρόκειται για ένα αύξοντα αριθμό που αποδίδεται σε κάθε τμήμα TCP και χρησιμεύει στην υλοποίηση της αξιόπιστης μεταφοράς, όπως αναλύεται στη συνέχεια.

- **Αριθμός Επιβεβαίωσης – 32 bits**: Όταν το bit ελέγχου της σημαίας (flag) ACK στην επικεφαλίδα έχει τεθεί στην τιμή 1, τότε το πεδίο αυτό εκφράζει τον επόμενο αριθμό ακολουθίας που περιμένει ο αποστολέας της επιβεβαίωσης από τη διεργασία στο άλλο άκρο της σύνδεσης TCP. Ο αριθμός επιβεβαίωσης χρησιμοποιείται από το μηχανισμό αξιόπιστης μεταφοράς για τη δήλωση και αναγνώριση επιτυχούς λήψης δεδομένων.

- **Μήκος Επικεφαλίδας – 4 bits**: Καθώς η επικεφαλίδα TCP μπορεί να είναι από 20 έως 60 bytes, το πεδίο αυτό δηλώνει το μήκος της (σε λέξεις των 32-bit). Με τον τρόπο αυτό, ο παραλήπτης του τμήματος είναι σε θέση να ξεχωρίσει την επικεφαλίδα από τα ενθυλακωμένα δεδομένα εφαρμογής.

- **Σημαίες – 6 bits**: Το πεδίο αυτό αποτελείται από έξι bit ελέγχου, καθένα από τα οποία εκφράζει την ενεργοποίηση ή όχι της αντίστοιχης σημαίας, ανάλογα με την τιμή του (0 ή 1). Το ACK bit δηλώνει ότι ο αριθμός επιβεβαίωσης που μεταφέρει το τμήμα είναι έγκυρος. Το SYN bit τίθεται 1 στα πρώτα πακέτα που ανταλλάσσονται κατά τη διαδικασία εγκατάστασης σύνδεσης. Το FIN bit τίθεται ένα 1 στο τελευταίο πακέτο που στέλνει ο αποστολέας και υποδηλώνει το τέλος της σύνδεσης. Το RST bit τίθεται 1 όταν συμβεί κάποιο απροσδόκητο γεγονός και η σύνδεση πρέπει να επανεκκινηθεί (connection reset). Οι δύο τελευταίες σημαίες είναι οι PSH (push) και URG (urgent pointer), οι οποίες ωστόσο δεν χρησιμοποιούνται στην πράξη.

- **Παράθυρο Λήψης – 16 bits**: Το πεδίο αυτό χρησιμοποιείται στον έλεγχο ροής και δηλώνει το διαθέσιμο αποθηκευτικό χώρο στον ενταμιευτή λήψης, δηλαδή το μέγιστο αριθμό από bytes που μπορεί να λάβει ο παραλήπτης της σύνδεσης TCP.

- **Άθροισμα Ελέγχου – 16 bits**: Όπως και στις επικεφαλίδες άλλων πρωτοκόλλων, το πεδίο αυτό χρησιμοποιείται για την ανίχνευση σφαλμάτων στο τμήμα TCP, αναγνωρίζοντας έτσι τα αλλοιωμένα πακέτα.

- **Επιλογές – 0 έως 40 bytes**: Το πεδίο αυτό είναι προαιρετικό και έχει μεταβλητό μήκος. Χρησιμοποιείται από τα δύο άκρα της σύνδεσης TCP όταν θέλουν να διαπραγματευτούν ορισμένες πρόσθετες παραμέτρους. Οι σημαντικότερες είναι το μέγιστο μέγεθος τμήματος (maximum segment size - MSS), ο παράγοντας κλίμακας παραθύρου (window scaling) και οι επιλεκτικές επιβεβαιώσεις (selective acknowledgements), οι οποίες θα αναλυθούν στη συνέχεια.

- **Δεδομένα**: Εδώ περιέχονται τα δεδομένα της εφαρμογής. Το μέγιστο επιτρεπτό μέγεθος δεδομένων που μπορεί να μεταφέρονται σε ένα τμήμα TCP καθορίζεται από το MSS.

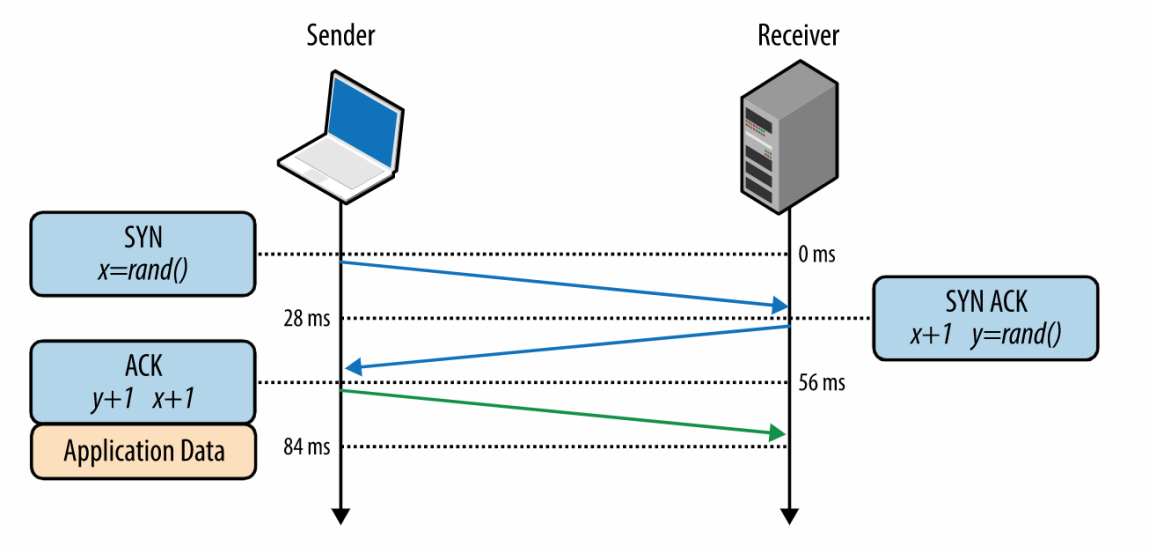
Συνολικά, η λειτουργία του TCP βασίζεται στην ανταλλαγή πληροφοριών μεταξύ των δύο επικοινωνούντων τερματικών, οι οποίες μεταφέρονται μέσω των προαναφερθέντων πεδίων επικεφαλίδας. Με βάση αυτές τις πληροφορίες, καθορίζονται οι ενέργειες που πρέπει να γίνουν στη σύνδεση TCP.

2.3 Η Σύνδεση TCP

Στην παράγραφο αυτή θα αναλυθεί η αρχή και το τέλος μιας σύνδεσης TCP. Αν και αποτελούν σχετικά απλές διαδικασίες είναι αρκετά σημαντικές, ιδιαίτερα η εγκατάσταση της σύνδεσης. Επειδή το TCP είναι συνδεσμικό πρωτόκολλο, η διαδικασία αυτή είναι απαραίτητο να πραγματοποιηθεί πριν από την ανταλλαγή των πρώτων δεδομένων εφαρμογής και ενδέχεται να προσθέτει σημαντικές καθυστερήσεις, οι οποίες γίνονται αντιληπτές από το χρήστη.

2.3.1 Εγκατάσταση Σύνδεσης - Η Τριμερής Χειραψία

Ας υποτεθεί ότι ένας χρήστης θέλει να περιηγηθεί σε κάποια ιστοσελίδα του Διαδικτύου. Για το σκοπό αυτό, η διεργασία του επιπέδου εφαρμογής στο χρήστη (client) πρέπει να ανταλλάξει μηνύματα HTTP με το επίπεδο εφαρμογής στον εξυπηρετητή (server). Έτσι, η εφαρμογή πελάτη πληροφορεί το TCP ότι επιθυμεί να ανοίξει μία σύνδεση προς τη διεργασία εξυπηρετητή που εκτελείται στο server. Στη συνέχεια, το TCP προχωρά στην ανταλλαγή συγκεκριμένων segments με το TCP στον εξυπηρετητή, τα οποία εγκαθιστούν τη σύνδεση. Η αλληλουχία αυτών των πακέτων δίνεται στο Σχήμα 2.2 και αναλύεται ακολούθως:



Σχήμα 2.2 *Η ανταλλαγή τμημάτων κατά την τριμερή χειραψία του TCP*
[Πηγή: https://hpbn.co/building-blocks-of-tcp/]

- Στο πρώτο βήμα για την εγκατάσταση σύνδεσης, ο πελάτης αποστέλλει στον εξυπηρετητή ένα ειδικό τμήμα TCP, το οποίο δεν περιέχει δεδομένα εφαρμογής και η σημαία SYN στην επικεφαλίδα τίθεται ίση με 1. Για το λόγο αυτό, το πρώτο τμήμα της σύνδεσης TCP ονομάζεται και τμήμα SYN. Η τιμή της θύρας προορισμού αντιστοιχεί σε κάποια γνωστή θύρα, όπως για παράδειγμα η θύρα 80 για τις περιπτώσεις των HTTP servers. Επιπλέον, ο πελάτης επιλέγει τυχαία ένα αρχικό αριθμό ακολουθίας (έστω *x*), τον οποίο τοποθετεί στο αντίστοιχο πεδίο της επικεφαλίδας. Με αυτό τον τρόπο, πληροφορεί τον εξυπηρετητή ότι η αρίθμηση των τμημάτων που θα αποστέλλονται από τη δική του πλευρά θα ξεκινήσει από αυτόν τον αριθμό. Αυτό είναι ιδιαίτερα σημαντικό, καθώς έτσι υλοποιείται η υπηρεσία αξιόπιστης παράδοσης του TCP που αναλύεται στην §2.4. Τέλος, το τμήμα αυτό ενθυλακώνεται σε ένα δεδομενόγραμμα IP και αποστέλλεται προς την διεύθυνση IP του server.

- Υποθέτοντας ότι το πρώτο τμήμα SYN φθάνει στον προορισμό του, ο εξυπηρετητής δεσμεύει τους ενταμιευτές και μεταβλητές για τη σύνδεση και απαντά με ένα ειδικό τμήμα αποδοχής σύνδεσης, το οποίο επίσης δεν περιέχει δεδομένα εφαρμογής. Οι σημαίες SYN και ACK τίθεται στην τιμή 1. Για το λόγο αυτό, το τμήμα αυτό ονομάζεται και τμήμα SYNACK. Ο αριθμός επιβεβαίωσης

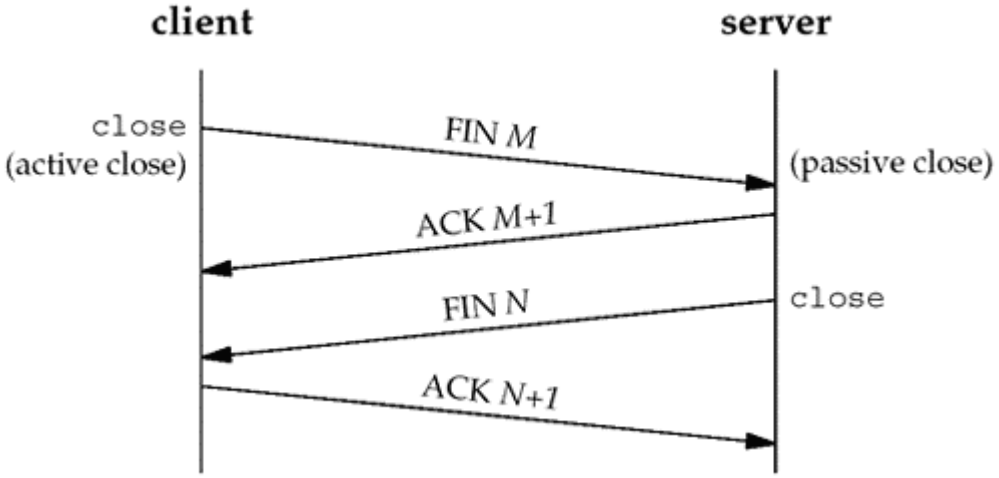
στην επικεφαλίδα τίθεται στην τιμή *x+1*. Με τον τρόπο αυτό ο server επιβεβαιώνει ότι έλαβε το αρχικό τμήμα με αριθμό ακολουθίας *x*, ενώ το επόμενο τμήμα που περιμένει στη σωστή σειρά πρέπει να έχει αριθμό ακολουθίας *x+1*. Τέλος, ο server επιλέγει τυχαία το δικό του αρχικό αριθμό ακολουθίας (έστω *y*) και τον τοποθετεί στο αντίστοιχο πεδίο της επικεφαλίδας TCP.

- Με τη λήψη του τμήματος SYNACK, ο πελάτης επίσης δεσμεύει ενταμιευτές και μεταβλητές για τη σύνδεση. Στη συνέχεια, αποστέλλει ένα τελευταίο τμήμα προς τον εξυπηρετητή, το οποίο επιβεβαιώνει το τμήμα SYNACK. Η σημαία SYN τίθεται στην τιμή 0 καθώς η εγκατάσταση της σύνδεσης έχει ολοκληρωθεί, ενώ η σημαία ACK τίθεται στην τιμή 1. Ο αριθμός ακολουθίας έχει την τιμή *x+1*, ενώ ο αριθμός επιβεβαίωσης τίθεται κατ' αντιστοιχία στην τιμή *y+1*. Το τμήμα αυτό μπορεί, πλέον, να μεταφέρει δεδομένα εφαρμογής, όπως για παράδειγμα ένα αίτημα "HTTP GET".

Με την ολοκλήρωση της ανταλλαγής των τριών αυτών μηνυμάτων η σύνδεση έχει εγκατασταθεί και τα δύο τερματικά μπορούν να ανταλλάσσουν δεδομένα εφαρμογής χρησιμοποιώντας το TCP ως πρωτόκολλο μεταφοράς. Η διαδικασία αυτή έχει γίνει γνωστή ως η «τριμερής χειραψία» του TCP.

2.3.2 Τερματισμός Σύνδεσης

Μία σύνδεση TCP μπορεί να τερματιστεί από οποιοδήποτε άκρο της σύνδεσης. Κατά τον τερματισμό, ότι έχει δεσμευτεί, όπως ενταμιευτές και μεταβλητές, απελευθερώνονται. Στο Σχήμα 2.3 δίνεται η περίπτωση που ο πελάτης αποφασίζει να κλείσει τη σύνδεση. Μία εντολή close μεταβιβάζεται από την εφαρμογή προς το TCP, το οποίο στη συνέχεια αποστέλλει προς το server ένα ειδικό τμήμα με τη σημαία FIN να έχει την τιμή 1. Ο εξυπηρετητής απαντά με μία επιβεβαίωση και το δικό του αντίστοιχο τμήμα FIN. Τέλος, ο πελάτης επιβεβαιώνει το τμήμα FIN του εξυπηρετητή και ύστερα από τη λήξη ενός μετρητή, η σύνδεση θεωρείται ότι έχει τερματιστεί.

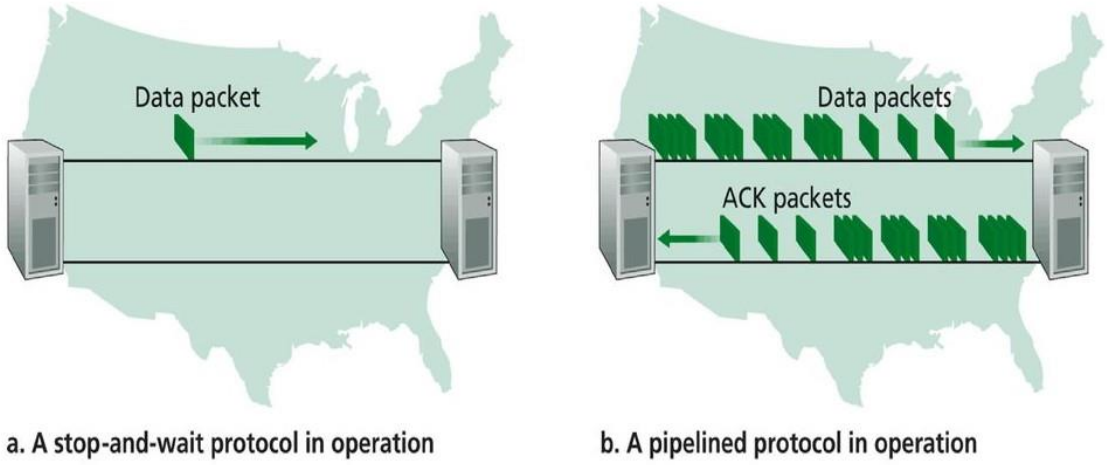


Σχήμα 2.3 *Τερματισμός σύνδεσης TCP*
[Πηγή: http://intronetworks.cs.luc.edu/]

2.4 Αξιόπιστη Μεταφορά Δεδομένων

Καθώς το Διαδίκτυο αποτελεί ένα μη αξιόπιστο δίκτυο στο οποίο τα πακέτα που μεταδίδονται μπορεί να αλλοιώνονται ή να χάνονται, το TCP πρέπει να υλοποιήσει μία υπηρεσία αξιόπιστης παράδοσης στο επίπεδο μεταφοράς. Για την επίτευξη αξιόπιστης παράδοσης, ο αποστολέας χρησιμοποιεί αριθμούς ακολουθίας για να αριθμεί τα πακέτα ενώ ο παραλήπτης χρησιμοποιεί αριθμούς επιβεβαίωσης για ορθά παραληφθέντα πακέτα. Υπενθυμίζεται ότι μία σύνδεση TCP είναι αμφίδρομη, επομένως ένα τερματικό συνήθως λειτουργεί ταυτόχρονα και ως αποστολέας και ως παραλήπτης.

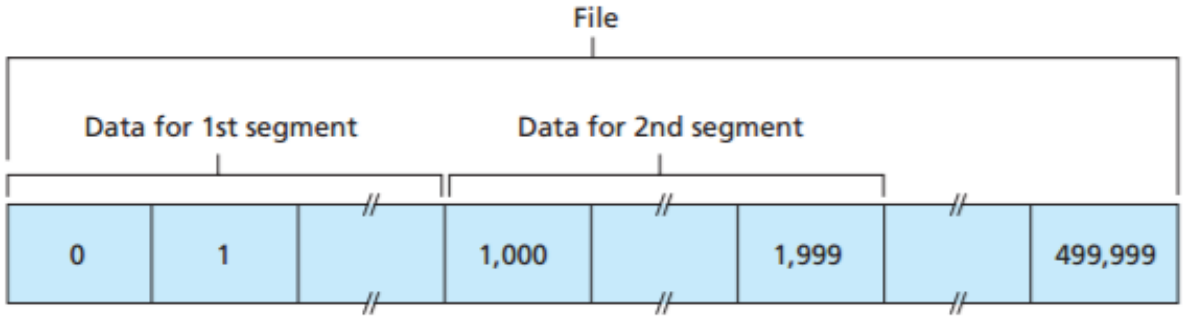
Πριν από την ανάλυση των μηχανισμών αυτών, ας γίνει μία αναλογία της αξιόπιστης παράδοσης στην περίπτωση της ανθρώπινης επικοινωνίας. Ας υποτεθεί ότι δύο άνθρωποι συνομιλούν μέσω τηλεφώνου και θέλουν να βεβαιωθούν ότι ένα μήνυμα με πολλές προτάσεις έχει παραληφθεί σωστά στον προορισμό. Μία απλή προσέγγιση θα ήταν ο αποστολέας να εκφωνεί μία λέξη και να περιμένει ένα «ΟΚ» από τον παραλήπτη, πριν εκφωνήσει την επόμενη λέξη. Αν η λέξη δεν ακουστεί καθαρά, ο παραλήπτης θα ζητήσει την επανάληψή της. Κατ' αντιστοιχία, ένα τερματικό θα μπορούσε να αποστέλλει ένα πακέτο δεδομένων και να περιμένει επιβεβαίωση ACK πριν στείλει το επόμενο. Πρωτόκολλα που υιοθετούν αυτόν τον τρόπο λειτουργίας είναι γνωστά ως «πρωτόκολλα παύσης και αναμονής» (stop and wait). Ωστόσο, στην περίπτωση που το μήνυμα περιέχει πολλές προτάσεις, η εκφώνηση κάθε λέξης ξεχωριστά και η αναμονή για επιβεβαίωση δεν είναι αποδοτική επικοινωνία. Μία καλύτερη προσέγγιση θα ήταν ο ομιλητής να εκφωνεί μία ολόκληρη πρόταση και ύστερα να περιμένει για το «ΟΚ» του παραλήπτη. Αντίστοιχα, μπορεί να επιτρέπεται σε ένα τερματικό να αποστέλλει πολλά πακέτα δεδομένων χωρίς να περιμένει κάποια επιβεβαίωση. Η τεχνική αυτή είναι γνωστή ως «διοχέτευση» (pipelining) και ουσιαστικά αυξάνει τον βαθμό χρησιμοποίησης του αποστολέα. Ο επιτρεπτός αριθμός των μη επιβεβαιωμένων πακέτων που μπορούν να βρίσκονται σε διέλευση (in flight) περιορίζεται από ένα μέγιστο αριθμό N κάθε χρονική στιγμή. Το μέγεθος N ονομάζεται γενικά «παράθυρο» και έχει σημαντικό ρόλο στη λειτουργία των αξιόπιστων πρωτοκόλλων μεταφοράς, όπως το TCP.



Σχήμα 2.4 *Οι δύο βασικές προσεγγίσεις για την υλοποίηση αξιόπιστης μεταφοράς* [2]

2.4.1 Αριθμός Ακολουθίας

Το TCP στην πλευρά του αποστολέα παραλαμβάνει από το επίπεδο εφαρμογής δεδομένα υπό τη μορφή ρεύματος από bytes, το οποίο πρέπει να μεταδοθεί με την ίδια ακριβώς σειρά στο TCP του παραλήπτη. Οι αριθμοί ακολουθίας αντιπροσωπεύουν τη σειριακή διάταξη των bytes μέσα στο ρεύμα δεδομένων, και αποστέλλονται προκειμένου ο παραλήπτης να μπορεί να τα τοποθετήσει στη σωστή σειρά, πριν τα διαβιβάσει στην εφαρμογή προορισμού. Ο αριθμός ακολουθίας που τοποθετείται στο αντίστοιχο πεδίο της επικεφαλίδας TCP είναι ο αύξων αριθμός του πρώτου byte του τμήματος TCP, ως προς το πρώτο byte (με αριθμό ακολουθίας 0) του συνολικού ρεύματος δεδομένων. Ας θεωρηθεί ένα απλό παράδειγμα όπου ένας client θέλει να λάβει ένα αρχείο μεγέθους 500.000 bytes από κάποιο server. Αρχικά, το συνολικό αυτό byte stream θα τεμαχιστεί σε τμήματα μήκους MSS, καθώς αυτό είναι το μέγιστο επιτρεπτό μέγεθος δεδομένων εφαρμογής που μπορεί να μεταφέρει ένα τμήμα TCP (βλ. §2.2). Αν υποτεθεί ότι το μέγεθος MSS είναι 1000 bytes, θα δημιουργηθούν συνολικά 500 τμήματα TCP, όπως φαίνεται στο Σχήμα 2.5. Το πρώτο τμήμα θα έχει αριθμό ακολουθίας 0 και περιέχει τα bytes από 0 έως 999. Επισημαίνεται ότι το δεύτερο τμήμα δεν θα έχει αριθμό ακολουθίας 2 αλλά 1000, καθώς το 1ο byte του δεύτερου τμήματος είναι το 1000ο byte μέσα στο συνολικό ρεύμα δεδομένων. Αντίστοιχα, το τρίτο τμήμα θα έχει αριθμό ακολουθίας 3000, το τέταρτο 4000 κ.λπ. Η χρήση αυτής της αρίθμησης, σε συνδυασμό με τη χρήση επιβεβαιώσεων, επιτρέπει στον αποστολέα να διαχειρίζεται τις απώλειες πακέτων, καθώς κάθε byte είναι μοναδικά αριθμημένο και άρα μπορεί εύκολα να αναμεταδώσει συγκεκριμένους αριθμούς ακολουθίας.



Σχήμα 2.5 *Τεμαχισμός αρχείου σε τμήματα TCP και ανάθεση αριθμών ακολουθίας* [2]

2.4.2 Αριθμός Επιβεβαίωσης

Ο δεύτερος κρίκος για την υλοποίηση της αξιόπιστης παράδοσης δεδομένων είναι η παροχή ανάδρασης από τον παραλήπτη στον αποστολέα, σχετικά με την έγκυρη παραλαβή πακέτων. Όταν ένας παραλήπτης λάβει ένα πακέτο TCP αποστέλλει ένα αριθμό επιβεβαίωσης ως απάντηση, ο οποίος είναι ο αριθμός ακολουθίας του επόμενου byte που περιμένει να λάβει στη σωστή σειρά. Χρησιμοποιώντας το προηγούμενο παράδειγμα, αν υποτεθεί ότι ο παραλήπτης έχει λάβει σωστά τα πρώτα δύο πακέτα, δηλαδή τα bytes 0 έως 1999, τότε ο αριθμός επιβεβαίωσης που θα στείλει προς τον αποστολέα στο αντίστοιχο πεδίο της επικεφαλίδας θα είναι ο 2000. Ως άλλο παράδειγμα, ας υποτεθεί ότι ο παραλήπτης έχει λάβει σωστά τα bytes 0 έως 1999 και 3000 έως 3999, ενώ δεν έχει λάβει ακόμα τα bytes 2000 έως 2999. Επειδή ο παραλήπτης περιμένει ακόμα αυτά τα bytes προκειμένου να αναδημιουργήσει σωστά το ρεύμα δεδομένων, ο αριθμός

ακολουθίας που θα στείλει είναι ο 2000, καίτοι έχει λάβει σωστά και bytes με μεγαλύτερους αριθμούς ακολουθίας. Καθώς το TCP επιβεβαιώνει μόνο bytes που λαμβάνονται σωστά και σε διαδοχικές θέσεις στο συνολικό ρεύμα δεδομένων, λέγεται ότι το TCP παρέχει συσσωρευτικές επιβεβαιώσεις (cumulative acknowledgements).

Ωστόσο, η αρχική αυτή σχεδίαση των επιβεβαιώσεων μπορεί να προκαλέσει μείωση στην αποδοτικότητα του πρωτοκόλλου. Στο προηγούμενο παράδειγμα, ο παραλήπτης δεν είχε λάβει τα bytes 2000 έως 2999 ενώ είχε λάβει τα επόμενα, γεγονός όμως για το οποίο δεν μπορούσε να ειδοποιήσει τον αποστολέα. Επομένως, ο αποστολέας ενδέχεται να αναμεταδώσει τα bytes 2000 έως 3999, ενώ στην πραγματικότητα χρειάζεται να στείλει ξανά μόνο τα bytes 2000 έως 2999. Για την αντιμετώπιση αυτού του φαινομένου, προτάθηκε το RFC 2018, το οποίο εισήγαγε τις λεγόμενες «επιλεκτικές επιβεβαιώσεις» (selective acknowledgements) στο πεδίο επιλογών της επικεφαλίδας του TCP. Ο μηχανισμός αυτός επιτρέπει στον παραλήπτη την επιβεβαίωση της ορθής λήψης τμημάτων TCP, τα οποία ωστόσο έχουν φθάσει εκτός σειράς και έχουν μεγαλύτερους αριθμούς ακολουθίας από αυτόν που αναμένει ο παραλήπτης, με βάση την τελευταία επιβεβαίωση ACK που έστειλε. Για το σκοπό αυτό, ο παραλήπτης συμπεριλαμβάνει στην επικεφαλίδα δύο αριθμούς, που αντιπροσωπεύουν ένα συνεχόμενο block από TCP τμήματα τα οποία επιβεβαιώνονται εκτός σειράς: η «αριστερή ακμή» (left edge) είναι ο μικρότερος αριθμός ακολουθίας, ενώ η «δεξιά ακμή» (right edge) ο μεγαλύτερος αριθμός ακολουθίας αυτού του block. Ένα τμήμα TCP από τον παραλήπτη προς τον αποστολέα μπορεί να επιβεβαιώνει με τη χρήση αυτών των «ακμών» περισσότερα από ένα τέτοια block, τα οποία ονομάζονται και SACK blocks.

Στο προηγούμενο παράδειγμα έχει υποτεθεί ότι τα bytes 2000 έως 2999 έχουν χαθεί, ενώ τα bytes 3000 έως 3999 έχουν ληφθεί σωστά αλλά εκτός σειράς. Στην περίπτωση αυτή, ο παραλήπτης θα στείλει μία συσσωρευτική επιβεβαίωση όπως και προηγουμένως, αλλά θα περιλαμβάνει και την επιλεκτική επιβεβαίωση. Στο πεδίο του αριθμού επιβεβαίωσης στην επικεφαλίδα θα τοποθετηθεί ο αριθμός 2000, καθώς αυτός είναι ο επόμενος αριθμός ακολουθίας που περιμένει ο παραλήπτης για να αναδομήσει το συνολικό ρεύμα δεδομένων. Για την επιλεκτική επιβεβαίωση, η «αριστερή ακμή» θα είναι 3000 και η «δεξιά ακμή» 3999, σηματοδοτώντας έτσι ότι το TCP block με αριθμούς ακολουθίας από 3000 έως 3999 έχει παραληφθεί σωστά. Συνεπώς, ο αποστολέας ύστερα από την αναμετάδοση του χαμένου τμήματος, μπορεί να συνεχίσει την αποστολή δεδομένων από το τμήμα με αριθμό ακολουθίας 4000, αυξάνοντας την αποδοτικότητα με μείωση των άσκοπων αναμεταδόσεων.

2.4.3 Ανίχνευση Απωλειών

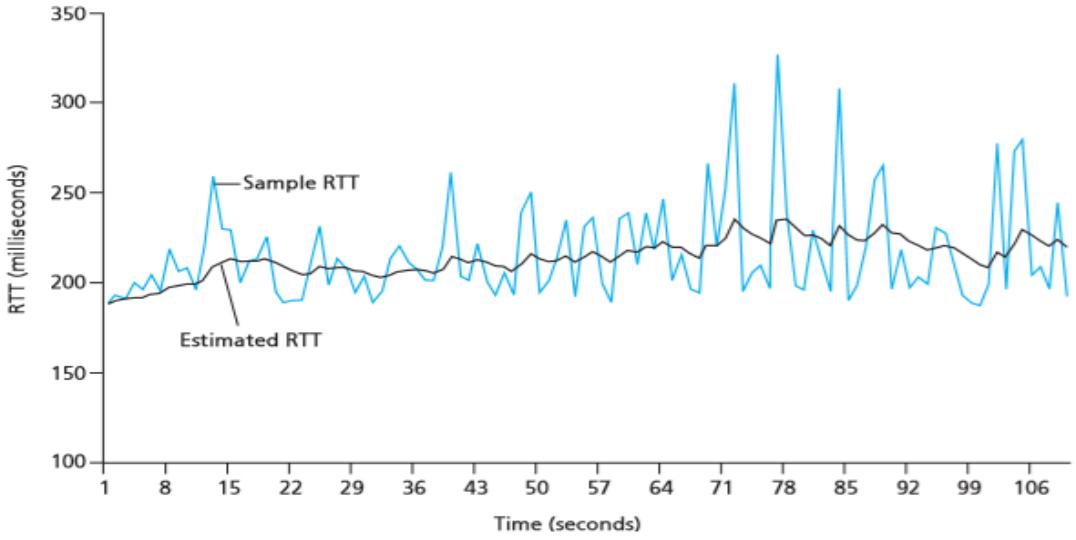
Παρουσιάστηκε προηγουμένως ο τρόπος με τον οποίο ένας TCP αποστολέας αριθμεί τα bytes ενός ρεύματος δεδομένων, αλλά και η διαδικασία που ακολουθεί ένας TCP παραλήπτης για να επιβεβαιώσει την ορθή λήψη τους. Ωστόσο, αυτά τα πακέτα δεδομένων ή επιβεβαιώσεων μπορεί να χαθούν, καθώς διαδίδονται μέσα στο αναξιόπιστο δίκτυο του Internet. Πρέπει, επομένως, να οριστούν κάποιοι μηχανισμοί με τους οποίους το TCP θα αναγνωρίζει τέτοια περιστατικά απωλειών. Στις σημερινές υλοποιήσεις του TCP, η ανίχνευση αυτή γίνεται με δύο τρόπους:

- **Διάστημα Λήξης Χρόνου Αναμετάδοσης**

Το TCP χρησιμοποιεί ένα μηχανισμό λήξης χρονομετρητή προκειμένου να αναμεταδώσει ένα χαμένο τμήμα και να ανακάμψει από κάποιο περιστατικό απώλειας. Η τιμή του χρονομετρητή, που ονομάζεται διάστημα λήξης χρόνου, πρέπει να είναι μεγαλύτερη από την τιμή του χρόνου μετ' επιστροφής (Round Trip Time - RTT) της σύνδεσης, δηλαδή του χρονικού διαστήματος μεταξύ της αποστολής ενός τμήματος και της λήψης της επιβεβαίωσής του. Ένα βασικό ζήτημα για αυτή την υλοποίηση είναι ο τρόπος εκτίμησης του RTT ανάμεσα στον αποστολέα και τον παραλήπτη. Αρχικά, το TCP λαμβάνει ένα δείγμα του χρόνου RTT, που το συμβολίζουμε ως *SampleRTT*, για κάποιο από τα τμήματα που έχουν μεταδοθεί αλλά δεν έχουν ακόμα επιβεβαιωθεί. Με αυτό τον τρόπο, λαμβάνεται ένα δείγμα *SampleRTT* περίπου κάθε RTT. Ωστόσο, αυτά τα δείγματα θα εμφανίζουν υψηλή διακύμανση, καθώς κάθε τμήμα υφίσταται διαφορετικές καθυστερήσεις στους ενδιάμεσους δρομολογητές και στα τερματικά συστήματα. Είναι, επομένως, λογικό να λαμβάνεται ένα είδος μέσης τιμής, που ανανεώνεται δυναμικά με κάθε νέο δείγμα του RTT, για τον υπολογισμό μιας καλής εκτίμησης του χρόνου RTT. Αυτή η μέση τιμή ονομάζεται *EstimatedRTT* και υπολογίζεται ως μία «εκθετικά σταθμισμένη κινητή μέση τιμή» (exponential weighted moving average - EWMA) των δειγμάτων RTT με βάση τον τύπο [6]:

$$EstimatedRTT = (1 - a) \cdot EstimatedRTT + a \cdot SampleRTT \qquad (2.1)$$

Μία προτεινόμενη τιμή για το a είναι 0.125, ενώ η *EstimatedRTT* αρχικοποιείται συνήθως στην τιμή του 1s. Το Σχήμα 2.6 δείχνει τις τιμές αυτών των μεταβλητών για μία σύνδεση TCP. Ο υπολογισμός της EWMA εξομαλύνει ενδεχόμενες ακραίες τυχαίες διακυμάνσεις που μπορεί να εμφανίζουν ορισμένα δείγματα του RTT. Συνεπώς, επιτυγχάνεται συνολικά καλύτερη εκτίμηση του RTT, καθώς προκύπτει από μετρήσεις που λαμβάνονται καθ' όλη τη διάρκεια της σύνδεσης TCP.



Σχήμα 2.6 *Εκτίμηση του χρόνου μετ' επιστροφής RTT με δείγματα SampleRTT*
[Πηγή: https://wushouyuan.com/posts/d23f7b3e/]

Εκτός από την εκτίμηση του χρόνου RTT, είναι απαραίτητο και ένα μέτρο για τη διακύμανσή του. Για το σκοπό αυτό, ορίζεται η απόκλιση του RTT που συμβολίζεται ως *DevRTT* και μετρά τη διακύμανση των δειγμάτων *SampleRTT* από την εκτίμηση *EstimatedRTT*. Ο υπολογισμός της γίνεται σύμφωνα με την εξίσωση:

$$DevRTT = (1 - \beta) \cdot DevRTT + \beta \cdot |SampleRTT - EstimatedRTT| \qquad (2.2)$$

Επισημαίνεται ότι η *DevRTT* υπολογίζεται επίσης ως μία εκθετικά σταθμισμένη κινητή μέση τιμή, ενώ η προτεινόμενη τιμή για το β είναι 0.25.

Με προσδιορισμένες τις τιμές των *EstimatedRTT* και *DevRTT*, το TCP μπορεί να καθορίσει ένα διάστημα λήξης χρόνου, με το πέρας του οποίου το τμήμα με το μικρότερο αριθμό ακολουθίας που έχει σταλεί αλλά δεν έχει επιβεβαιωθεί θα θεωρηθεί χαμένο, οπότε θα αναμεταδοθεί. Το διάστημα αυτό πρέπει να είναι τουλάχιστον ίσο με την εκτίμηση του RTT αλλά όχι πολύ μεγαλύτερο, προκειμένου να μην καθυστερούν πολύ οι αναμεταδόσεις χαμένων πακέτων. Λαμβάνοντας υπόψη τα προηγούμενα, το διάστημα λήξης χρόνου για αναμετάδοση δίνεται από την εξίσωση:

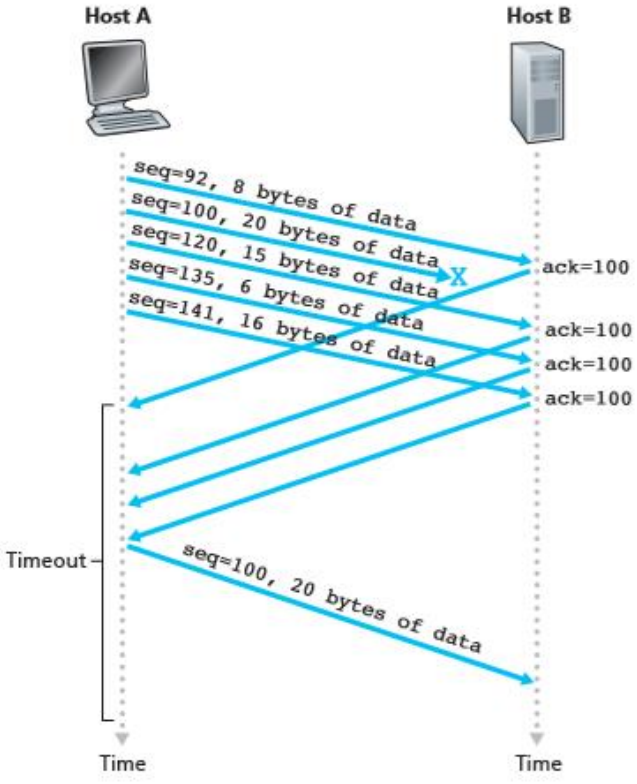
$$TimeoutInterval = EstimatedRTT + 4 \cdot DevRTT \qquad (2.3)$$

Η προτεινόμενη αρχική τιμή για το *TimeoutInterval* είναι το 1s. Τέλος, επισημαίνεται ότι όταν συμβεί λήξη χρόνου, ο αποστολέας TCP αναμεταδίδει το κατάλληλο τμήμα και η τιμή της *TimeoutInterval* διπλασιάζεται, οδηγώντας σε εκθετική αύξηση των διαστημάτων λήξης χρόνου μετά από κάθε αναμετάδοση. Επειδή η λήξη του χρονομετρητή οφείλεται συνήθως σε συμφόρηση, είναι αναγκαία αυτή η αύξηση στα διαστήματα λήξης προκειμένου οι αποστολείς TCP να αναμεταδίδουν πακέτα ολοένα και σπανιότερα, παρέχοντας χρόνο στο δίκτυο να αποσυμφορηθεί.

- **Λήψη Διπλότυπων ACK**

Αν και ο προηγούμενος μηχανισμός με τα διαστήματα λήξης χρόνου λειτουργεί, δεν είναι πάντα αποδοτικός. Συχνά, το διάστημα λήξης χρόνου *TimeoutInterval* είναι μεγάλο οπότε ο αποστολέας θα καθυστερήσει αρκετά να αναμεταδώσει κάποιο χαμένο τμήμα TCP, αυξάνοντας έτσι τη συνολική καθυστέρηση. Η λύση στο πρόβλημα δίνεται από τις επιβεβαιώσεις TCP, με την παρατήρηση των επονομαζόμενων διπλότυπων ACK. Υπενθυμίζεται ότι όταν ένας παραλήπτης TCP λαμβάνει κάποιο τμήμα με αριθμό ακολουθίας μεγαλύτερο από τον αναμενόμενο, πιθανότατα λόγω απώλειας πακέτου, αποστέλλει μία επιβεβαίωση που περιέχει στο πεδίο της επικεφαλίδας το σωστά αναμενόμενο αριθμό ακολουθίας. Σε συνδυασμό με το ότι ο αποστολέας TCP αποστέλλει πολλά πακέτα μαζί σε μορφή διοχέτευσης, αν κάποιο πακέτο χαθεί και τα υπόλοιπα ληφθούν σωστά, τότε ο παραλήπτης θα στείλει πολλά ACK με τον ίδιο αριθμό επιβεβαίωσης (διπλότυπα ACK). Όταν ο αποστολέας TCP δεχθεί τρία διαδοχικά διπλότυπα ACK πέραν του αρχικού, αντιλαμβάνεται ως χαμένο το τμήμα TCP με αριθμό ακολουθίας ίσο με τον αριθμό επιβεβαίωσης του διπλότυπου ACK και εκτελεί μία ταχεία αναμετάδοση του τμήματος TCP (Fast Retransmit), πριν λήξει ο χρονομετρητής για το τμήμα αυτό. Για να κατανοηθεί η διαδικασία αυτή, ας θεωρηθεί το παράδειγμα που παρουσιάζεται

στο Σχήμα 2.7. Έστω ότι ο αποστολέας αποστέλλει πέντε τμήματα TCP προς τον παραλήπτη, με το τμήμα με αριθμό ακολουθίας 100 να χάνεται μέσα στο δίκτυο. Ο παραλήπτης λαμβάνει το πρώτο τμήμα με αριθμό ακολουθίας 92 και στέλνει ένα ACK με αριθμό επιβεβαίωσης 100. Η επιβεβαίωση αυτή ενημερώνει τον αποστολέα ότι ο παραλήπτης έχει λάβει σωστά τα πρώτα 99 bytes (συσσωρευτική επιβεβαίωση) του συνολικού ρεύματος δεδομένων και περιμένει το τμήμα με αριθμό ακολουθίας 100. Στη συνέχεια, ο παραλήπτης λαμβάνει τα επόμενα τρία τμήματα σωστά, τα οποία ωστόσο έχουν αριθμό ακολουθίας μεγαλύτερο από τον αναμενόμενο και έχουν ληφθεί εκτός σειράς. Για κάθε νέο τμήμα που λαμβάνει σωστά, ο παραλήπτης εξακολουθεί να αποστέλλει το ίδιο διπλότυπο (duplicate) ACK, ενημερώνοντας τον αποστολέα ότι εξακολουθεί να περιμένει το τμήμα με αριθμό ακολουθίας 100. Τέλος, ο αποστολέας λαμβάνει τα τρία επιπλέον διπλότυπα ACK ως ένδειξη απώλειας αυτού του τμήματος και το αναμεταδίδει πριν από τη λήξη χρόνου, αυξάνοντας έτσι την αποδοτικότητα του πρωτοκόλλου.



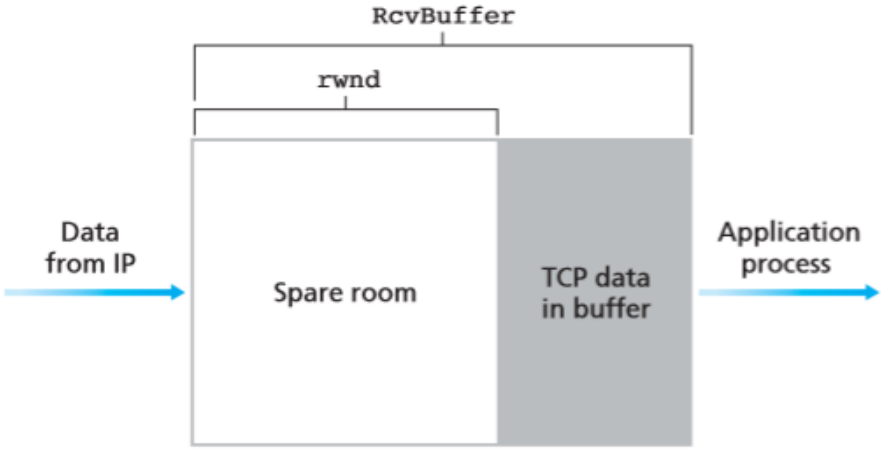
Σχήμα 2.7 *Λήψη τριών διπλότυπων ACK και ταχεία επαναμετάδοση*
[Πηγή: https://wushouyuan.com/posts/d23f7b3e/]

Ο συνδυασμός των δύο ανωτέρω μηχανισμών επιτρέπει στο TCP να ανακάμπτει από όλες τις περιπτώσεις απώλειας πακέτων που προκαλεί η μετάδοση μέσω του μη αξιόπιστου Διαδικτύου.

Συνοψίζοντας, η αρίθμηση των bytes του ρεύματος δεδομένων που πρόκειται να μεταδοθεί με αριθμούς ακολουθίας, η παροχή ανάδρασης από τον τελικό παραλήπτη μέσω επιβεβαιώσεων και οι μηχανισμοί ανίχνευσης απωλειών συνθέτουν την τελική υλοποίηση της επιθυμητής αξιόπιστης υπηρεσίας παράδοσης του πρωτοκόλλου TCP [2].

2.5 Έλεγχος Ροής

Όπως αναλύθηκε στην §2.4, το πρωτόκολλο TCP έχει τη δυνατότητα να διοχετεύει πολλαπλά τμήματα στη σύνδεση TCP, πριν σταματήσει για να περιμένει την άφιξη των επιβεβαιώσεων. Όταν τα τμήματα αυτά φθάσουν στον παραλήπτη στη σωστή σειρά (ενδεχομένως και μετά από αναμεταδόσεις), τοποθετούνται στον ενταμιευτή λήψης που έχει δεσμευτεί για αυτή τη σύνδεση. Ωστόσο, δεν θα μεταβιβαστούν κατευθείαν στο στρώμα εφαρμογής, αφού μπορεί εκείνη την ώρα να εκτελείται κάποια άλλη διεργασία. Τα δεδομένα μπορεί να παραμείνουν στον ενταμιευτή για αρκετή ώρα πριν τελικά τα διαβάσει η εφαρμογή, ιδιαίτερα αν το τερματικό λήψης έχει περιορισμένους υπολογιστικούς πόρους. Στην περίπτωση που ο αποστολέας αποστείλει ταχέως πολλά δεδομένα, ενδέχεται να υπερχειλίσει τον ενταμιευτή λήψης του παραλήπτη. Συνεπώς, ο αποστολέας πρέπει να περιορίζει δυναμικά την ποσότητα δεδομένων που διοχετεύει ταυτόχρονα στη σύνδεση, προσαρμόζοντάς την στο διαθέσιμο αποθηκευτικό χώρο στον παραλήπτη.



Σχήμα 2.8 *Το παράθυρο λήψης (rwnd) ενός παραλήπτη TCP*
[Πηγή: https://wushouyuan.com/posts/d23f7b3e/]

Η υπηρεσία ελέγχου ροής (flow control service) αντιμετωπίζει το πρόβλημα της πιθανής υπερχείλισης ενός ενταμιευτή λήψης. Ουσιαστικά, προσπαθεί να προσαρμόσει το ρυθμό αποστολής δεδομένων από τον αποστολέα στο ρυθμό ανάγνωσης δεδομένων από τον ενταμιευτή λήψης του παραλήπτη. Για το σκοπό αυτό, ορίζεται μία μεταβλητή που ονομάζεται παράθυρο λήψης (receive window) και αντιπροσωπεύει τον αποθηκευτικό χώρο που είναι διαθέσιμος κάθε φορά στην πλευρά του παραλήπτη. Η μεταβλητή *rwnd* αλλάζει δυναμικά με το χρόνο και περιγράφεται γραφικά στο Σχήμα 2.8. Προκειμένου να εξεταστεί ο τρόπος αξιοποίησης του παραθύρου λήψης, ας υποτεθεί ότι ένα αρχείο μεταφέρεται από ένα αποστολέα προς ένα παραλήπτη μέσω μίας σύνδεσης TCP. Όταν ο παραλήπτης αποστέλλει μία επιβεβαίωση ACK για κάποιο τμήμα TCP, ταυτόχρονα τοποθετεί την τρέχουσα τιμή της μεταβλητής *rwnd* στο αντίστοιχο πεδίο της επικεφαλίδας TCP (βλ. §2.2), ενημερώνοντας έτσι τον αποστολέα για το διαθέσιμο αποθηκευτικό χώρο στον ενταμιευτή. Αντίστοιχα, ο αποστολέας παρακολουθεί δύο μεταβλητές: τον αριθμό ακολουθίας του τελευταίου byte που έχει σταλεί (*LastByteSent*) και τον αριθμό ακολουθίας του τελευταίου byte που έχει επιβεβαιωθεί (*LastByteAcked*).

Η διαφορά τους είναι η ποσότητα των δεδομένων που έχουν σταλεί αλλά δεν έχουν ακόμα επιβεβαιωθεί, βρίσκονται δηλαδή σε διαδικασία μετάδοσης (in flight). Διατηρώντας αυτή την ποσότητα μικρότερη από την εκάστοτε τρέχουσα τιμή της μεταβλητής *rwnd* που διαφημίζει ο παραλήπτης, ο αποστολέας εξασφαλίζει ότι δεν θα υπερχειλίσει τον ενταμιευτή λήψης στην άλλη πλευρά της σύνδεσης. Συνεπώς, η αναγκαία συνθήκη που πρέπει να τηρείται για όλη τη διάρκεια της σύνδεσης είναι η εξής:

$$\underbrace{(LastByteSent - LastByteAcked)}_{Bytes\ in\ Flight} \leq rwnd \tag{2.4}$$

Ωστόσο, η μεταβλητή του παραθύρου λήψης *rwnd* αντιμετωπίζει ένα εγγενές ζήτημα. Το πεδίο του παραθύρου λήψης στην επικεφαλίδα TCP έχει μήκος 16 bits. Επομένως, η μέγιστη τιμή της *rwnd* που μπορεί να διαφημιστεί από έναν παραλήπτη σε έναν αποστολέα είναι 65536 bytes. Η τιμή αυτή ήταν ικανοποιητική για τη λειτουργία των πρώτων υλοποιήσεων του TCP, καθώς το μέγεθος των ενταμιευτών λήψης δεν ξεπερνούσε αυτό το άνω όριο. Ωστόσο, τα σημερινά τερματικά συστήματα διαθέτουν ενταμιευτές λήψης με μεγαλύτερο διαθέσιμο αποθηκευτικό χώρο, για τον οποίο οφείλουν να ενημερώνουν τον αποστολέα. Για το λόγο αυτό, χρησιμοποιείται ο παράγοντας κλίμακας παραθύρου (window scaling factor) στο πεδίο επιλογών της επικεφαλίδας TCP, ο οποίος αυξάνει τη μέγιστη τιμή της *rwnd* από 65536 bytes στο 1GB. Ο παράγοντας κλίμακας παραθύρου αναπαριστά των αριθμό των δυαδικών ολισθήσεων προς τα αριστερά που πρέπει να υποστεί το παράθυρο λήψης και παίρνει τιμές στο διάστημα (0, 14). Υπενθυμίζεται ότι η δυαδική ολίσθηση προς τα αριστερά ισοδυναμεί με διπλασιασμό στο δεκαδικό σύστημα. Για παράδειγμα, αν ο παράγοντας κλίμακας έχει την τιμή 3, τότε η νέα μέγιστη τιμή της μεταβλητής *rwnd* υπολογίζεται ως εξής:

$$rwnd_{new} = rwnd \cdot 2^{(scaling\ factor)} = 65536 \cdot 2^3 = 512kB \tag{2.5}$$

Η τιμή των 512kB είναι αντιπροσωπευτική του μεγέθους ενός σημερινού ενταμιευτή λήψης μίας σύνδεσης TCP σε κινητές συσκευές, ενώ σε επιτραπέζιους υπολογιστές το μέγεθος αυτό μπορεί να είναι της τάξης των Megabytes. Συνεπώς, με τη χρήση του παράγοντα κλίμακας η τιμή της μεταβλητής *rwnd* δεν αποτελεί πλέον ανασταλτικό παράγοντα στην επίδοση του πρωτοκόλλου [7].

Τέλος, υπάρχει ένα τεχνικό πρόβλημα σε αυτόν το συνολικό σχεδιασμό. Έστω ότι κάποια στιγμή ο ενταμιευτής λήψης γεμίζει (*rwnd = 0*) και ο παραλήπτης διαφημίζει αυτή την τιμή στον αποστολέα, ο οποίος σταματά να αποστέλλει δεδομένα. Ας υποτεθεί, επιπλέον, ότι ο παραλήπτης δεν έχει σε εκκρεμότητα κάποια επιβεβαίωση ACK που πρέπει να στείλει στον αποστολέα. Καθώς, λοιπόν, η εφαρμογή λήψης καταναλώνει δεδομένα από τον ενταμιευτή και απελευθερώνεται χώρος (άρα αυξάνεται η *rwnd*), το TCP του παραλήπτη δεν μπορεί να διαφημίσει τις νέες τιμές αφού δεν εκκρεμεί καμία επιβεβαίωση ACK για μετάδοση. Ως αποτέλεσμα, ο αποστολέας δεν ενημερώνεται ποτέ για αυτή την αλλαγή και δεν επανεκκινεί την αποστολή δεδομένων. Για τη λύση του προβλήματος αυτού, οι προδιαγραφές του TCP απαιτούν από τον αποστολέα να συνεχίσει να αποστέλλει τμήματα του ενός byte προς τον παραλήπτη, ακόμα και όταν το παράθυρο λήψης είναι μηδενικό. Έτσι, όταν αρχίζει να αδειάζει ο ενταμιευτής λήψης, οι επιβεβαιώσεις ACK για αυτά τα μικρά τμήματα θα περιέχουν μία μη μηδενική τιμή της μεταβλητής *rwnd* και ο αποστολέας θα μπορέσει να συνεχίσει την μετάδοση δεδομένων.

2.6 Έλεγχος Συμφόρησης

Στις προηγούμενες παραγράφους παρουσιάστηκαν οι αρχές λειτουργίας και οι μηχανισμοί που διαθέτει το TCP για την αντιμετώπιση των απωλειών πακέτων. Πρακτικά, αυτές συμβαίνουν όταν το δίκτυο βρίσκεται σε κατάσταση συμφόρησης και οι ενταμιευτές στους ενδιάμεσους δρομολογητές υπερχειλίζουν λόγω της υπερβολικής δικτυακής κίνησης. Κατ' αντιστοιχία με περιπτώσεις της καθημερινής ζωής (π.χ. δρόμος αυξημένης κυκλοφορίας), αιτία της συμφόρησης σε ένα δίκτυο αποτελεί το ότι πολλές συνδέσεις αποστέλλουν δεδομένα με υψηλό ρυθμό, έχοντας, όμως, στη διάθεσή τους πεπερασμένους κοινούς πόρους όπως εύρος ζώνης ή ενταμιευτές.

Η προσέγγιση του TCP για τη λύση αυτού του προβλήματος είναι η προσπάθεια κάθε αποστολέα να περιορίζει το ρυθμό αποστολής δεδομένων στη σύνδεση, με βάση το επίπεδο συμφόρησης που γίνεται αντιληπτό από την πλευρά του. Όταν γίνεται αντιληπτή χαμηλή συμφόρηση ο αποστολέας αυξάνει το ρυθμό αποστολής, ενώ, σε αντίθετη περίπτωση, ο ρυθμός πρέπει να μειωθεί για να αποφευχθεί ενδεχόμενη κατάρρευση του δικτύου. Το σύνολο των μηχανισμών και αλγορίθμων που χρησιμοποιεί το TCP για την επίτευξη του ανωτέρω μηχανισμού ονομάζεται συγκεντρωτικά έλεγχος συμφόρησης (congestion control) και αποτελεί ένα από τα σημαντικότερα ζητήματα της δικτύωσης.

2.6.1 Παράθυρο Συμφόρησης

Ο τρόπος με τον οποίο ένας αποστολέας TCP ελέγχει το ρυθμό μετάδοσης είναι ο περιορισμός του αριθμού των τμημάτων TCP που έχουν σταλεί αλλά δεν έχουν επιβεβαιωθεί. Έχει ήδη παρουσιαστεί η μεταβλητή του παραθύρου λήψης *rwnd* και ο τρόπος αξιοποίησής της κατά τον έλεγχο ροής του TCP. Αντίστοιχα, ο έλεγχος συμφόρησης στην πλευρά του αποστολέα χρησιμοποιεί μία νέα μεταβλητή, που ονομάζεται παράθυρο συμφόρησης (congestion window) και συμβολίζεται ως *cwnd*. Το παράθυρο συμφόρησης ορίζει τη μέγιστη ποσότητα μη επιβεβαιωθέντων δεδομένων που μπορεί να διατηρεί ο αποστολέας μέσα στη σύνδεση. Συνεπώς, η αναγκαία συνθήκη, που παρουσιάστηκε στην §2.5 (βλ. Εξίσωση 2.4) και οφείλει να ισχύει για όλη τη διάρκεια της σύνδεσης, τροποποιείται ως εξής:

$$\underbrace{(LastByteSent - LastByteAcked)}_{Bytes\ in\ Flight} \leq \min\{rwnd, cwnd\} \tag{2.6}$$

Το παράθυρο συμφόρησης καθορίζει έμμεσα το ρυθμό αποστολής. Έστω ότι σε μία σύνδεση δεν υπάρχουν απώλειες και ότι τη χρονική στιγμή t=0 ο αποστολέας αποστέλλει *cwnd* bytes προς τον παραλήπτη, πριν γίνει παύση για αναμονή των επιβεβαιώσεων ACK. Λίγο μετά τη χρονική στιγμή t=RTT, οι επιβεβαιώσεις αυτές λαμβάνονται από τον αποστολέα που αποστέλλει τα επόμενα δεδομένα, με τη διαδικασία να επαναλαμβάνεται. Επομένως, μία θεωρητική εκτίμηση του ρυθμού αποστολής είναι:

$$R = \frac{cwnd}{RTT} \cdot \frac{bytes}{seconds} \tag{2.7}$$

Επομένως, είναι εφικτή η μεταβολή του R μέσω κατάλληλης προσαρμογής της μεταβλητής *cwnd*.

2.6.2 Βασικές Αρχές

Από τη στιγμή όπου η κατάσταση ενός δικτύου αλλάζει δυναμικά με το χρόνο, είναι προφανές ότι η τιμή της μεταβλητής *cwnd* δεν θα είναι σταθερή. Κάθε αποστολέας TCP οφείλει να προσαρμόζει την τιμή της (κατά συνέπεια και το ρυθμό μετάδοσής του), έτσι ώστε συλλογικά για τους αποστολείς TCP να ικανοποιούνται δύο συνθήκες. Πρώτον, οι αποστολείς να μην αποστέλλουν με πολύ υψηλό ρυθμό, καθώς το δίκτυο ενδέχεται να οδηγηθεί σε κατάρρευση. Δεύτερον, οι αποστολείς να μην είναι πολύ συντηρητικοί με το ρυθμό αποστολής δεδομένων, επειδή τότε θα προκύψει υποχρησιμοποίηση του διαθέσιμου εύρους ζώνης. Συνεπώς, κάθε αποστολέας TCP συντονίζει το ρυθμό αποστολής του με κριτήριο αυτές τις συνθήκες και ακολουθώντας τις εξής βασικές αρχές:

- Ένα περιστατικό απώλειας τμήματος TCP υποδηλώνει την ύπαρξη συμφόρησης στο δίκτυο. Οι μηχανισμοί που χρησιμοποιούνται για την ανίχνευση απωλειών παρουσιάστηκαν στην §2.4.3. Ύστερα από την αναμετάδοση του χαμένου τμήματος, ο έλεγχος συμφόρησης οφείλει να μειώσει το μέγεθος του παραθύρου συμφόρησης, προς αντιμετώπιση του περιστατικού απώλειας.

- Η λήψη ενός τμήματος επιβεβαίωσης ACK υπονοεί ότι τα τμήματα του αποστολέα παραδίδονται από το δίκτυο επιτυχώς στον παραλήπτη. Συνεπώς, η λήψη μιας επιβεβαίωσης ACK για ένα τμήμα που δεν είχε προηγουμένως επιβεβαιωθεί, προκαλεί την αύξηση του παραθύρου συμφόρησης.

- Με βάση την απόκριση του πρωτοκόλλου στα ανωτέρω δύο χαρακτηριστικά λειτουργίας, η στρατηγική του TCP για τον έλεγχο της συμφόρησης περιλαμβάνει δυναμική μεταβολή του παραθύρου συμφόρησης. Κατά την εκκίνηση μίας σύνδεσης TCP, και ενόσω λαμβάνονται επιβεβαιώσεις ACK, το παράθυρο συμφόρησης αυξάνεται. Όταν συμβεί μία απώλεια τμήματος TCP, το παράθυρο συμφόρησης μειώνεται για να αποφευχθεί περαιτέρω συμφόρηση. Κατόπιν, ο αποστολέας αυξάνει πάλι σταδιακά το παράθυρο συμφόρησης μέχρι να συμβεί εκ νέου μία απώλεια τμήματος. Η ανωτέρω διαδικασία επαναλαμβάνεται μέχρις ότου να τερματιστεί η σύνδεση TCP.

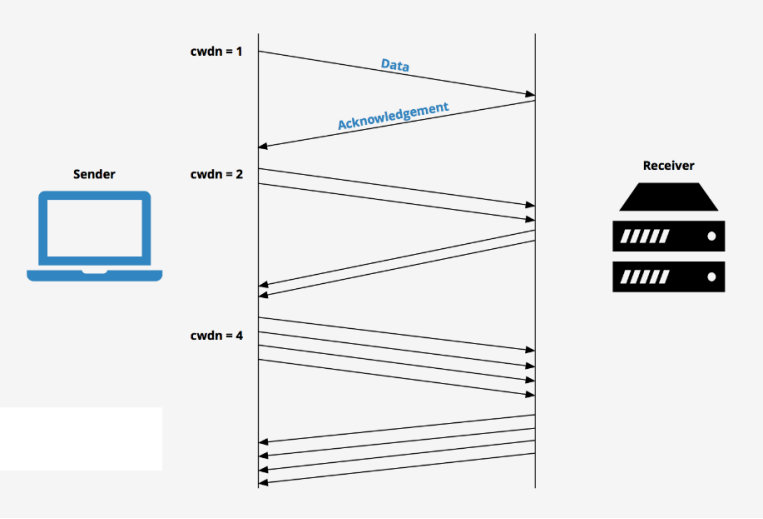
Πρέπει, τέλος, να τονιστεί ότι κάθε αποστολέας TCP πραγματοποιεί το δικό του έλεγχο συμφόρησης, ανεξάρτητα από τους άλλους αποστολείς. Στο [8] αποδείχθηκε ότι ο έλεγχος συμφόρησης του TCP λειτουργεί ως κατανεμημένος αλγόριθμος ασύγχρονης βελτιστοποίησης. Από τότε ακολούθησε η ανάπτυξη μιας πλούσιας μαθηματικής θεωρίας για τον έλεγχο συμφόρησης [9].

2.6.3 Ο Αλγόριθμος Ελέγχου Συμφόρησης

Χρησιμοποιώντας τις βασικές αρχές που αναφέρθηκαν στην §2.6.2, το TCP υλοποιεί ένα αλγόριθμο ελέγχου συμφόρησης (congestion control algorithm), ο οποίος προτυποποιήθηκε στο RFC 5681 [10]. Ο αλγόριθμος ορίζει τρεις διακριτές καταστάσεις, στις οποίες μπορεί να μεταπέσει ένας αποστολέας TCP: αργή εκκίνηση (slow start), αποφυγή συμφόρησης (congestion avoidance) και ταχεία ανάκαμψη (fast recovery). Οι μεταβάσεις μεταξύ των καταστάσεων ενεργοποιούνται όταν συμβούν συγκεκριμένες

συνθήκες ή περιστατικά, και επιβάλλουν επίσης αλλαγές στην τιμή της μεταβλητής *cwnd*. Η περιγραφή κάθε κατάστασης ακολουθεί στη συνέχεια.

- **Αργή Εκκίνηση**: Κατά την εκκίνηση μίας σύνδεσης TCP, ο αποστολέας δεν γνωρίζει την κατάσταση κίνησης του δικτύου. Για να ανιχνεύσει ταχέως το διαθέσιμο εύρος ζώνης, αλλά χωρίς να αυξήσει το επίπεδο φόρτου στο δίκτυο, ο αποστολέας αρχίζει τη μετάδοση δεδομένων με μικρό αρχικό ρυθμό, τον οποίο όμως αυξάνει εκθετικά. Η αρχική τιμή της *cwnd* είναι ένα μικρό πολλαπλάσιο του μεγέθους MSS (συνήθως 1,2,4 ή 10 MSS) και αυξάνεται κατά 1 MSS, κάθε φορά που λαμβάνεται μία επιβεβαίωση ACK. Στο Σχήμα 2.9 απεικονίζεται η φάση της αργής εκκίνησης, για αρχική τιμή της μεταβλητής *cwnd=1*. Με τη λήψη της πρώτης επιβεβαίωσης, το παράθυρο συμφόρησης διπλασιάζεται στην τιμή 2 MSS. Αντίστοιχα, η λήψη των δύο νέων επιβεβαιώσεων διπλασιάζει εκ νέου το παράθυρο συμφόρησης στην τιμή των 4 MSS. Με αυτή τη διαδικασία, αν και το παράθυρο συμφόρησης *cwnd* έχει μικρή αρχική τιμή, ωστόσο αυξάνεται εκθετικά με τον διπλασιασμό του ύστερα από κάθε RTT. Το TCP εξέρχεται από την κατάσταση της αργής εκκίνησης όταν προκύψει κάποια από τις εξής τρεις δυνατές καταστάσεις. Πρώτον, σε περίπτωση απώλειας τμήματος που ανιχνεύεται ύστερα από λήξη του χρονομετρητή *TimeoutInterval*, ο αποστολέας θέτει *cwnd=1* και αρχίζει εκ νέου το στάδιο της αργής εκκίνησης. Επιπλέον, θέτει την τιμή μίας δεύτερης μεταβλητής, που ονομάζεται κατώφλι αργής εκκίνησης (slow start threshold) και συμβολίζεται ως *ssthresh*, στο μισό της τιμής του παραθύρου συμφόρησης πριν το περιστατικό απώλειας (δηλαδή *cwnd/2*). Δεύτερον, όταν η τιμή της *cwnd* υπερβεί το κατώφλι *ssthresh*, η εκθετική αύξηση του παραθύρου συμφόρησης μετατρέπεται σε γραμμική και το TCP μεταπίπτει στο στάδιο της αποφυγής συμφόρησης. Τέλος, σε περίπτωση ανίχνευσης τριών διπλότυπων επιβεβαιώσεων ACK, το TCP εκτελεί ταχεία αναμετάδοση του χαμένου τμήματος (βλ. §2.4.3) και εισέρχεται στην κατάσταση της ταχείας ανάκαμψης.

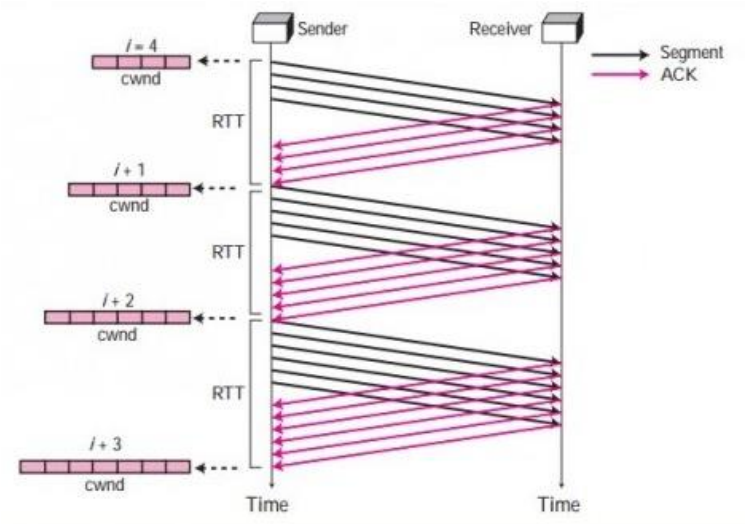


Σχήμα 2.9 *Η αργή εκκίνηση του TCP*
[Πηγή: https://www.keycdn.com/support/tcp-slow-start]

- **Αποφυγή Συμφόρησης**: Η είσοδος στην κατάσταση της αποφυγής συμφόρησης γίνεται όταν το παράθυρο συμφόρησης υπερβεί την τιμή *ssthresh*. Υπενθυμίζεται ότι η τιμή της *ssthresh* γίνεται ίση με *cwnd*/2, κάθε φορά που συμβαίνει λήξη χρόνου. Επομένως, η τιμή της *cwnd* όταν εκκινεί το στάδιο αποφυγής συμφόρησης είναι το μισό της αντίστοιχης τιμής της, για την οποία υπήρξε απώλεια πακέτου. Προκειμένου ο αποστολέας να αποφύγει τη συμφόρηση πριν αυτή συμβεί, η εκθετική αύξηση της *cwnd* μετατρέπεται σε γραμμική και αυξάνεται κατά 1 MSS κάθε RTT (αντί για διπλασιασμό). Η προσέγγιση για την επίτευξη αυτής της απαίτησης απεικονίζεται στο Σχήμα 2.10. Ας υποτεθεί ότι, κάποια χρονική στιγμή όπου ο αποστολέας TCP βρίσκεται στη φάση αποφυγής συμφόρησης, το παράθυρο συμφόρησης έχει μέγεθος 4 MSS. Τότε, η άφιξη επιβεβαίωσης ACK για ένα τμήμα εντός του παραθύρου προκαλεί αύξηση του παραθύρου συμφόρησης κατά MSS/4. Με αυτόν τον τρόπο, όταν ληφθούν και οι τέσσερεις επιβεβαιώσεις ACK, το παράθυρο συμφόρησης θα έχει αυξηθεί κατά 1 MSS σε χρονικό διάστημα RTT. Το TCP εξέρχεται από τη φάση αποφυγής συμφόρησης όταν συμβεί κάποια απώλεια πακέτου. Αν το συμβάν απώλειας αντιστοιχεί σε λήξη χρόνου, το TCP μειώνει απότομα το ρυθμό αποστολής του, θέτοντας την τιμή της *ssthresh* σε *cwnd/2* και το παράθυρο συμφόρησης *cwnd* στην τιμή 1 MSS. Κατόπιν, το TCP μεταβαίνει στη φάση της αργής εκκίνησης. Σε περίπτωση όπου το περιστατικό απώλειας ανιχνευθεί από την άφιξη τριών διπλότυπων ACK, το TCP αντιδρά λιγότερο δραστικά ως προς τη μείωση του ρυθμού μετάδοσης, μεταβάλλοντας τις μεταβλητές ως εξής:

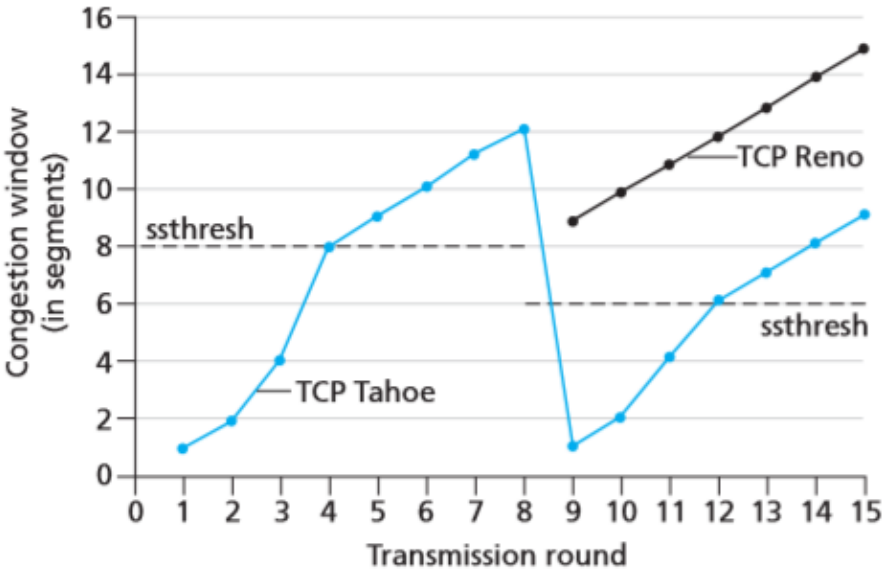
$$\begin{cases} ssthresh = \dfrac{cwnd}{2} \\ cwnd' = \dfrac{cwnd}{2} + 3 \end{cases} \tag{2.8}$$

Στη συνέχεια, το χαμένο τμήμα αναμεταδίδεται και ακολουθεί η φάση ταχείας ανάκαμψης.



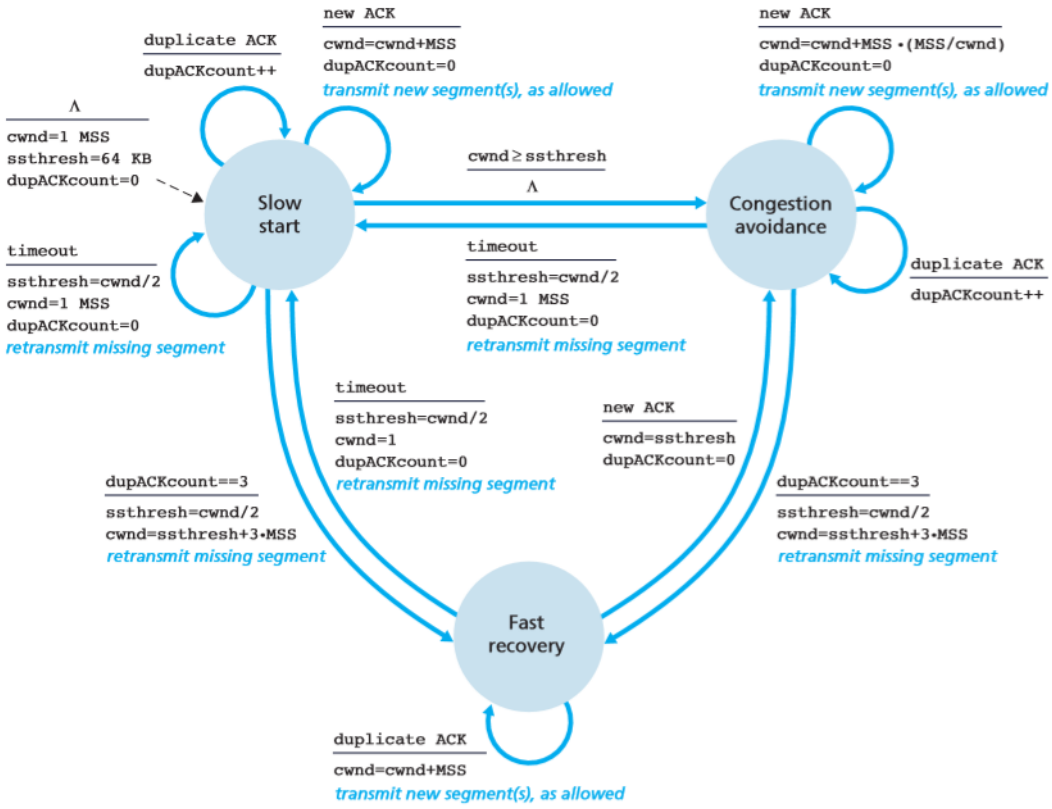
Σχήμα 2.10 *Η αποφυγή συμφόρησης του TCP*
[Πηγή: https://gateoverflow.in/119354/congestion-avoidance-in-tcp]

- **Ταχεία Ανάκαμψη**: Το TCP εισέρχεται στο στάδιο της ταχείας ανάκαμψης ύστερα από την αναμετάδοση ενός χαμένου τμήματος για το οποίο έχει λάβει τρία διπλότυπα ACK. Ενόσω ο αποστολέας εξακολουθεί να λαμβάνει διπλότυπα ACK, για κάθε ένα αυξάνει την τιμή της μεταβλητής *cwnd* κατά 1 MSS και μεταδίδει ένα νέο τμήμα, εφόσον υπάρχει διαθέσιμο. Όταν τελικά ληφθεί μία επιβεβαίωση ACK για το χαμένο τμήμα, το TCP μειώνει τη μεταβλητή *cwnd* στην τιμή *ssthresh* (όπως αυτή ορίστηκε στο τέλος της αποφυγής συμφόρησης), μεταπίπτοντας ύστερα στην κατάσταση αποφυγής συμφόρησης. Σε περίπτωση όπου συμβεί λήξη χρόνου, το TCP μεταβαίνει στην αργή εκκίνηση, εκτελώντας τις ίδιες ενέργειες που περιγράφηκαν στις προηγούμενες καταστάσεις. Αξίζει να σημειωθεί ότι οι πρώτες εκδόσεις του TCP, όπως για παράδειγμα το TCP Tahoe, δεν χρησιμοποιούσαν την ταχεία ανάκαμψη. Αντιθέτως, το παράθυρο συμφόρησης μειωνόταν στην τιμή 1 MSS ύστερα από κάποιο συμβάν απώλειας και το TCP εισερχόταν στο στάδιο της αργής εκκίνησης. Ωστόσο, η προσέγγιση αυτή οδηγούσε σε μη αποδοτική αξιοποίηση του εύρους ζώνης, με αποτέλεσμα νεότερες εκδόσεις, όπως το TCP Reno, να περιλαμβάνουν την ταχεία ανάκαμψη. Στο Σχήμα 2.11 απεικονίζεται η συγκεκριμένη διαφοροποίηση. Αρχικά, το παράθυρο συμφόρησης αυξάνεται εκθετικά και ύστερα γραμμικά, μόλις υπερβεί το κατώφλι *ssthresh*. Το στάδιο αυτό είναι κοινό και για τις δύο εκδόσεις TCP. Μετά τον 8ο γύρο μετάδοσης, λαμβάνει χώρα ένα περιστατικό λήψης τριών διπλότυπων ACK, όταν το παράθυρο συμφόρησης έχει την τιμή 12 MSS. Το TCP Reno μειώνει το παράθυρο συμφόρησης στην τιμή 9 MSS και κατόπιν το αυξάνει γραμμικά παραμένοντας στη φάση της ταχείας ανάκαμψης. Το TCP Tahoe το μειώνει στην τιμή 1 MSS και εισέρχεται στη φάση της αργής εκκίνησης. Επομένως, ο αλγόριθμος του TCP Reno αντιμετωπίζει την απώλεια τμήματος με ταυτόχρονη διατήρηση υψηλότερου ρυθμού μετάδοσης, συγκριτικά με το TCP Tahoe.



Σχήμα 2.11 *Διαφορές στην εξέλιξη του cwnd μεταξύ TCP Tahoe και TCP Reno*
[Πηγή: https://wushouyuan.com/posts/d23f7b3e/]

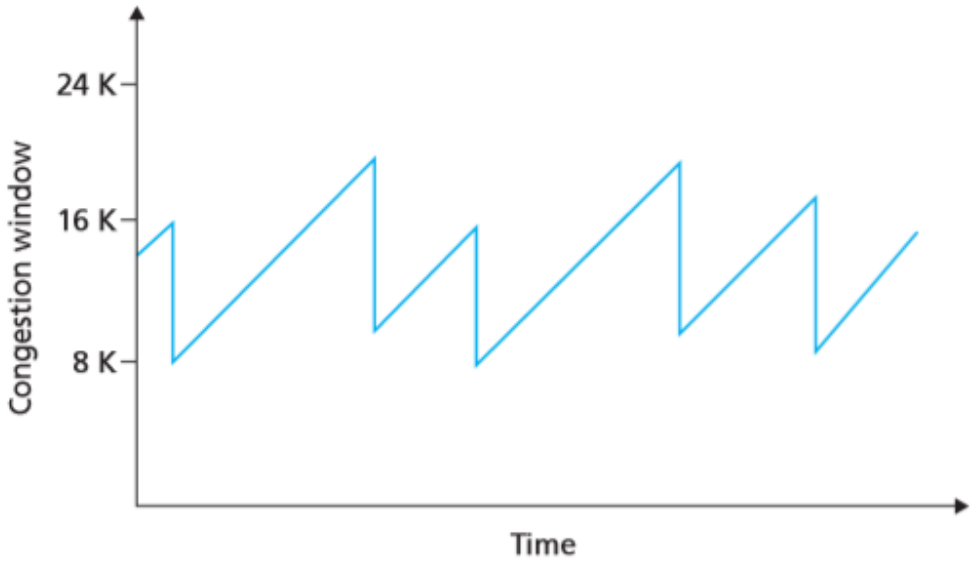
Στο Σχήμα 2.12 απεικονίζονται συγκεντρωτικά οι καταστάσεις του αλγορίθμου συμφόρησης του TCP, σε συνδυασμό με τις συνθήκες μετάβασης. Συνοψίζοντας την προηγούμενη ανάλυση, είναι χρήσιμο να παρουσιαστεί μία μακροσκοπική θεώρηση του ελέγχου συμφόρησης. Έστω ότι μία σύνδεση TCP έχει ολοκληρώσει το στάδιο της αργής εκκίνησης και ότι οι απώλειες ανιχνεύονται από διπλότυπα ACK, καθώς οι λήξεις χρόνου είναι σπάνιες. Σε βάθος χρόνου, ο έλεγχος συμφόρησης θα αποτελείται από συνεχείς εναλλαγές μεταξύ γραμμικής αύξησης της μεταβλητής *cwnd* και υποδιπλασιασμού της, όταν συμβεί μία απώλεια τμήματος. Για το λόγο αυτό, ο αλγόριθμος συμφόρησης του TCP αναφέρεται ως αλγόριθμος «προσθετικής αύξησης, πολλαπλασιαστικής μείωσης» (additive increase & multiplicative decrease - AIMD) και οδηγεί στην επονομαζόμενη «πριονωτή» συμπεριφορά, όπως απεικονίζεται στο Σχήμα 2.13.



Σχήμα 2.12 *Διάγραμμα καταστάσεων του αλγορίθμου ελέγχου συμφόρησης TCP*
[Πηγή: https://wushouyuan.com/posts/d23f7b3e/]

2.6.4 Δικαιοσύνη

Όπως έχει ήδη αναφερθεί, ένα δίκτυο χρησιμοποιείται από πολλούς χρήστες ταυτόχρονα. Για λόγους απλότητας, ας υποτεθεί ότι οι χρήστες παράγουν μόνο κίνηση TCP. Έστω, λοιπόν, ότι N συνδέσεις TCP, εκάστη των οποίων χαρακτηρίζεται από διαφορετικό χρόνο RTT, χρησιμοποιούν ταυτόχρονα την ίδια ζεύξη, η οποία χαρακτηρίζεται από χωρητικότητα (capacity) R. Ένας αλγόριθμος συμφόρησης είναι «δίκαιος», όταν ο μέσος ρυθμός μετάδοσης κάθε σύνδεσης TCP είναι περίπου R/N. Έχει αποδειχθεί ότι η χρήση της τεχνικής AIMD συγκλίνει σε ισορροπία, ώστε ανταγωνιστικές συνδέσεις TCP να καταλαμβάνουν ίσα μερίδια της χωρητικότητας του δικτύου [11].



Σχήμα 2.13 *Προσθετική αύξηση-Πολλαπλασιαστική μείωση του ελέγχου συμφόρησης TCP*
[Πηγή: https://wushouyuan.com/posts/d23f7b3e/]

2.6.5 Δημοφιλείς Αλγόριθμοι Συμφόρησης

Ο τρόπος με τον οποίο το πρωτόκολλο TCP ανταποκρίνεται σε φαινόμενα συμφόρησης έχει αποτελέσει σημαντικό πεδίο έρευνας για αρκετά χρόνια. Ωστόσο, η προδιαγραφή στο [10] επιτρέπει την υλοποίηση παραλλαγών του αλγορίθμου, που προσαρμόζουν την επίδοση του TCP στους διαφορετικούς τύπους δικτύων που έχουν αναπτυχθεί (π.χ. δίκτυα υψηλής χωρητικότητας, ασύρματα δίκτυα). Η ανάλυση στην §2.6.3 βασίστηκε στον αλγόριθμο TCP Reno, ο οποίος αποτελεί έναν από τους πλέον διαδεδομένους [12]. Στη συνέχεια, παρουσιάζονται ορισμένες δημοφιλείς παραλλαγές του ελέγχου συμφόρησης.

Ο αλγόριθμος TCP New Reno βελτιώνει τις αναμεταδόσεις πακέτων κατά το στάδιο της ταχείας ανάκαμψης [13]. Ο αρχικός αλγόριθμος TCP Reno εισέρχεται στη φάση της ταχείας ανάκαμψης όταν λάβει τρία διπλότυπα ACK για ένα τμήμα, ενώ εξέρχεται από αυτή όταν το χαμένο τμήμα επιβεβαιωθεί, με ταυτόχρονο υποδιπλασιασμό του παραθύρου συμφόρησης. Ωστόσο, σε περίπτωση όπου υπάρχουν επιπλέον απώλειες για τμήματα που έχουν αρχικά μεταδοθεί μέσα στο ίδιο παράθυρο, τότε ο αλγόριθμος θα εισέλθει πάλι στο στάδιο της ταχείας ανάκαμψης. Αυτό έχει ως αποτέλεσμα διαδοχικούς υποδιπλασιασμούς του παραθύρου συμφόρησης, γεγονός που μειώνει την επίδοση του πρωτοκόλλου. Αντίθετα, το TCP New Reno αναμεταδίδει όλα τα τμήματα που έχουν χαθεί και είχαν μεταδοθεί μέσα στο ίδιο παράθυρο συμφόρησης. Για το σκοπό αυτό, ο αλγόριθμος καταγράφει τον υψηλότερο αριθμό ακολουθίας τμήματος που έχει σταλεί αλλά δεν έχει επιβεβαιωθεί (έστω *LastByteSent)*, πριν εισέλθει στην ταχεία ανάκαμψη. Ύστερα, για κάθε ληφθέν ACK που δεν είναι διπλότυπο, συγκρίνει τον αριθμό επιβεβαίωσής του με τον αριθμό ακολουθίας *LastByteSent*. Αν είναι μικρότερος, τότε θεωρεί ότι το τμήμα με αριθμό ακολουθίας ίσο με τον αριθμό επιβεβαίωσης έχει επίσης χαθεί και το αναμεταδίδει. Αν είναι μεγαλύτερος, τότε όλα τα τμήματα που είχαν ήδη μεταδοθεί πριν ανιχνευθεί η πρώτη απώλεια θεωρείται ότι έχουν παραδοθεί σωστά στον παραλήπτη. Κατόπιν, ο αλγόριθμος New Reno υποδιπλασιάζει το παράθυρο

συμφόρησης και εισέρχεται στο στάδιο της αποφυγής συμφόρησης. Συνολικά, το TCP New Reno πετυχαίνει σημαντικά καλύτερη απόδοση σε δίκτυα με υψηλό ρυθμό απωλειών, συγκριτικά με το TCP Reno [14].

Ο αλγόριθμος TCP Vegas βασίζεται στη συχνή παρατήρηση των χρόνων RTT των τμημάτων της σύνδεσης TCP, προκειμένου να ανιχνεύσει πρώιμα μία επερχόμενη συμφόρηση στο δίκτυο [15]. Η προσέγγιση αυτή διαφέρει από άλλους αλγορίθμους, οι οποίοι ανιχνεύουν τη συμφόρηση στο δίκτυο μόνο αφού προκληθεί κάποια απώλεια τμήματος. Η υλοποίηση βασίζεται σημαντικά στη σωστή μέτρηση της μεταβλητής *BaseRTT*, που αντιπροσωπεύει την ελάχιστη τιμή του χρόνου RTT κατά τη διάρκεια της σύνδεσης. Στη συνέχεια, ο αλγόριθμος υπολογίζει τη διαφορά *Delta* μεταξύ του αναμενόμενου (expected) και του μετρήσιμου (actual) ρυθμού μετάδοσης:

$$\begin{cases} actual = \dfrac{cwnd}{RTT} \\ expected = \dfrac{cwnd}{BaseRTT} \end{cases} \tag{2.9}$$

Προς αποφυγή συμφόρησης, ο αλγόριθμος Vegas μεταβάλλει γραμμικά την τιμή της μεταβλητής *cwnd*, διασφαλίζοντας ότι η διαφορά *Delta* θα ανήκει στο διάστημα [α, β]. Οι μεταβλητές α και β αποτελούν δύο κατώφλια, με προεπιλεγμένες τις τιμές 2 και 4 αντίστοιχα. Η συνθήκη γραμμικής αύξησης/μείωσης είναι η εξής:

$$cwnd = \begin{cases} increase\ by\ 1\ MSS, & if\ Delta < a \\ decrease\ by\ 1\ MSS, & if\ Delta > \beta \\ no\ change, & if\ Delta \in [a, \beta] \end{cases} \tag{2.10}$$

Εξαιτίας της διαφορετικής φύσης του αλγορίθμου Vegas ως προς την ανίχνευση της συμφόρησης, μελέτες έχουν δείξει ότι μία ροή TCP Vegas εμφανίζει προβλήματα δικαιοσύνης, όταν συνυπάρχει στην ίδια ζεύξη με ροές TCP Reno [16].

Η έκδοση TCP CUBIC βελτιστοποιεί τον έλεγχο συμφόρησης για δίκτυα υψηλού εύρους ζώνης και μεγάλης καθυστέρησης (Long Fat Networks - LFN), ενώ αποτελεί την προεπιλεγμένη έκδοση για το λειτουργικό σύστημα Linux [17]. Ο αλγόριθμος CUBIC χρησιμοποιεί μία κυβική συνάρτηση για την αύξηση του παραθύρου συμφόρησης κατά το στάδιο της αποφυγής συμφόρησης. Έστω ότι το παράθυρο συμφόρησης έχει την τιμή w_{max}, ακριβώς πριν την ανίχνευση μίας απώλειας τμήματος και, επομένως, πριν την ανάγκη μείωσης του μεγέθους του. Ο αλγόριθμος CUBIC χρησιμοποιεί το κοίλο τμήμα της κυβικής συνάρτησης για να επαναφέρει ταχέως την τιμή της μεταβλητής *cwnd* στη τιμή που είχε πριν την απώλεια τμήματος. Στη συνέχεια, το κυρτό τμήμα χρησιμοποιείται για σταδιακή αύξηση του παραθύρου συμφόρησης, στην αρχή αργά και ύστερα ταχύτερα, στην προσπάθεια του TCP να διερευνήσει το δίκτυο με στόχο να αξιοποιήσει περισσότερο εύρος ζώνης. Θέτοντας τον χρόνο που έχει παρέλθει από το τελευταίο συμβάν απώλειας ως *T*, η κυβική συνάρτηση είναι η εξής:

$$\begin{cases} cwnd = C \cdot (T - K)^3 + w_{max} \\ where\ K = \sqrt[3]{\dfrac{w_{max}(1-\beta)}{C}} \end{cases} \quad (2.11)$$

Η μεταβλητή β αποτελεί τον παράγοντα πολλαπλασιαστικής μείωσης, ενώ η μεταβλητή C είναι μία σταθερά κανονικοποίησης. Οι προτεινόμενες τιμές είναι 0.7 και 0.4, αντίστοιχα. Συγκριτικά με άλλες υλοποιήσεις, η βασική διαφορά του αλγορίθμου TCP CUBIC είναι το ότι δεν βασίζεται στην άφιξη επιβεβαιώσεων ACK για την αύξηση του παραθύρου συμφόρησης. Αντιθέτως, η αύξηση του παραθύρου συμφόρησης εξαρτάται μόνο από τη χρονική στιγμή που συνέβη το πλέον πρόσφατο συμβάν απώλειας. Στις περισσότερες εκδόσεις του TCP, οι ροές με μικρότερους χρόνους RTT καταλαμβάνουν μεγαλύτερο εύρος ζώνης συγκρινόμενες με άλλες ροές, καθώς οι επιβεβαιώσεις ACK λαμβάνονται με υψηλότερο ρυθμό και άρα το παράθυρο συμφόρησης αυξάνεται ταχύτερα. Ο αλγόριθμος CUBIC επιτυγχάνει υψηλότερο επίπεδο δικαιοσύνης, καθώς η αύξηση της μεταβλητής *cwnd* είναι ανεξάρτητη του χρόνου RTT.

Για μία εκτενέστερη παρουσίαση των αλγορίθμων ελέγχου συμφόρησης που χρησιμοποιούνται σε σύγχρονες εκδόσεις του πρωτοκόλλου TCP, ο ενδιαφερόμενος αναγνώστης παραπέμπεται στο [18].

2.7 Ασφάλεια Συνδέσεων TCP

Η ψηφιακή επανάσταση του Διαδικτύου έφερε στο προσκήνιο πολλές Web εφαρμογές, όπως το ηλεκτρονικό εμπόριο ή τις ηλεκτρονικές τραπεζικές συναλλαγές, που σήμερα χρησιμοποιούνται ευρύτατα. Ωστόσο, οι εφαρμογές αυτές απαιτούν την ανταλλαγή ευαίσθητων ή προσωπικών πληροφοριών, οι οποίες αναγκαστικά μεταδίδονται μέσω του «ανασφαλούς» Internet. Αυτό οδήγησε στην ανάπτυξη του πεδίου της Ασφάλειας Δικτύων, προκειμένου να εξασφαλιστεί ότι η επικοινωνία μεταξύ δύο οποιονδήποτε οντοτήτων στο Διαδίκτυο θα διαθέτει τις εξής ιδιότητες:

- **Εμπιστευτικότητα**: Το περιεχόμενο ενός μηνύματος οφείλει να είναι προσβάσιμο αποκλειστικά από τον αποστολέα και τον παραλήπτη. Για το λόγο αυτό, τα μηνύματα μεταδίδονται κρυπτογραφημένα, ώστε να είναι προστατευμένα από τρίτα μέρη.

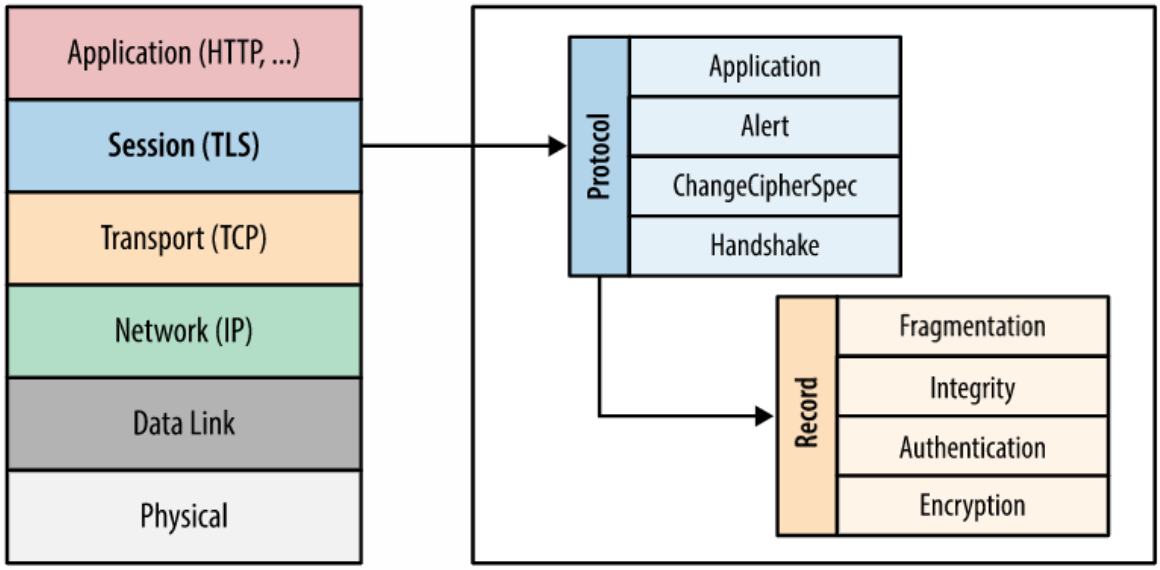
- **Ακεραιότητα**: Ο παραλήπτης ενός μηνύματος οφείλει να ελέγχει ότι το περιεχόμενο που λαμβάνει δεν έχει αλλοιωθεί ή τροποποιηθεί, δηλαδή ότι το ληφθέν μήνυμα είναι πράγματι αυτό που είχε αρχικά σταλεί από τον αποστολέα.

- **Ταυτοποίηση**: Κάθε οντότητα που χρησιμοποιεί το Διαδίκτυο πρέπει να είναι σε θέση να επιβεβαιώνει την ταυτότητα του άλλου άκρου που συμμετέχει στην επικοινωνία, ώστε να διασφαλίζεται ότι ο συνομιλητής είναι πράγματι αυτός που ισχυρίζεται.

Οι πλήρεις τεχνικές και μηχανισμοί που έχουν αναπτυχθεί για την υλοποίηση των ανωτέρω δεν ανήκουν στο αντικείμενο της παρούσας εργασίας. Ωστόσο, για λόγους πληρότητας, θα γίνει μία σύντομη εισαγωγή στο πρωτόκολλο που χρησιμοποιείται

σήμερα για την ασφάλεια των συνδέσεων TCP. Για περισσότερες πληροφορίες σχετικά με την Ασφάλεια Δικτύων, ο αναγνώστης παραπέμπεται στο [19].

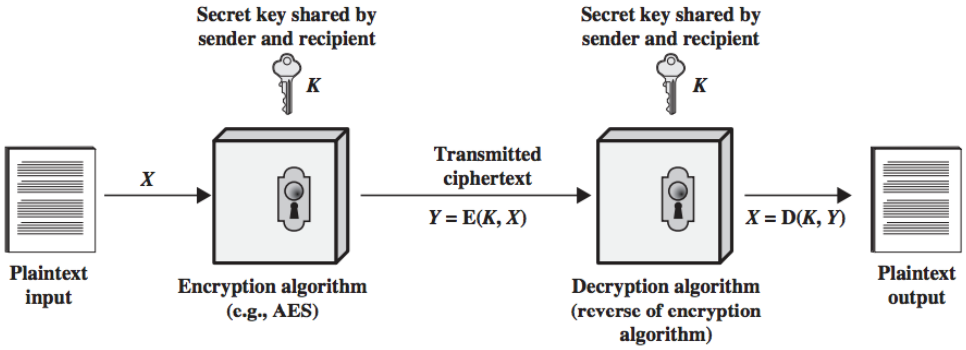
Το πρωτόκολλο που προσφέρει υπηρεσίες ασφαλείας στο TCP ονομάζεται Transport Layer Security (TLS), και νοηματικά τοποθετείται μεταξύ των στρωμάτων μεταφοράς και εφαρμογής στη στοίβα πρωτοκόλλων (βλ. Σχήμα 2.14). Το πρότυπο HTTP/2.0 απαιτεί κατηγορηματικά τη χρήση του πρωτοκόλλου TLS, για τη διασφάλιση όλων των συνδέσεων TCP μεταξύ πελατών και εξυπηρετητών στο Διαδίκτυο [20]. Στη συνέχεια, ακολουθεί η παρουσίαση των διαφοροποιήσεων που εισάγει το TLS στη διαδικασία εγκατάστασης σύνδεσης του TCP, αφού, όμως, προηγουμένως δοθούν ορισμένες βασικές αρχές κρυπτογράφησης.



Σχήμα 2.14 *Η στοίβα πρωτοκόλλων με ενσωμάτωση του TLS*
[Πηγή: https://hpbn.co/transport-layer-security-tls/]

2.7.1 Κρυπτογραφία Συμμετρικού Κλειδιού

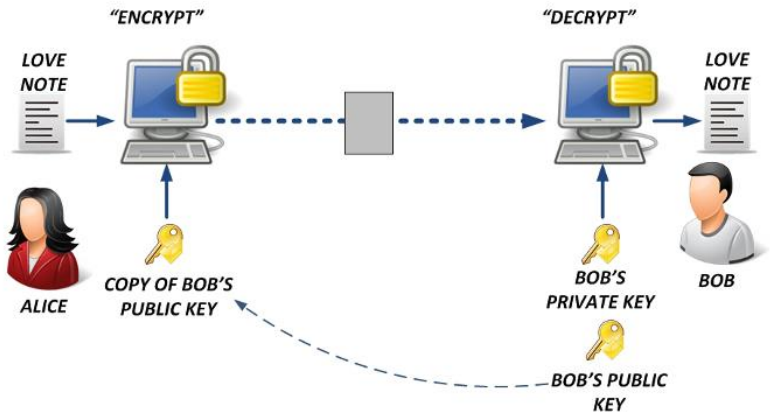
Κάθε αλγόριθμος κρυπτογράφησης περιλαμβάνει κατά βάση την αντικατάσταση ενός αρχικού μηνύματος, που βρίσκεται σε μορφή απλού κειμένου (plaintext), σε μορφή κρυπτοκειμένου (ciphertext). Απαραίτητη προϋπόθεση είναι η παροχή ενός κλειδιού (key) στον αλγόριθμο κρυπτογράφησης, το οποίο, γενικά, μπορεί να είναι μία σειρά από αριθμούς ή χαρακτήρες. Στα συστήματα συμμετρικού κλειδιού, ο αποστολέας και ο παραλήπτης χρησιμοποιούν το ίδιο κλειδί K για την κρυπτογράφηση και αποκρυπτογράφηση ενός μηνύματος [21], όπως απεικονίζεται στο Σχήμα 2.15. Δημοφιλείς αλγόριθμοι συμμετρικής κρυπτογράφησης είναι οι DES (Data Encryption Standard) και AES (Advanced Encryption Standard), μία επισκόπηση των οποίων μπορεί να βρεθεί στο [22]. Η χρήση αποκλειστικά συμμετρικής κρυπτογράφησης στο Διαδίκτυο θα απαιτούσε τη φυσική διανομή ενός κλειδιού K μεταξύ των επικοινωνουσών οντοτήτων, προκειμένου να αποφευχθούν φαινόμενα υποκλοπής, γεγονός το όποιο δεν είναι εφικτό. Με την εισαγωγή της κρυπτογράφησης δημόσιου κλειδιού, τα μοντέρνα συστήματα κρυπτογράφησης εξαλείφουν την ανάγκη αυτή.



Σχήμα 2.15 *Κρυπτογραφία Συμμετρικού Κλειδιού*
[Πηγή: https://notes.shichao.io/cnspp/ch2/]

2.7.2 Κρυπτογραφία Δημόσιου Κλειδιού

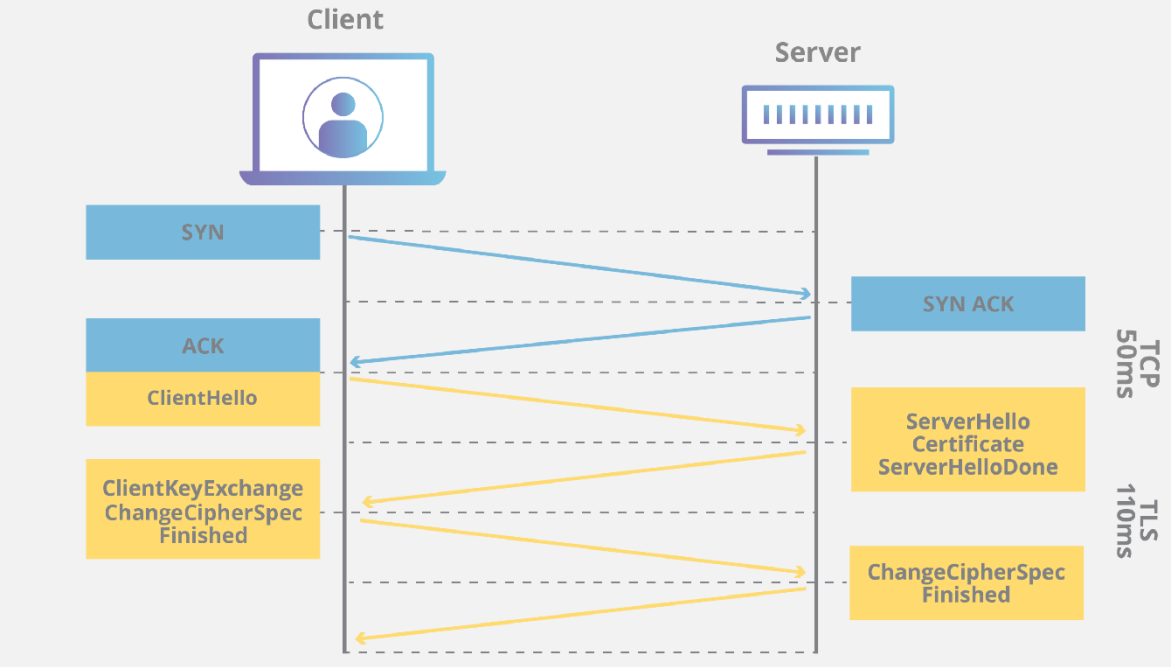
Τα συστήματα δημόσιου κλειδιού επιτρέπουν σε δύο οντότητες να επικοινωνήσουν με ασφάλεια, πάνω από ένα μη ασφαλές φυσικό μέσο. Αυτή η κρυπτογράφηση ονομάζεται ασύμμετρη, καθώς χρησιμοποιείται ένα ζεύγος κλειδιών [23]. Το ένα ονομάζεται δημόσιο κλειδί (public key) και γνωστοποιείται ελεύθερα από μία οντότητα στο Διαδίκτυο, ενώ το άλλο ονομάζεται ιδιωτικό κλειδί (private key) και πρέπει να παραμένει γνωστό μόνο στον ιδιοκτήτη του. Η παραγωγή του ζεύγους κλειδιών βασίζεται σε αλγορίθμους κρυπτογράφησης που χρησιμοποιούν τις ιδιότητες των πρώτων αριθμών και το πρόβλημα της παραγοντοποίησης. Ο αλγόριθμος Diffie - Hellman [24] και ο αλγόριθμος RSA [25] είναι οι δύο πλέον διαδεδομένοι. Ένα παράδειγμα χρήσης της ασύμμετρης κρυπτογραφίας απεικονίζεται στο Σχήμα 2.16. Το μειονέκτημά της είναι η υψηλή υπολογιστική πολυπλοκότητά της. Για το λόγο αυτό, οι δύο τεχνικές χρησιμοποιούνται συνδυαστικά. Ο αποστολέας επιλέγει ένα κλειδί *K*, το οποίο και γνωστοποιεί με ασφαλή τρόπο στον παραλήπτη, με χρήση του δημόσιου κλειδιού του. Ο παραλήπτης αποκρυπτογραφεί το μήνυμα με το ιδιωτικό κλειδί του και γνωρίζει πλέον το κοινό μυστικό κλειδί *K* που θα χρησιμοποιηθεί στην υπόλοιπη διάρκεια της σύνδεσης.



Σχήμα 2.16 *Κρυπτογραφία Δημόσιου Κλειδιού*
[Πηγή: http://richardgoyette.com/Infosec/Alice/BobandAlice.html]

2.7.3 Η χειραψία TLS

Πριν ξεκινήσει η ανταλλαγή δεδομένων εφαρμογής μεταξύ ενός πελάτη και ενός εξυπηρετητή, είναι απαραίτητη η συμφωνία στη χρήση ορισμένων παραμέτρων για την ασφάλεια της σύνδεσης. Η διαδικασία αυτή ονομάζεται χειραψία TLS, κατά την οποία τα δύο άκρα της σύνδεσης επιλέγουν από κοινού έναν αλγόριθμο κρυπτογράφησης και ανταλλάσσουν τα μυστικά κλειδιά. Στο Σχήμα 2.17 απεικονίζεται μία χειραψία TLS, που χρησιμοποιεί τον αλγόριθμο RSA για την ανταλλαγή των μυστικών κλειδιών:



Σχήμα 2.17 *Η χειραψία TLS με χρήση του αλγορίθμου RSA*

[Πηγή: https://www.cloudflare.com/learning/ssl/what-happens-in-a-tls-handshake/]

- Το πρωτόκολλο TLS λειτουργεί πάνω από το TCP, επομένως είναι απαραίτητη η αρχική εγκατάσταση μίας αξιόπιστης σύνδεσης TCP, όπως παρουσιάστηκε στην §2.3.1.

- Ο πελάτης αποστέλλει ένα μήνυμα “Client Hello”, στο οποίο περιλαμβάνει μία λίστα με τους αλγορίθμους κρυπτογράφησης που υποστηρίζει και ένα string από τυχαία bytes (client random).

- Ο εξυπηρετητής ανταποκρίνεται με ένα μήνυμα “Server Hello”, στο οποίο περιλαμβάνονται το πιστοποιητικό του server, ο αλγόριθμος κρυπτογράφησης που επιλέχθηκε και ένα string από τυχαία bytes (server random).

- Ο πελάτης λαμβάνει το πιστοποιητικό του server και ελέγχει την αυθεντικότητά του. Με αυτόν τον τρόπο, ο πελάτης επιτυγχάνει ταυτοποίηση του εξυπηρετητή. Στη συνέχεια, ο πελάτης εξάγει το δημόσιο κλειδί του εξυπηρετητή από το πιστοποιητικό.

- Ο πελάτης αποστέλλει ένα ακόμα τυχαίο string από bytes (premaster secret) με το μήνυμα "ClientKeyExchange", το οποίο κρυπτογραφεί με το δημόσιο κλειδί του εξυπηρετητή και μπορεί να αποκρυπτογραφηθεί μόνο με το αντίστοιχο ιδιωτικό κλειδί.

- Ο πελάτης και ο εξυπηρετητής χρησιμοποιούν τα client random, server random και premaster secret για να δημιουργήσουν τα μυστικά κλειδιά, χρησιμοποιώντας τον επιλεγμένο αλγόριθμο κρυπτογράφησης. Εξαιτίας των μαθηματικών ιδιοτήτων του, θα παραχθούν τα ίδια ακριβώς κλειδιά στα δύο άκρα της σύνδεσης.

- Ο πελάτης και ο εξυπηρετητής αποστέλλουν από ένα μήνυμα "Finished", κρυπτογραφημένο με το νέο μυστικό κλειδί, που περιέχει τη συνένωση όλων των μηνυμάτων που έστειλαν αντίστοιχα στη διάρκεια της χειραψίας.

Το τελευταίο βήμα προστατεύει τη διαδικασία της χειραψίας από ενδεχόμενη παραποίηση των μηνυμάτων της. Ύστερα από αυτό το σημείο, όλη η επικοινωνία μεταξύ πελάτη και εξυπηρετητή είναι κρυπτογραφημένη με το κοινό μυστικό κλειδί και μπορεί να εκκινήσει η ανταλλαγή δεδομένων εφαρμογής.

2.8 Σημερινά Προβλήματα του TCP

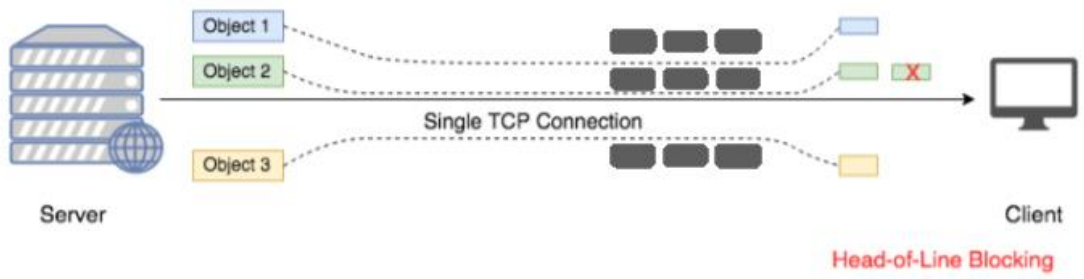
Το TCP αποτελεί το πλέον διαδεδομένο πρωτόκολλο μεταφοράς, το οποίο έχει υποστεί αρκετές βελτιστοποιήσεις από την εποχή που έγινε διαθέσιμο στο Διαδίκτυο μέχρι και σήμερα. Ωστόσο, πλέον, οι τρέχουσες εκδόσεις του TCP εμφανίζουν συλλογικά τα εξής προβλήματα:

- **Καθυστέρηση Εγκατάστασης Σύνδεσης**: Τόσο η αξιόπιστη μεταφορά δεδομένων μέσω του TCP όσο και η κρυπτογράφηση των δεδομένων με το πρωτόκολλο TLS, εισάγουν πρόσθετες καθυστερήσεις στην εκκίνηση ανταλλαγής δεδομένων εφαρμογής. Όπως παρουσιάστηκε στην §2.7.3, η διαδικασία ολοκλήρωσης των χειραψιών TCP και TLS καταναλώνει χρόνο τουλάχιστον ίσο με 3 RTT. Επομένως, η εμφάνιση μίας ιστοσελίδας στον browser ενός χρήστη μπορεί να καθυστερήσει σημαντικά, γεγονός που μειώνει την αντιληπτή ποιότητα υπηρεσίας.

- **Head-of-Line (HOL) Blocking**: Το πρωτόκολλο HTTP/2 υποστηρίζει την πολύπλεξη των αντικειμένων (objects) μίας ιστοσελίδας σε διαφορετικά και ανεξάρτητα HTTP streams. Τα streams αυτά μεταδίδονται πάνω από την ίδια σύνδεση με το πρωτόκολλο TCP, το οποίο όμως δεν είναι σε θέση να ξεχωρίζει σε ποιο stream ανήκουν τα δεδομένα ενός τμήματος TCP, όπως απεικονίζεται στο Σχήμα 2.18. Επομένως, σε περίπτωση απώλειας πακέτου, το TCP θα σταματήσει προσωρινά να διαβιβάζει δεδομένα προς το στρώμα εφαρμογής για όλα τα streams, ακόμα και αν κάποια από αυτά δεν έχουν υποστεί απώλεια και μπορούν να παραδώσουν τα δεδομένα τους στη σωστή σειρά.

- **IP/Port Identifier**: Μία σύνδεση TCP αναγνωρίζεται από μία τετράδα τιμών διευθύνσεων IP και θυρών TCP (βλ. §2.1). Οποιαδήποτε αλλαγή σε αυτές τις τιμές

θα προκαλέσει τερματισμό και επανεκκίνηση της σύνδεσης. Καθώς σήμερα ένας χρήστης μπορεί να έχει πρόσβαση στο Internet μέσω διαφορετικών τεχνολογιών, όπως για παράδειγμα Wi-fi ή 4G σε κινητές συσκευές, η εναλλαγή μεταξύ τους προκαλεί αλλαγή στη διεύθυνση IP του κινητού τερματικού. Συνεπώς, η επανεκκίνηση συνδέσεων TCP εξαιτίας τέτοιων αλλαγών στα κατώτερα στρώματα της στοίβας πρωτοκόλλων οδηγεί σε μειωμένη απόδοση και ποιότητα υπηρεσίας.



Σχήμα 2.18 *Head-of-Line (HOL) Blocking in TCP*
[Πηγή: https://marioskogias.github.io/students/adhi.pdf]

Η φύση αυτών των προβλημάτων συνδέεται άμεσα με τον πυρήνα της λειτουργίας του TCP. Η λύση τους θα απαιτούσε πρακτικά μεγάλες αλλαγές σε βασικούς μηχανισμούς του πρωτοκόλλου. Επομένως, έγινε αντιληπτή η ανάγκη για το σχεδιασμό ενός νέου πρωτοκόλλου μεταφοράς, το οποίο παρουσιάζεται στο Κεφάλαιο 3.

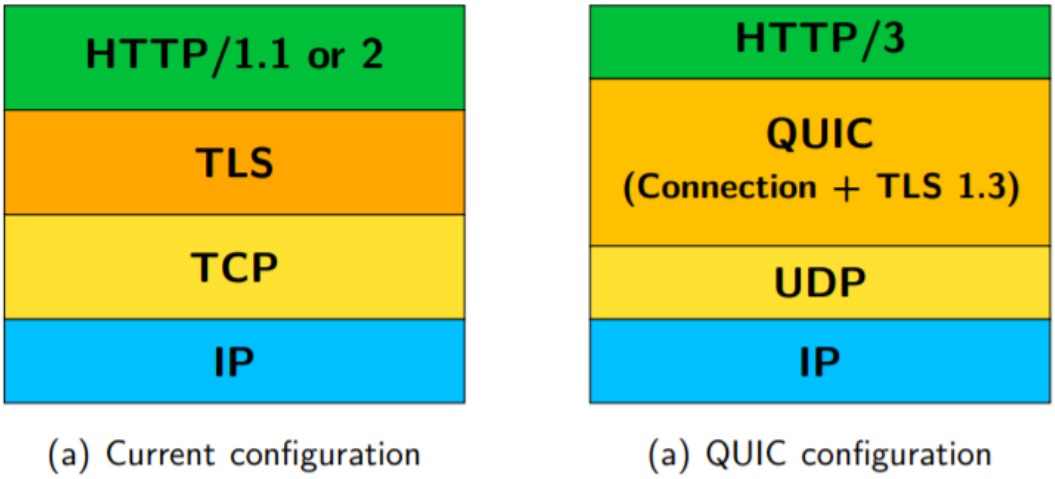
Κεφάλαιο 3. Quick UDP Internet Connections (QUIC)

Τα τελευταία χρόνια, οι μεγάλες εταιρίες που δραστηριοποιούνται στο χώρο του Internet προσπαθούν να βελτιώσουν την ποιότητα και την ασφάλεια των διαδικτυακών υπηρεσιών τους (web services). Η ανάπτυξη νέων εκδόσεων βασικών πρωτοκόλλων, όπως το HTTP/2 [20] και το TLS 1.3 [26], πραγματοποιήθηκε από τη διαδικτυακή επιστημονική και εταιρική κοινότητα με γνώμονα αυτή την κατεύθυνση. Ωστόσο, ο συνδυασμός του πρωτοκόλλου TCP με τα δύο αυτά πρωτόκολλα εξακολουθεί να αποτελεί ανασταλτικό παράγοντα της προσπάθειας για περαιτέρω βελτιστοποίηση της εμπειρίας του χρήστη (user experience) [27]. Επιπλέον, οποιαδήποτε αλλαγή στους μηχανισμούς του TCP, που στοχεύει στην επίλυση των προβλημάτων του (βλ. §2.8), αποτελεί χρονοβόρα και κοστοβόρα διαδικασία, όπως θα εξηγηθεί ακολούθως. Επομένως, αναδείχθηκε η ανάγκη για το σχεδιασμό ενός νέου πρωτοκόλλου μεταφοράς, με στόχο την επίλυση των ανωτέρω προβλημάτων.

Λαμβάνοντας υπόψη τα ανωτέρω, το 2013 προτάθηκε από τη Google ένα πρωτοποριακό πρωτόκολλο μεταφοράς, το οποίο ονομάστηκε Quick UDP Internet Connections - QUIC [28]. Ο σχεδιασμός του πρωτοκόλλου QUIC βασίστηκε σε βέλτιστες πρακτικές για την υλοποίηση μηχανισμών αξιοπιστίας και ασφάλειας σε δικτυακά πρωτόκολλα, όπως αυτές προέκυψαν ύστερα από δεκαετίες έρευνας και χρήσης του πρωτοκόλλου TCP. Το QUIC ενεργοποιήθηκε πειραματικά στον Google Chrome browser, με τις πρώτες μετρήσεις σε ελεγχόμενο περιβάλλον υποδομών της Google να εμφανίζουν σημαντικές βελτιώσεις στην ποιότητα υπηρεσίας: ταχύτερη εμφάνιση ιστοσελίδων (έως και 1s για το 99% των δοκιμών) και 30% λιγότερες παύσεις (video pauses) για βίντεο συνεχούς ροής [29]. Τα αποτελέσματα αυτά ήταν αρκετά ελπιδοφόρα ώστε να οδηγήσουν την οργάνωση IETF να αναλάβει το συνολικό συντονισμό του εγχειρήματος. Το 2016 συστάθηκε μία ομάδα εργασίας (QUIC Working Group), με κύριο στόχο το σχεδιασμό, την ανάπτυξη και την προτυποποίηση του πρωτοκόλλου QUIC, έργο το οποίο είναι μέχρι σήμερα σε εξέλιξη.

3.1 Η Στοίβα Πρωτοκόλλων του QUIC

Ένα εύλογο ερώτημα είναι η αιτία πίσω από την επιλογή για την ανάπτυξη ενός νέου πρωτοκόλλου, όπως το QUIC, αντί της ανανέωσης του υπάρχοντος πρωτοκόλλου TCP. Ο βασικός λόγος έγκειται στο ότι το TCP είναι ήδη υλοποιημένο στο λειτουργικό σύστημα (OS-Kernel space) των τερματικών συστημάτων. Επομένως, ενδεχόμενες αλλαγές στη λειτουργία του TCP απαιτούν την κυκλοφορία και εγκατάσταση αναβαθμίσεων (updates) στα λειτουργικά συστήματα, διαδικασία όμως που είναι εξαιρετικά αργή [30]. Επιπλέον, οι ISPs έχουν εξοπλίσει τα δίκτυά τους με διατάξεις ειδικά σχεδιασμένες για τη διαχείριση της δικτυακής κίνησης, όπως δρομολογητές, τείχη προστασίας (firewalls) και ισορροπιστές φορτίου (load balancers). Τέτοιες συσκευές έχουν συλλογικά επονομαστεί ως «ενδιάμεσα κουτιά» (middleboxes), χωρίς τα οποία δεν θα λειτουργούσε το Διαδίκτυο. Συνεπώς, οποιαδήποτε αλλαγή στο TCP θα απαιτούσε επιπλέον την αναβάθμιση των middleboxes σε παγκόσμια κλίμακα. Το εγχείρημα αυτό, εκτός από χρονοβόρο, είναι σε κάθε περίπτωση και επίφοβο, αν αναλογιστεί κανείς τον αριθμό των κρίσιμων υπηρεσιών που βασίζονται στην ορθή λειτουργία του Internet.



Σχήμα 3.1 *Η στοίβα πρωτοκόλλων του QUIC*
[Πηγή: https://webthesis.biblio.polito.it/10904/1/tesi.pdf]

Για την αντιμετώπιση αυτών των δύο προβλημάτων, το QUIC ακολουθεί μία νέα αρχιτεκτονική για την οργάνωση της στοίβας πρωτοκόλλων, η οποία απεικονίζεται στο Σχήμα 3.1. Τα βασικά σημεία του νέου σχεδιασμού είναι τα εξής:

- Το πρωτόκολλο QUIC χρησιμοποιεί ως «υπόστρωμα» το πρωτόκολλο UDP, δηλαδή κάθε πακέτο QUIC ενθυλακώνεται σε ένα UDP datagram, πριν αποσταλεί στο δίκτυο. Με τον τρόπο αυτό, διατηρείται η συμβατότητα με τα υπάρχοντα middleboxes, τα οποία, στη γενική περίπτωση, απορρίπτουν πακέτα που χρησιμοποιούν πρωτόκολλα πέραν των TCP/UDP. Επιπλέον, αποφεύγεται η χρονική καθυστέρηση που προκαλείται από την τριμερή χειραψία του TCP κατά την εγκατάσταση μιας σύνδεσης TCP.

- Δεδομένου ότι το UDP, ως «ασυνδεσμικό» πρωτόκολλο, δεν προσφέρει υπηρεσία αξιόπιστης παράδοσης (βλ. §1.3.2), το QUIC είναι υπεύθυνο για την υλοποίηση αυτής της υπηρεσίας, με μηχανισμούς ανίχνευσης απωλειών και αναμεταδόσεων, όπως θα αναλυθούν στη συνέχεια. Επιπλέον, για κρυπτογράφηση των δεδομένων, το QUIC ενσωματώνει ως απαραίτητες τις λειτουργίες του πρωτοκόλλου TLS. Συνεπώς, το πρωτόκολλο QUIC πρακτικά συνενώνει τις λειτουργίες των TCP και TLS σε ένα ενιαίο στρώμα της στοίβας πρωτοκόλλων.

- Εκτός από το QUIC, η IETF ετοιμάζει και μία νέα έκδοση του πρωτοκόλλου HTTP, την HTTP/3, η οποία θα είναι ειδικά σχεδιασμένη να λειτουργεί πάνω από το πρωτόκολλο QUIC [31].

- Σε αντίθεση με τα TCP και UDP, το QUIC είναι υλοποιημένο στο χώρο χρήστη (user space) ενός τερματικού συστήματος. Αυτό σημαίνει ότι οποιαδήποτε αλλαγή στους μηχανισμούς του QUIC, ακόμα και ουσιώδης, δεν απαιτεί καμία αναβάθμιση του λειτουργικού συστήματος. Επομένως, με την εισαγωγή αυτής της καινοτομίας, επιταχύνεται η διανομή (distribution) και ο έλεγχος (testing) του QUIC, καθώς και η κυκλοφορία των μελλοντικών βελτιωμένων εκδόσεων.

3.2 Επισκόπηση Νέων Χαρακτηριστικών

Το πρωτόκολλο QUIC αποσκοπεί στην αποδοτική, αξιόπιστη και ασφαλή μεταφορά δεδομένων μεταξύ δύο τερματικών σημείων. Για το σκοπό αυτό, το QUIC υιοθετεί νέα χαρακτηριστικά, αντιμετωπίζοντας παράλληλα τα προβλήματα που εισάγει το TCP. Το σύνολο των νέων αυτών χαρακτηριστικών θα αναλυθεί στη συνέχεια, με τα σπουδαιότερα από αυτά να είναι συνοπτικά τα εξής:

- **Μείωση Καθυστέρησης Εγκατάστασης Σύνδεσης**: Μία τυπική χειραψία του συνδυασμού των πρωτοκόλλων TCP και TLS εισάγει καθυστέρηση ίση με 3RTT, πριν μεταδοθούν τα πρώτα δεδομένα εφαρμογής (βλ. Σχήμα 2.17). Η έκδοση TLS 1.3 επιτρέπει τη μετάδοση δεδομένων εφαρμογής μέσα στο μήνυμα "Finished" από τον πελάτη προς τον εξυπηρετητή, μειώνοντας έτσι το χρόνο αυτό σε 2RTT. Αντιθέτως, επειδή το QUIC ενσωματώνει πλήρως τις λειτουργικότητες του TLS, έχει τη δυνατότητα να μεταδίδει όλες τις απαραίτητες παραμέτρους κρυπτογράφησης μέσα στο πρώτο πακέτο QUIC που αποστέλλεται προς τον εξυπηρετητή. Με τον τρόπο αυτό, η χρονική καθυστέρηση για την εγκατάσταση μιας σύνδεσης QUIC μειώνεται σε 1RTT. Τέλος, αν ο πελάτης και ο εξυπηρετητής έχουν επικοινωνήσει ξανά στο παρελθόν, μπορούν να χρησιμοποιήσουν αποθηκευμένες πληροφορίες και να πραγματοποιήσουν μία χειραψία 0-RTT. Στην περίπτωση αυτή, δεδομένα εφαρμογής αποστέλλονται από το πρώτο κιόλας πακέτο του πελάτη προς τον εξυπηρετητή, επιτυγχάνοντας έτσι τη βέλτιστη επίδοση όσον αφορά τη διαδικασία της εγκατάστασης σύνδεσης.

- **QUIC streams**: Το πρωτόκολλο QUIC, ακολουθώντας το πρότυπο του HTTP/2, σχεδιάστηκε ώστε να υποστηρίζει την πολύπλεξη δεδομένων σε διαφορετικές ροές (streams) επιπέδου μεταφοράς, οι οποίες μεταδίδονται πάνω από την ίδια σύνδεση QUIC. Τα δεδομένα κάθε stream μεταφέρονται μέσα σε πλαίσια (frames), ενώ ένα πακέτο QUIC μπορεί να μεταφέρει frames από διαφορετικά streams (βλ. §3.4). Το QUIC θεωρεί τα streams αυτά ανεξάρτητα και εγγυάται ότι τα δεδομένα που μεταφέρει κάθε stream ξεχωριστά θα μεταδοθούν αξιόπιστα και με σωστή σειρά στον παραλήπτη. Έτσι, επιλύεται το πρόβλημα του Head of Line Blocking του TCP, καθώς σε περίπτωση απώλειας ενός πακέτου, θα «παγώσουν» μόνο τα streams των οποίων τα frames περιέχονταν στο χαμένο πακέτο. Τα υπόλοιπα streams, για τα οποία δεν έχει χαθεί κάποιο frame, μπορούν να συνεχίσουν κανονικά να παραδίδουν δεδομένα εφαρμογής στην πλευρά του παραλήπτη.

- **Connection ID**: Όπως αναφέρθηκε προηγουμένως, το QUIC λειτουργεί πάνω από το πρωτόκολλο UDP. Αυτό σημαίνει ότι μία σύνδεση QUIC μεταξύ δύο τερματικών σημείων μπορεί να αναγνωριστεί μέσω της εξής τετράδας τιμών της σύνδεσης UDP: διεύθυνση IP πηγής και προορισμού, θύρα (port) πηγής και προορισμού. Ωστόσο, οποιαδήποτε αλλαγή στις διευθύνσεις IP, θα προκαλέσει διακοπή της σύνδεσης UDP. Προκειμένου μία σύνδεση QUIC να μην επηρεάζεται από αλλαγές στο επίπεδο IP, γίνεται εισαγωγή της μεταβλητής *ConnectionID*. Επομένως, σε περίπτωση όπου η διεύθυνση IP ενός QUIC client αλλάξει κατά τη διάρκεια επικοινωνίας με έναν QUIC server, η σύνδεση QUIC δεν θα διακοπεί, υπό την προϋπόθεση ότι ο client συνεχίσει να χρησιμοποιεί το ίδιο *ConnectionID*.

3.3 Πακέτα και Επικεφαλίδες QUIC

Δύο τερματικά συστήματα που χρησιμοποιούν το πρωτόκολλο QUIC για να επικοινωνήσουν ανταλλάσσουν μεταξύ τους πακέτα QUIC. Σε κάθε πακέτο προστίθεται μία επικεφαλίδα QUIC, πριν αυτό μεταβιβαστεί στο πρωτόκολλο UDP και ενσωματωθεί σε ένα δεδομενόγραμμα. Συγκριτικά με το TCP, το QUIC διαθέτει δύο μορφές επικεφαλίδας: η μία μορφή χρησιμοποιείται για τα πακέτα που ανταλλάσσονται κατά την εγκατάσταση της σύνδεσης, ενώ η άλλη χρησιμοποιείται για την μετέπειτα ανταλλαγή δεδομένων. Η διαφοροποίηση αυτή υιοθετήθηκε προκειμένου να ελαχιστοποιηθεί το μέγεθος της επικεφαλίδας των πακέτων QUIC, καθώς μία επικεφαλίδα δεν περιέχει «ωφέλιμο» φορτίο, υπό την έννοια ότι δεν περιλαμβάνει δεδομένα εφαρμογής αλλά πληροφορίες που χρειάζονται για τη διαχείριση της σύνδεσης.

3.3.1 Μέγεθος Πακέτων QUIC

Το μέγεθος ενός πακέτου QUIC ορίζεται ως το άθροισμα της επικεφαλίδας και του ωφέλιμου φορτίου (payload), ενώ δεν περιλαμβάνεται η επικεφαλίδα UDP ή IP. Το μέγιστο μέγεθος ενός πακέτου QUIC εξαρτάται από τη Μέγιστη Μονάδα Μεταφοράς (Maximum Transmission Unit - MTU) που υποστηρίζει το πρωτόκολλο IP για τη σύνδεση [32]. Η Μέγιστη Μονάδα Μεταφοράς αποτελεί το μέγιστο μέγεθος ενός πακέτου IP (συμπεριλαμβανομένης της επικεφαλίδας IP), που μπορεί να σταλεί από τον αποστολέα στον παραλήπτη και δίνεται από την εξίσωση:

$$MTU = IP_{header} + UDP_{header} + \underbrace{QUIC_{header} + QUIC_{payload}}_{UDP_{payload}} \tag{3.1}$$

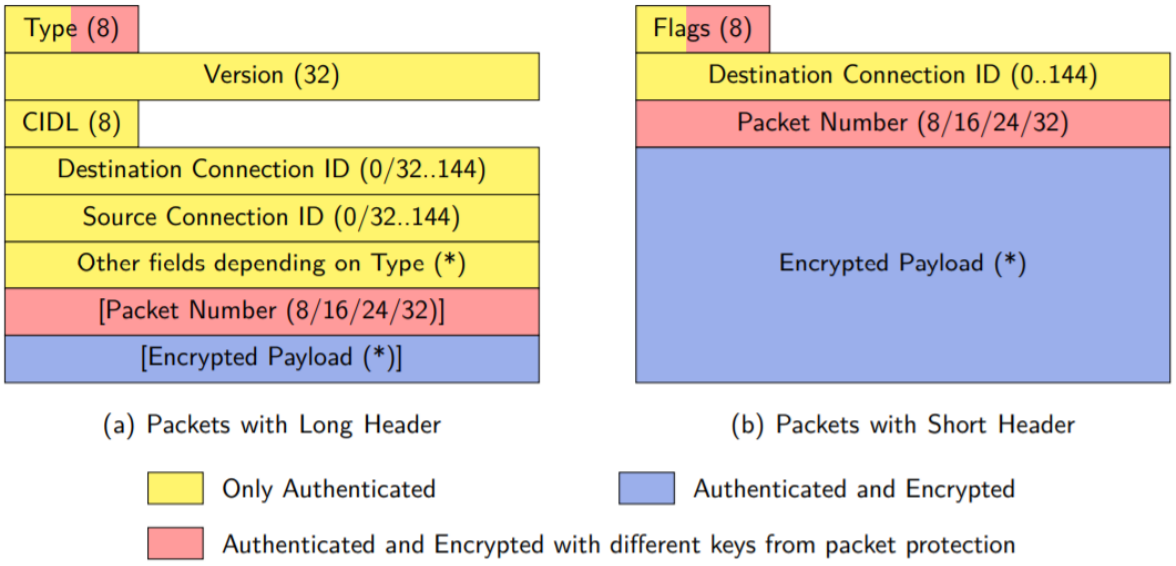
Επομένως, το μέγιστο μέγεθος ενός πακέτου QUIC είναι ίσο με το μέγιστο payload ενός δεδομενογράμματος UDP. Η προεπιλεγμένη τιμή της MTU για το πρωτόκολλο QUIC είναι 1280 bytes. Αφαιρώντας τις επικεφαλίδες IP και UDP, το προεπιλεγμένο μέγιστο μέγεθος ενός πακέτου QUIC είναι 1252 bytes. Τέλος, επιπλέον του ανωτέρω περιορισμού, το πρώτο πακέτο QUIC που αποστέλλεται από τον πελάτη στον εξυπηρετητή οφείλει να έχει μέγεθος τουλάχιστον ίσο με 1200 bytes, για λόγους που θα αναφερθούν στη συνέχεια.

3.3.2 Long Packet Header

Τα πακέτα QUIC που ανταλλάσσονται κατά τη διάρκεια της εγκατάστασης σύνδεσης, χρειάζεται να μεταφέρουν αρκετές πληροφορίες στην επικεφαλίδα τους, η οποία ονομάζεται “long header”. Ωστόσο, μόλις ολοκληρωθεί η χειραψία TLS και συμφωνηθεί το συμμετρικό κλειδί που θα χρησιμοποιηθεί για την αποκρυπτογράφηση δεδομένων, τα πακέτα QUIC χρησιμοποιούν την επονομαζόμενη “short header”, η οποία μεταφέρει μόνο ορισμένες απαραίτητες πληροφορίες, ελαχιστοποιώντας έτσι το πλήθος των επιπλέον bytes που προστίθενται σε ένα πακέτο QUIC.

Η δομή των δύο διαφορετικών επικεφαλίδων που χρησιμοποιεί το QUIC απεικονίζεται στο Σχήμα 3.2, ενώ ταυτόχρονα επισημαίνεται με διαφορετικό χρώμα το επίπεδο ασφάλειας με το οποίο προστατεύεται κάθε πεδίο. Με κίτρινο χρώμα

υποδεικνύονται τα πεδία για τα οποία γίνεται μόνο επαλήθευση της ταυτότητας του αποστολέα (authentication), ενώ τα υπόλοιπα πεδία διαθέτουν ταυτοποίηση και κρυπτογράφηση. Περισσότερες πληροφορίες σχετικά με την ασφάλεια του πρωτοκόλλου QUIC δίνονται στην §3.7.



Σχήμα 3.2 *Η δομή των πακέτων QUIC*
[Πηγή: https://webthesis.biblio.polito.it/10904/1/tesi.pdf]

Τα πεδία που περιέχονται σε μία long header ενός πακέτου QUIC είναι τα ακόλουθα:

- **Type – 8 bits**: Διαχωρίζει τα διαφορετικού τύπου πακέτα που χρησιμοποιούν long headers και έχει το εξής μοτίβο: **11TTRRPP**. Το 1ο bit τίθεται στην τιμή 1 στην περίπτωση των long headers. Το 2ο bit έχει σταθερή τιμή ίση με 1. Τα επόμενα δύο bits (TT) κωδικοποιούν τον τύπο του πακέτου QUIC, με τις δυνατές τιμές να φαίνονται στον Πίνακα 3.1. Τα πακέτα Initial και Handshake χρησιμοποιούνται κατά την εγκατάσταση της σύνδεσης, όπως θα αναλυθεί στη συνέχεια, ενώ τα πακέτα 0-RTT χρησιμοποιούνται για την πρώιμη μετάδοση δεδομένων εφαρμογής, στις περιπτώσεις όπου ο πελάτης έχει επικοινωνήσει πρόσφατα στο παρελθόν με τον εξυπηρετητή. Οι περιπτώσεις αυτές είναι και οι πλέον σημαντικές. Τα πακέτα Retry χρησιμοποιούνται από ένα εξυπηρετητή όταν απορρίπτει μία νέα σύνδεση και σηματοδοτεί στον πελάτη να στείλει ξανά το αίτημά του. Τα RR bits είναι δεσμευμένα για μελλοντική χρήση. Τέλος, στα δύο τελευταία bits (PP) κωδικοποιείται το μήκος (από 1 έως 4 bytes) του πεδίου της επικεφαλίδας, που περιέχει τον αριθμό πακέτου (packet number).

Packet Type (TT bits)	Name
0x0 (00)	Initial
0x1 (01)	0-RTT
0x2 (10)	Handshake
0x3 (11)	Retry

Πίνακας 3.1 *Διαφορετικοί τύποι πακέτων QUIC για την περίπτωση των long headers*

- **Έκδοση – 32 bits**: Υποδεικνύει τον αριθμό της έκδοσης του πρωτοκόλλου QUIC που χρησιμοποιείται. Καθορίζει τον τρόπο με τον οποίο ερμηνεύονται τα επόμενα πεδία της επικεφαλίδας.

- **Μήκος Connection ID (CIDL) – 8 bits**: Περιέχει το μήκος (σε bytes) των Connection ID's πηγής και προορισμού. Τα τέσσερα περισσότερο σημαντικά bits χρησιμοποιούνται για το Connection ID προορισμού (destination), ενώ τα τέσσερα λιγότερο σημαντικά bits χρησιμοποιούνται για το Connection ID πηγής (source). Μηδενική τιμή υποδηλώνει ότι η αντίστοιχη Connection ID έχει μηδενικό μήκος, δηλαδή δεν χρησιμοποιείται. Οι μη μηδενικές τιμές πρέπει να αυξηθούν κατά 3, προκειμένου να εξαχθεί το σωστό μήκος. Επομένως, το μήκος μίας μη μηδενικής Connection ID ανήκει στο διάστημα [4, 18] bytes.

- **Connection ID πηγής/προορισμού – 32 έως 144 bits**: Στην περίπτωση που το μήκος του αντίστοιχου Connection ID είναι μηδενικό, τότε και το πεδίο αυτό έχει μήκος 0 bits. Σε αντίθετη περίπτωση, το πεδίο αυτό περιέχει την αντίστοιχη Connection ID (πηγής ή προορισμού). Σε ένα τυπικό σενάριο επικοινωνίας ενός πελάτη με ένα εξυπηρετητή, κάθε άκρο της σύνδεσης επιλέγει αυθαίρετα το δικό του Connection ID, το οποίο και επικοινωνεί στο άλλο άκρο. Για παράδειγμα, ένα πακέτο που στέλνει ο πελάτης στον εξυπηρετητή, θα έχει ως Source Connection ID την τιμή που έχει διαλέξει ο πελάτης, ενώ ως Destination Connection ID θα έχει την τιμή που έχει διαλέξει ο εξυπηρετητής. Υπενθυμίζεται ότι το Connection ID προσθέτει στο πρωτόκολλο QUIC ανθεκτικότητα, όσον αφορά τις αλλαγές παραμέτρων σε υποκείμενα στρώματα της στοίβας πρωτοκόλλων. Με αυτό τον τρόπο, μία αλλαγή στη διεύθυνση IP ενός τερματικού δεν θα προκαλέσει διακοπή της σύνδεσης QUIC, εφόσον το τερματικό δεν αλλάξει την τιμή του πεδίου Source Connection ID στα πακέτα που αποστέλλει. Συνεπώς, όταν το πακέτο αυτό φθάσει στον παραλήπτη, καίτοι έχει διαφορετική διεύθυνση IP πηγής, θα μπορέσει να αναγνωριστεί ως πακέτο της συγκεκριμένης σύνδεσης QUIC, από τη στιγμή που διαθέτει έγκυρο Connection ID [33].

- **Μήκος – 1/2/4/8 bytes**: Το πεδίο αυτό δηλώνει το υπολειπόμενο μήκος του πακέτου QUIC, δηλαδή το άθροισμα των πεδίων του αριθμού πακέτου και του payload. Το πεδίο αυτό είναι παρών μόνο στα πακέτα τύπου Initial, Handshake και 0-RTT.

- **Αριθμός Πακέτου – 8/16/24/32 bits**: Χρησιμοποιείται για τη γνησίως αύξουσα αρίθμηση των πακέτων QUIC που αποστέλλει ένα τερματικό σε μία σύνδεση. Συνεπώς, ένα πακέτο με μεγαλύτερο αριθμό πακέτου έχει σταλεί σε μετέπειτα χρονική στιγμή, συγκριτικά με ένα πακέτο με μικρότερο αριθμό πακέτου. Σε αντιπαράθεση με το TCP, ο αριθμός πακέτου QUIC είναι ο αντίστοιχος αριθμός ακολουθίας ενός τμήματος TCP, καθώς χρησιμοποιείται από το QUIC για την υλοποίηση της αξιόπιστης υπηρεσίας παράδοσης. Ωστόσο, ο αριθμός πακέτου και ο αριθμός ακολουθίας είναι δύο εντελώς διαφορετικές έννοιες, όπως εξηγείται στην §3.5.

- **Κρυπτογραφημένο Ωφέλιμο Φορτίο**: Το ωφέλιμο φορτίο ενός πακέτου QUIC αποτελείται από μία ακολουθία πλαισίων (frames), που μεταφέρουν δεδομένα

εφαρμογής ή πληροφορίες ελέγχου για τη διαχείριση της σύνδεσης. Οι διαφορετικοί τύποι πλαισίων παρουσιάζονται στην §3.4.

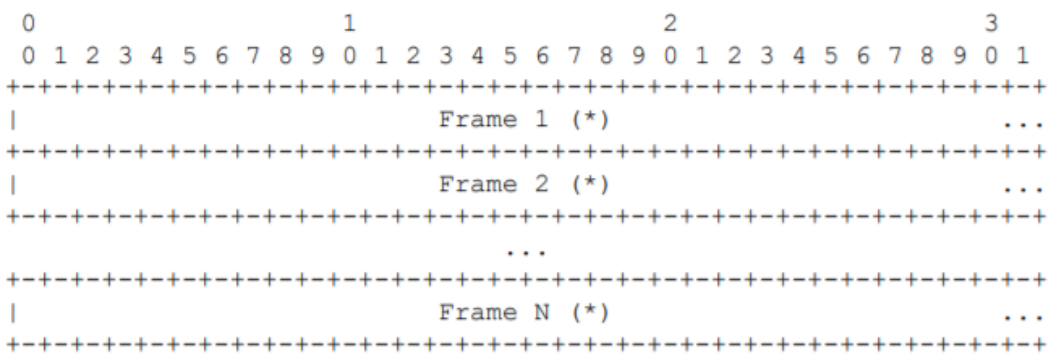
3.3.3 Short Packet Header

Με την ολοκλήρωση της εγκατάστασης της σύνδεσης QUIC, τα δύο τερματικά έχουν διαθέσιμο ένα κοινό μυστικό κλειδί, το οποίο και χρησιμοποιούν για την κρυπτογράφηση των δεδομένων. Τα πακέτα που κρυπτογραφούνται με αυτό το κλειδί, ονομάζονται πακέτα 1-RTT και χρησιμοποιούν το δεύτερο τύπο επικεφαλίδας (short header). Τα πεδία που είναι κοινά με μία long header έχουν την ίδια ακριβώς ερμηνεία, με τη μόνη διαφοροποίηση να εντοπίζεται στα πρώτα 8 bits (βλ. Σχήμα 3.2):

- **Σημαίες - 8 bits**: Το πρώτο byte μίας short header ακολουθεί το εξής μοτίβο: **01SRRKPP**. Το 1ο bit τίθεται στην τιμή 0, στην περίπτωση των short headers. Το 2ο bit έχει σταθερή τιμή ίση με 1. Το 3ο bit (S) ονομάζεται Spin Bit, χρησιμεύει στην μέτρηση του χρόνου RTT από έναν παθητικό παρατηρητή και αναλύεται στην §5.3. Τα επόμενα δύο bits (RR) είναι δεσμευμένα για μελλοντική χρήση. Το 6ο bit (K) ονομάζεται bit Φάσης Κλειδιού (Key Phase bit). Μόλις η σύνδεση εγκατασταθεί, παρέχεται η δυνατότητα στα δύο άκρα της σύνδεσης να ανανεώνουν περιοδικά το συμμετρικό κλειδί που χρησιμοποιούν για την κρυπτογράφηση. Προκειμένου να μπορεί να αναγνωρίσει ένας παραλήπτης ποιο κλειδί έχει χρησιμοποιηθεί για την κρυπτογράφηση ενός πακέτου, χρησιμοποιεί την τιμή του bit Φάσης Κλειδιού. Η αρχική τιμή αυτού του bit είναι 0, ενώ η τιμή του αντιστρέφεται με κάθε ανανέωση κλειδιού (key update). Ανιχνεύοντας, επομένως, την αντιστροφή της τιμής του bit Φάσης Κλειδιού, ο παραλήπτης γνωρίζει ποιο κλειδί πρέπει να χρησιμοποιήσει, ώστε να αποκρυπτογραφήσει σωστά το πακέτο. Τέλος, τα PP bits κωδικοποιούν το μήκος (από 1 έως 4 bytes) του πεδίου της επικεφαλίδας, που περιέχει τον αριθμό πακέτου (packet number).

3.4 Πλαίσια QUIC

Το φορτίο (payload) ενός πακέτου QUIC αποτελείται από μία ακολουθία πλαισίων (frames), όπως απεικονίζεται στο Σχήμα 3.3. Ένα πακέτο QUIC οφείλει να περιέχει τουλάχιστον ένα πλαίσιο QUIC, ενώ τα πλαίσια μπορεί να είναι διαφορετικού τύπου .



Σχήμα 3.3 *Το payload ενός πακέτου QUIC* [34]

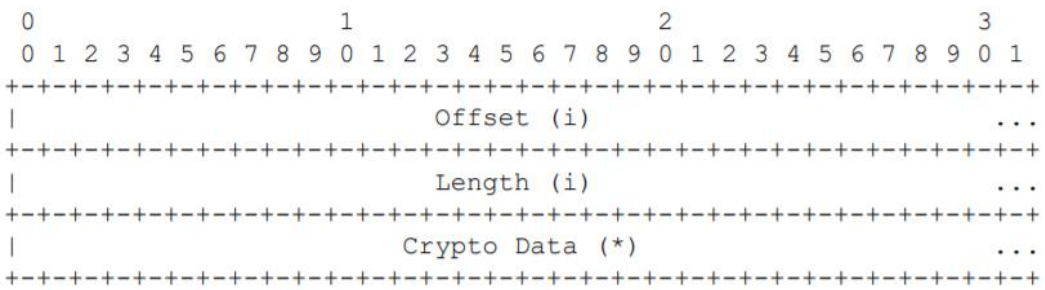
Frame Type Value	Frame Type Name
0x00	PADDING
0x01	PING
0x02 - 0x03	ACK
0x04	RESET_STREAM
0x05	STOP_SENDING
0x06	CRYPTO
0x07	NEW_TOKEN
0x08 - 0x0f	STREAM
0x10	MAX_DATA
0x11	MAX_STREAM_DATA
0x12 - 0x13	MAX_STREAMS
0x14	DATA_BLOCKED
0x15	STREAM_DATA_BLOCKED
0x16 - 0x17	STREAMS_BLOCKED
0x18	NEW_CONNECTION_ID
0x19	RETIRE_CONNECTION_ID
0x1a	PATH_CHALLENGE
0x1b	PATH_RESPONSE
0x1c - 0x1d	CONNECTION_CLOSE

Πίνακας 3.2 *Διαφορετικοί τύποι πλαισίων QUIC*

Το πρώτο πεδίο ενός πλαισίου QUIC είναι ο τύπος του πλαισίου. Οι διαφορετικοί τύποι πλαισίων που ορίζει το πρωτόκολλο QUIC παρουσιάζονται στον Πίνακα 3.2. Στη συνέχεια, αναλύονται οι σημαντικότεροι τύποι πλαισίων, οι οποίοι χρησιμοποιούνται από τους μηχανισμούς που θα παρουσιαστούν σε επόμενες παραγράφους. Για την ανάλυση όλων των τύπων πλαισίων, ο αναγνώστης παραπέμπεται στο [34].

3.4.1 Πλαίσιο CRYPTO

Ένα πλαίσιο CRYPTO χρησιμοποιείται για τη μεταφορά όλων των μηνυμάτων που, μέσω της χειραψίας TLS, ανταλλάσσονται κατά τη διαδικασία της εγκατάστασης μίας ασφαλούς σύνδεσης. Πρακτικά, όλα τα μηνύματα του πρωτοκόλλου TLS που μεταφέρονται πάνω από μία σύνδεση QUIC, ενθυλακώνονται σε πλαίσια CRYPTO. Η δομή και τα πεδία ενός πλαισίου CRYPTO έχουν ως εξής:



Σχήμα 3.4 *Η δομή του πλαισίου CRYPTO* [34]

- **Offset**: Κάθε τύπος πλαισίου είναι υπεύθυνος για την παράδοση δεδομένων στον παραλήπτη, με τη σωστή σειρά με την οποία στάλθηκαν. Για παράδειγμα, τα πλαίσια CRYPTO είναι υπεύθυνα για την αξιόπιστη παράδοση δεδομένων που σχετίζονται με το πρωτόκολλο TLS. Υπό αυτό το πρίσμα, το πεδίο offset αντιπροσωπεύει τη σειριακή αρίθμηση των bytes που μεταφέρονται σε αυτό το πλαίσιο, ως προς το πρώτο byte που στάλθηκε μέσα σε ένα πλαίσιο CRYPTO για τη σύνδεση αυτή. Το πρώτο byte της ροής δεδομένων CRYPTO σε μία σύνδεση QUIC έχει στο πεδίο offset την τιμή 0. Σε αντιπαράθεση με το TCP, το πεδίο offset έχει αντίστοιχη λειτουργικότητα με τον αριθμό ακολουθίας, καθώς επιτρέπει στον παραλήπτη να αναδιατάσσει τα δεδομένα στη σωστή τους σειρά.

- **Μήκος**: Καθορίζει το μήκος των δεδομένων που μεταφέρει ένα πλαίσιο CRYPTO.

- **Δεδομένα Crypto**: Εδώ μεταφέρονται τα δεδομένα του πρωτοκόλλου TLS. Για παράδειγμα, μπορεί να περιέχεται ένα μήνυμα “Client Hello” ή “Server Hello” (βλ. §3.7.1).

3.4.2 Πλαίσιο STREAM

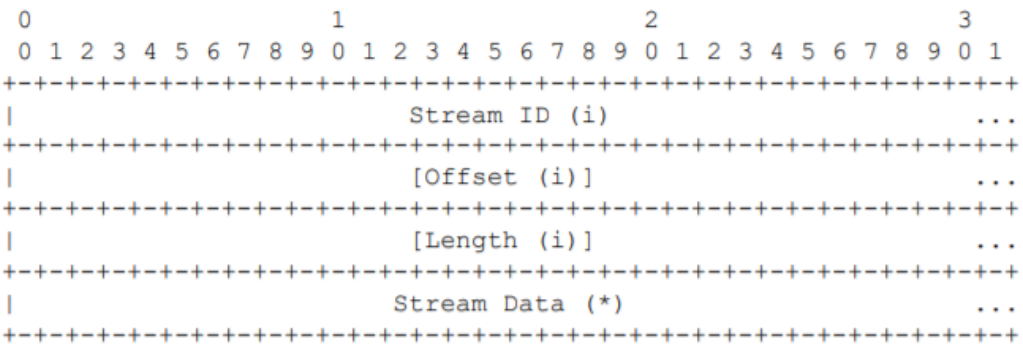
Το πρωτόκολλο QUIC επεκτείνει το μηχανισμό των HTTP streams στο επίπεδο μεταφοράς, με την εισαγωγή των QUIC streams. Κάθε αντικείμενο (object) μίας ιστοσελίδας HTTP, αντιστοιχίζεται μονοσήμαντα σε ένα QUIC stream. Ένα QUIC stream εγγυάται ότι το συνολικό ρεύμα δεδομένων που του αντιστοιχεί (δηλαδή το αντίστοιχο object), θα διαβιβαστεί στον παραλήπτη με τη σωστή σειρά, προσφέροντας πρακτικά μία υπηρεσία αξιόπιστης παράδοσης. Κάθε QUIC stream είναι ανεξάρτητο από τα υπόλοιπα, ενώ διαθέτει έναν αριθμό ως αναγνωριστικό (stream ID), που είναι μοναδικός για κάθε stream. Τα QUIC streams που εκκινεί ένας client, έχουν άρτιο αριθμό ως stream ID, ενώ τα streams που εκκινεί ένας server έχουν περιττό αριθμό ως stream ID.

Τα δεδομένα ενός QUIC stream ενθυλακώνονται σε πλαίσια τύπου STREAM. Ο παραλήπτης ενός πλαισίου STREAM μπορεί να αναγνωρίσει σε ποιο stream ανήκει το πλαίσιο, με βάση το stream ID. Για την περίπτωση των πλαισίων STREAM, το πεδίο που περιέχει τον τύπο του πλαισίου (frame type), έχει το δεκαεξαδικό εύρος 0x08 έως 0x0f, δηλαδή είναι της μορφής: 0b00001ABC. Τα πεδία που περιέχονται σε ένα πλαίσιο STREAM, όπως αυτό απεικονίζεται στο Σχήμα 3.5, καθορίζονται από την τιμή των τριών τελευταίων bits ως εξής:

- Το bit A ονομάζεται OFF bit και τίθεται στην τιμή 1, αν το πλαίσιο περιέχει το πεδίο Offset. Αν το OFF bit έχει την τιμή 0, αυτό σημαίνει ότι το πλαίσιο περιέχει τα πρώτα bytes που έχουν σταλεί για αυτό το stream, με την αρίθμηση να εκκινεί από το 0.

- Το bit B ονομάζεται LEN bit και υποδηλώνει την παρουσία ή μη του πεδίου μήκους (Length). Το LEN bit τίθεται στην τιμή 1, όταν το πεδίο Length περιέχεται στο πλαίσιο, και στην τιμή 0 σε αντίθετη περίπτωση.

- Το bit C ονομάζεται FIN bit και τίθεται στην τιμή 1, προκειμένου να σηματοδοτηθεί η ολοκλήρωση της μετάδοσης δεδομένων για αυτό το stream.



Σχήμα 3.5 *Η δομή του πλαισίου STREAM* [34]

Τα υπόλοιπα πεδία του πλαισίου STREAM είναι τα εξής:

- **Stream ID**: Εδώ περιέχεται το stream ID και υποδεικνύει το stream στο οποίο ανήκει αυτό το πλαίσιο.

- **Offset**: Περιέχει την αρίθμηση των bytes που μεταφέρει αυτό το πλαίσιο, ως το πρώτο byte του συνολικού ρεύματος δεδομένων για το συγκεκριμένο stream. Η αρίθμηση των bytes εκκινεί από το 0. Το πεδίο αυτό χρησιμοποιείται από ένα παραλήπτη για την υλοποίηση αξιόπιστης παράδοσης σε επίπεδο stream, καθώς χρειάζεται να ανασυνθέσει το συνολικό ρεύμα δεδομένων στη σωστή σειρά, πριν το διαβιβάσει στην εφαρμογή προορισμού.

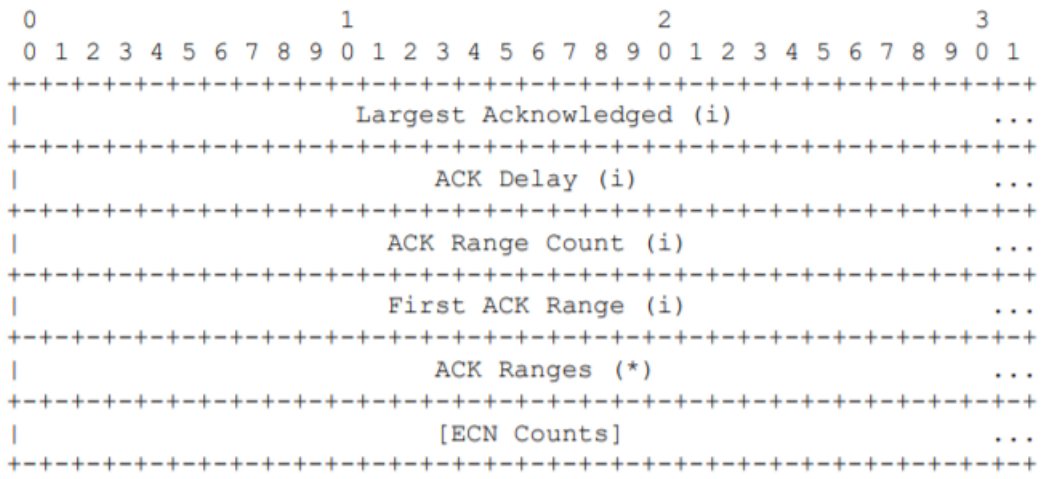
- **Length**: Καθορίζει το μήκος (σε bytes) των δεδομένων που μεταφέρονται μέσα στο πλαίσιο.

- **Δεδομένα Stream**: Εδώ περιέχονται τα δεδομένα εφαρμογής που θα διαβιβαστούν στο stream που υποδεικνύει ο αριθμός stream ID.

Ένα πακέτο QUIC μπορεί να περιέχει περισσότερα του ενός πλαίσια STREAM, τα οποία να ανήκουν σε διαφορετικά streams. Επιτυγχάνεται έτσι η επιθυμητή πολύπλεξη σε επίπεδο stream (stream multiplexing). Ο σχεδιασμός αυτός επιλύει το πρόβλημα του Head of Line Blocking από το οποίο υπέφερε το TCP. Σε περίπτωση απώλειας κάποιου πακέτου QUIC, «παγώνουν» (blocked) μόνο τα streams στα οποία ανήκαν τα πλαίσια STREAM που περιέχονταν στο χαμένο πακέτο. Τα υπόλοιπα streams μπορούν να συνεχίσουν κανονικά τη λειτουργία τους, αφού δεν έχουν επηρεαστεί από την απώλεια πακέτου. Συνεπώς, από τη στιγμή που κάθε QUIC stream μεταφέρει ουσιαστικά ένα διαφορετικό HTTP object, το πρωτόκολλο QUIC μειώνει το συνολικό χρόνο που απαιτείται για την εμφάνιση μιας ιστοσελίδας. Ωστόσο, χρειάζεται ιδιαίτερη προσοχή στον αριθμό των διαφορετικών stream που ενθυλακώνονται στο ίδιο πακέτο. Σε περίπτωση όπου ένα πακέτο QUIC περιέχει δεδομένα από όλα τα streams, τότε η απώλειά του θα «παγώσει» όλα τα streams. Επομένως, το πρωτόκολλο QUIC οφείλει να περιορίζει των αριθμό των διαφορετικών streams που εισάγονται στο ίδιο πακέτο QUIC, διατηρώντας παράλληλα ικανοποιητική ρυθμαπόδοση μετάδοσης. Συγκεντρωτικά, ένα QUIC stream εγγυάται την αξιόπιστη παράδοση δεδομένων και άρα παρουσιάζει παρόμοια λειτουργικότητα με μία σύνδεση TCP. Συνεπώς, μία σύνδεση QUIC που χρησιμοποιεί *N* streams, αντιστοιχεί σε *N* ξεχωριστές συνδέσεις TCP.

3.4.3 Πλαίσιο ACK

Ένας παραλήπτης αποστέλλει πλαίσια ACK προκειμένου να ενημερώσει τον αποστολέα για τα πακέτα QUIC που έχει λάβει και επεξεργαστεί επιτυχώς. Τα πλαίσια ACK είναι αντίστοιχα με τα τμήματα επιβεβαίωσης ACK του TCP, υπό την έννοια ότι επιβεβαιώνουν στον αποστολέα τους αριθμούς των πακέτων QUIC (packet numbers) που έχουν ληφθεί σωστά. Οι επιβεβαιώσεις του πρωτοκόλλου QUIC έχουν παρόμοια χαρακτηριστικά με τις επιλεκτικές επιβεβαιώσεις (SACK) του TCP, καθώς σε ένα πλαίσιο ACK περιέχονται ένα ή περισσότερα εύρη από διαδοχικούς αριθμούς πακέτων που επιβεβαιώνονται (ACK ranges). Ένα πλαίσιο ACK απεικονίζεται στο Σχήμα 3.6 και περιέχει τα ακόλουθα πεδία:



Σχήμα 3.6 *Η δομή του πλαισίου ACK* [34]

- **Largest Acknowledged**: Το πεδίο αυτό περιέχει τον μεγαλύτερο αριθμό πακέτου που επιβεβαιώνει ο παραλήπτης. Καθώς οι αριθμοί πακέτων που εκχωρούνται στα πακέτα QUIC αυξάνονται μονότονα, η τιμή αυτού του πεδίου περιέχει συνήθως το μεγαλύτερο αριθμό πακέτου που έχει λάβει ο παραλήπτης, ακριβώς πριν την αποστολή του πλαισίου ACK.

- **Καθυστέρηση ACK**: Το πεδίο αυτό περιέχει τη χρονική διαφορά σε microseconds, μεταξύ της χρονικής στιγμής που στάλθηκε αυτό το πλαίσιο ACK και της χρονικής στιγμής λήψης του πακέτου με τον μεγαλύτερο έως τότε αριθμό πακέτου (όπως αυτός περιέχεται στο πεδίο Largest Acknowledged). Προκειμένου να εξαχθεί η σωστή τιμή της καθυστέρησης ACK, η τιμή αυτού του πεδίου μετασχηματίζεται ως εξής:

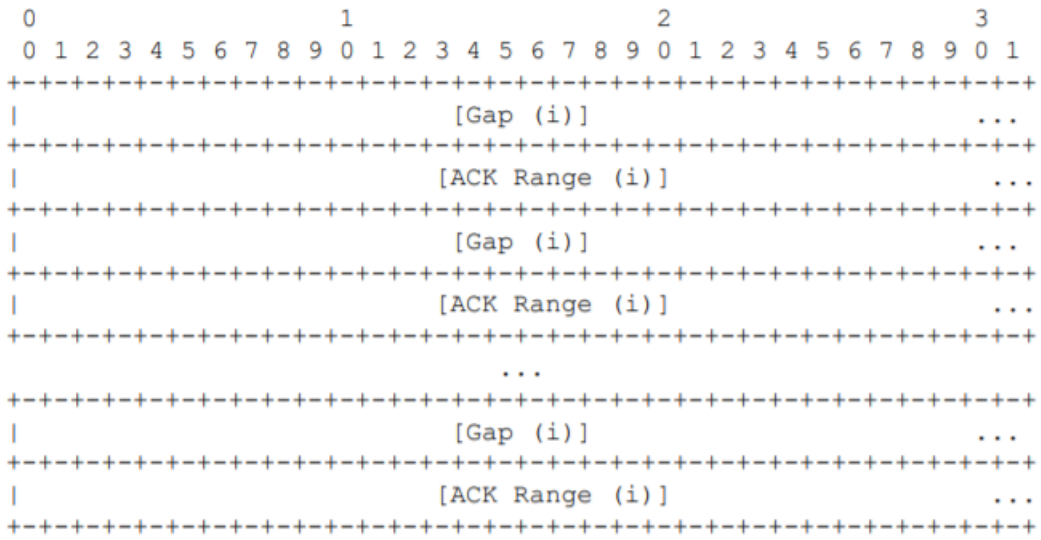
$$(ACK\ Delay)_{final} = ACK\ Delay \cdot 2^{ack\_delay\_exponent} \quad (3.2)$$

  Η τιμή της μεταβλητής *ack_delay_exponent* καθορίζεται κατά την εγκατάσταση της σύνδεσης, με προεπιλεγμένη την τιμή 3. Η τιμής της καθυστέρησης ACK χρησιμοποιείται για ακριβέστερη εκτίμηση του χρόνου RTT, όπως αναλύεται στην §3.5.

- **Πλήθος ACK Ranges**: Υποδεικνύει το πλήθος από κενά (gaps) και εύρη από διαδοχικούς αριθμούς πακέτων (ACK Ranges), που περιέχονται μέσα στο πλαίσιο ACK. Η ανάλυση αυτών των δύο εννοιών ακολουθεί στη συνέχεια.

- **Πρώτο ACK Range**: Το πεδίο αυτό δηλώνει τον αριθμό από συνεχόμενα πακέτα που επιβεβαιώνονται και προηγούνται του πακέτου με το μεγαλύτερο αριθμό. Πρακτικά, το πεδίο αυτό περιέχει το πρώτο εύρος από αριθμούς επιβεβαιωμένων πακέτων. Ο μικρότερος αριθμός πακέτου που επιβεβαιώνεται μέσα στο πρώτο εύρος βρίσκεται αν αφαιρεθεί από τον μεγαλύτερο αριθμό πακέτου (Largest Acknowledged) η τιμή αυτού του πεδίου. Για παράδειγμα, ας υποτεθεί ότι ο μεγαλύτερος αριθμός πακέτου που επιβεβαιώνεται ισούται με 6, ενώ το πρώτο ACK range έχει την τιμή 3. Τότε, τα πακέτα με αριθμούς 3,4 και 5 που προηγούνται του πακέτου με αριθμό 6, επιβεβαιώνονται επίσης.

Ακολουθώντας παρόμοια λογική, το τελευταίο πεδίο **ACK Ranges** αποτελείται από δύο εναλλασσόμενα είδη τιμών, οι οποίες είτε εκφράζουν ένα πλήθος από διαδοχικά μη επιβεβαιωθέντα πακέτα (Gap), είτε εκφράζουν ένα πλήθος από διαδοχικά επιβεβαιωθέντα πακέτα (ACK Range). Η μορφή του πεδίου αυτού δίνεται στο Σχήμα 3.7.



Σχήμα 3.7 *Η μορφή του πεδίου ACK Ranges ενός πλαισίου ACK* [34]

Κάθε πεδίο ACK Range επιβεβαιώνει ένα συνεχόμενο εύρος από αριθμούς πακέτων σε φθίνουσα σειρά. Για το σκοπό αυτό, η τιμή του υποδεικνύει τον αριθμό των πρόσθετων πακέτων που επιβεβαιώνονται, πέραν του πακέτου με το μεγαλύτερο αριθμό για αυτό το εύρος. Συνεπώς, ο ελάχιστος αριθμός πακέτου που επιβεβαιώνεται μέσα σε ένα ACK Range, υπολογίζεται ως εξής:

$$\underbrace{smallest\_packet\_number}_{in\ an\ ACK\ Range} = \underbrace{largest\_packet\_number}_{in\ an\ ACK\ Range} - Ack\_Range\_value \qquad (3.3)$$

Επομένως, ένα ACK Range επιβεβαιώνει όλα τα πακέτα μεταξύ του μικρότερου (*smallest*) και του μεγαλύτερου (*largest*) αριθμού πακέτου που περιέχονται μέσα σε αυτό, συμπεριλαμβανομένου του τελευταίου.

Αντιθέτως, κάθε πεδίο Gap υποδεικνύει ένα εύρος από αριθμούς πακέτων που δεν έχουν ακόμα επιβεβαιωθεί. Το πλήθος των πακέτων μέσα σε ένα τέτοιο εύρος, είναι κατά ένα μεγαλύτερο από την τιμή του πεδίου Gap. Δεδομένου του ελάχιστου αριθμού πακέτου (*previous_smallest)* του πεδίου ACK Range, που βρίσκεται ακριβώς πάνω από ένα πεδίο Gap, μπορεί να υπολογιστεί ο μέγιστος αριθμός πακέτου του επόμενου ACK Range ως εξής:

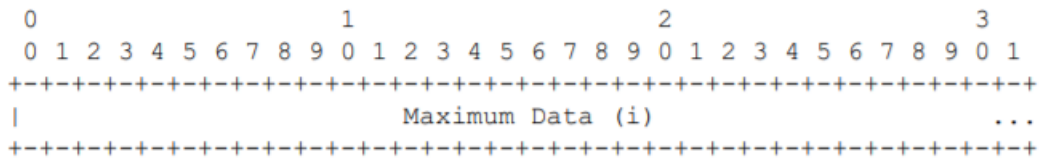
$$next\_largest = previous\_smallest - Gap\_value - 2 \quad (3.4)$$

Συνολικά, με την ανωτέρω διαδικασία, ξεκινώντας από το μεγαλύτερο αριθμό πακέτου που μεταφέρει το πλαίσιο ACK (Largest Acknowledged) και πραγματοποιώντας διαδοχικές αφαιρέσεις, καθίσταται δυνατή η εύρεση όλων των packet numbers που επιβεβαιώνονται από αυτό το πλαίσιο ACK, καθώς και των packet numbers που δεν έχουν ακόμα επιβεβαιωθεί. Η υλοποίηση του μηχανισμού αξιόπιστης παράδοσης του πρωτοκόλλου QUIC βασίζεται πάνω στα πλαίσια ACK, όπως θα εξηγηθεί στην §3.5.

3.4.4 Πλαίσιο MAX_DATA

Το πλαίσιο MAX_DATA χρησιμοποιείται για έλεγχο ροής, προκειμένου να ενημερωθεί ο αποστολέας για τη μέγιστη ποσότητα δεδομένων που μπορούν να σταλούν μέσα στη σύνδεση QUIC. Όλα τα δεδομένα που στέλνονται μέσα σε πλαίσια STREAM προσμετρώνται σε αυτό το όριο. Το πλαίσιο MAX_DATA περιέχει το εξής πεδίο:

- **Maximum Data**: Η μέγιστη συνολική επιτρεπτή ποσότητα δεδομένων (σε bytes) που μπορεί να στείλει ο αποστολέας σε μία σύνδεση QUIC.



Σχήμα 3.8 *Η δομή του πλαισίου MAX_DATA* [34]

Αντίστοιχα, ένα πλαίσιο MAX_STREAM_DATA δηλώνει την μέγιστη ποσότητα δεδομένων που μπορεί να στείλει ο αποστολέας μέσα σε ένα συγκεκριμένο QUIC stream.

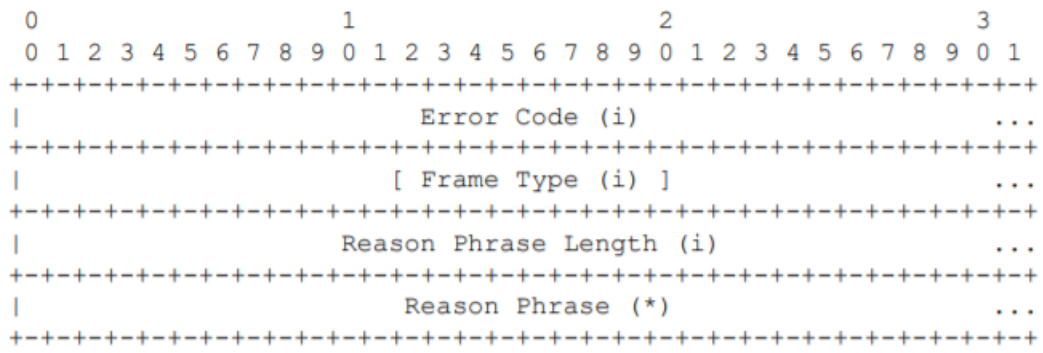
3.4.5 Πλαίσιο CONNECTION_CLOSE

Το πλαίσιο CONNECTION_CLOSE σηματοδοτεί τον τερματισμό της σύνδεσης. Τα βασικά πεδία που περιέχονται σε αυτό τον τύπο πλαισίου είναι τα εξής:

- **Κωδικός Σφάλματος**: Υποδεικνύει την αιτία τερματισμού της σύνδεσης. Σε περίπτωση κανονικού τερματισμού, χρησιμοποιείται η ετικέτα NO_ERROR.

- **Τύπος Πλαισίου**: Αν η σύνδεση τερματίστηκε από σφάλμα του πρωτοκόλλου QUIC, τότε στο πεδίο αυτό περιέχεται ο τύπος του πλαισίου που προκάλεσε το σφάλμα.

- **Επεξήγηση**: Σύντομη αιτιολόγηση σε μορφή κειμένου, για την αιτία τερματισμού.



Σχήμα 3.9 *Η δομή του πλαισίου CONNECTION_CLOSE* [34]

3.5 Αξιόπιστη Μεταφορά Δεδομένων QUIC

Το πρωτόκολλο QUIC αποτελεί κατά βάση ένα πρωτόκολλο μεταφοράς, επομένως είναι απαραίτητη η υλοποίηση μίας υπηρεσίας αξιόπιστης παράδοσης δεδομένων. Το QUIC δανείζεται βασικές ιδέες από τη συσσωρευμένη εμπειρία χρήσης του TCP. Ωστόσο, εισάγει και τα δικά του νέα χαρακτηριστικά. Ο μηχανισμός αξιόπιστης παράδοσης ακολουθεί ορισμένες βασικές αρχές, οι οποίες περιγράφονται συνοπτικά ως εξής:

- Σε κάθε πακέτο QUIC ανατίθεται ένας αριθμός πακέτου (packet number). Οι αριθμοί πακέτων αυξάνονται μονοτονικά και δεν επαναλαμβάνονται στη διάρκεια ζωής μιας σύνδεσης QUIC.

- Τα πακέτα QUIC μπορούν να περιέχουν πολλαπλά πλαίσια διαφορετικού τύπου. Το QUIC εξασφαλίζει ότι δεδομένα ή πλαίσια, για τα οποία απαιτείται αξιόπιστη παράδοση, είτε θα επιβεβαιωθούν με ένα πλαίσιο ACK, είτε θα θεωρηθούν ως χαμένα και θα αναμεταδοθούν κατάλληλα μέσα σε νέα πακέτα QUIC.

- Πακέτα QUIC που μεταφέρουν πλαίσια CRYPTO είναι κρίσιμης σημασίας για την εγκατάσταση της σύνδεσης QUIC (βλ. §3.7) και πρέπει να επιβεβαιώνονται όσο το δυνατό συντομότερα.

- Αντίστοιχα με το TCP, το QUIC διατηρεί ένα παράθυρο συμφόρησης (congestion window), προκειμένου να ελέγχει τον αριθμό των bytes που έχουν σταλεί, αλλά δεν έχουν ακόμη επιβεβαιωθεί (in flight). Πακέτα QUIC που περιέχουν μόνο πλαίσια ACK ή CONNECTION_CLOSE δεν επηρεάζουν τον υπολογισμό του παραθύρου συμφόρησης.

3.5.1 Διαφορές μεταξύ QUIC και TCP

Ένα πρόβλημα που αντιμετωπίζει το TCP είναι το ότι ταυτίζει τη σειρά μετάδοσης τμημάτων στον αποστολέα με τη σειρά παράδοσης των τμημάτων στον παραλήπτη. Υπενθυμίζεται ότι ο μηχανισμός αξιόπιστης παράδοσης του TCP χρησιμοποιεί αριθμούς ακολουθίας για την αρίθμηση των τμημάτων του αποστολέα, τα οποία επιβεβαιώνει ο παραλήπτης μέσω των αριθμών επιβεβαιώσεων. Ωστόσο, σε περίπτωση απώλειας ενός πακέτου, ο αποστολέας θα αναμεταδώσει το χαμένο τμήμα με τον ίδιο αριθμό ακολουθίας. Το γεγονός αυτό οδηγεί σε ορισμένες ασάφειες, οι οποίες προσθέτουν

πολυπλοκότητα στους μηχανισμούς του TCP. Για παράδειγμα, ας υποτεθεί ότι ένας αποστολέας αποστέλλει ένα τμήμα με αριθμό ακολουθίας ίσο με 100, για το οποίο δέχεται τρία διπλότυπα ACK, γεγονός που υποδεικνύει ότι το τμήμα έχει χαθεί. Ο αποστολέας θα αναμεταδώσει το ίδιο ακριβώς τμήμα με αριθμό ακολουθίας 100. Έστω, λοιπόν, ότι, σε μεταγενέστερη χρονική στιγμή, ο αποστολέας λαμβάνει ένα τμήμα ACK, το οποίο επιβεβαιώνει την ορθή λήψη του τμήματος με αριθμό ακολουθίας 100. Στην περίπτωση αυτή, ο αποστολέας δεν μπορεί να διακρίνει αν η επιβεβαίωση στάλθηκε για το αρχικό τμήμα, το οποίο μπορεί να έφθασε καθυστερημένα στον παραλήπτη, ή αν το αρχικό τμήμα όντως χάθηκε και η επιβεβαίωση στάλθηκε για το τμήμα που αναμεταδόθηκε. Ως εκ τούτου, μπορεί να εκτιμάται λανθασμένα ο χρόνος RTT και το TCP να κάνει χρήση άσκοπων αναμεταδόσεων.

Το QUIC επιλύει το ανωτέρω πρόβλημα, διαχωρίζοντας τη σειρά μετάδοσης δεδομένων στον αποστολέα από τη σωστή σειρά παραλαβής δεδομένων από τον παραλήπτη. Το QUIC χρησιμοποιεί αριθμούς πακέτων, οι οποίοι υποδεικνύουν αποκλειστικά την σειρά μετάδοσης των πακέτων QUIC. Αν ένα πακέτο QUIC έχει μεγαλύτερο αριθμό πακέτου από ένα άλλο, αυτό σημαίνει ότι το πακέτο έχει σταλεί σε μεταγενέστερη χρονική στιγμή. Για τα δεδομένα εφαρμογής που μεταφέρονται εντός των πακέτων QUIC με πλαίσια STREAM, η σωστή σειρά παράδοσης καθορίζεται από το πεδίο Offset ενός πλαισίου STREAM, όπως παρουσιάστηκε αναλυτικά στην §3.4.2. Όταν ένα πακέτο QUIC χαθεί, το πρωτόκολλο QUIC συγκεντρώνει τα απαραίτητα πλαίσια που πρέπει να αναμεταδοθούν, τα τοποθετεί σε ένα νέο πακέτο QUIC με καινούριο αριθμό πακέτου και το αποστέλλει στον παραλήπτη. Με αυτό τον τρόπο, όταν ο αποστολέας λάβει μία επιβεβαίωση, μπορεί να αναγνωρίσει αν αυτή αφορά το αρχικό πακέτο QUIC ή το πακέτο QUIC που αναμεταδόθηκε, καθώς τα δύο αυτά πακέτα διαθέτουν διαφορετικό αριθμό πακέτου QUIC. Ως αποτέλεσμα, το QUIC μπορεί να διατηρεί ακριβέστερη εκτίμηση του χρόνου RTT, ενώ διευκολύνεται επίσης η υλοποίηση του μηχανισμού της ταχείας αναμετάδοσης (Fast Retransmit).

Τέλος, οι επιβεβαιώσεις στο QUIC ακολουθούν τη λογική των επιλεκτικών επιβεβαιώσεων (SACK) του TCP, με την έννοια ότι ένα πλαίσιο ACK, όπως αναλύθηκε στην §3.4.3, περιέχει πολλαπλά εύρη από αριθμούς πακέτων οι οποίοι επιβεβαιώνονται (ACK Ranges). Αντίθετα, ενδιάμεσοι αριθμοί πακέτων που δεν επιβεβαιώνονται γίνονται αντιληπτοί ως κενά (Gaps). Η αξιοποίηση αυτής της πληροφορίας από το QUIC οδηγεί σε μεγαλύτερη ταχύτητα ανάκαμψης από απώλειες, καθώς ο αποστολέας γνωρίζει επακριβώς τα πακέτα που έχουν χαθεί, αποφεύγοντας έτσι τις άσκοπες αναμεταδόσεις.

3.5.2 Εκτίμηση Χρόνου RTT

Στην §2.4.3 παρουσιάστηκε ο μηχανισμός που χρησιμοποιεί το TCP, προκειμένου να διατηρεί μία εκτίμηση του χρόνου RTT της σύνδεσης. Η εκτίμηση του RTT είναι απαραίτητη, καθώς με βάση αυτή ορίζονται χρονομετρητές για τα διαστήματα λήξης χρόνου αναμετάδοσης πακέτων. Καθώς δεν αποτελεί εξαίρεση στον κανόνα, το QUIC βασίζεται στο μηχανισμό του TCP για την εκτίμηση του χρόνου RTT, με ορισμένες διαφορές που θα παρουσιαστούν στη συνέχεια. Το QUIC λαμβάνει δείγματα του χρόνου RTT, με τα οποία υπολογίζει τις εξής μεταβλητές: την ελάχιστη τιμή RTT που έχει παρατηρηθεί σε μία σύνδεση QUIC (*min_rtt)*, μία εκθετικά σταθμισμένη μέση τιμή

(EWMA) των δειγμάτων RTT (*smoothed_rtt)* και μία μέση απόκλιση ως μέτρο της διακύμανσης των δειγμάτων RTT (*rtt_var*).

Αρχικά, όταν το QUIC λαμβάνει ένα πλαίσιο ACK, ελέγχει αν ο μεγαλύτερος αριθμός πακέτου, που περιέχεται στο πεδίο Largest Acknowledged, επιβεβαιώνεται πρώτη φορά. Στην περίπτωση όπου αυτό ισχύει, το QUIC παράγει ένα δείγμα του χρόνου RTT (*latest_rtt*), ως τη χρονική διαφορά μεταξύ της στιγμής που παραλαμβάνεται το πλαίσιο ACK και της στιγμής που στάλθηκε το πακέτο με τον μεγαλύτερο αριθμό:

$$latest\_rtt = ack\_time - send\_time\_of\_largest\_packet\_acked \quad (3.5)$$

Για τα δείγματα RTT που λαμβάνονται, το QUIC αποθηκεύει την ελάχιστη τιμή που παρατηρεί. Κατόπιν, τα δείγματα RTT δεν χρησιμοποιούνται αυτούσια στον υπολογισμό της μεταβλητής *smoothed_rtt*. Όπως αναφέρθηκε στην §3.4.3, ένα πλαίσιο ACK διαθέτει το πεδίο ACK Delay, το οποίο εκφράζει το χρονικό διάστημα κατά το οποίο καθυστέρησε η αποστολή του πλαισίου ACK από την πλευρά του παραλήπτη. Η καθυστέρηση αυτή, που συνήθως αναφέρεται και ως χρόνος επεξεργασίας, δεν αποτελεί τμήμα της χρονικής καθυστέρησης που εισάγει το δίκτυο. Επομένως, προκειμένου τα δείγματα RTT να αποτελούν ακριβέστερη αναπαράσταση της χρονικής καθυστέρησης που εισάγει το δίκτυο, από κάθε δείγμα RTT αφαιρείται ο χρόνος επεξεργασίας στον παραλήπτη, όπως αυτός υπολογίζεται με βάση το πεδίο ACK Delay. Ωστόσο, στην περίπτωση όπου το προσαρμοσμένο δείγμα RTT είναι μικρότερο από την ελάχιστη τιμή του RTT που έχει παρατηρηθεί καθ' όλη τη διάρκεια της σύνδεσης, το δείγμα RTT δεν θα αλλάξει τιμή. Συνεπώς, η διαδικασία προσαρμογής των δειγμάτων RTT, με βάση την τιμή του πεδίου Ack Delay, συνοψίζεται ως εξής:

$$\begin{cases} if\ (latest\_rtt - ack\_delay) < \min\_rtt \Rightarrow adjusted\_rtt = latest\_rtt \\ else \Rightarrow adjusted\_rtt = latest\_rtt - ack\_delay \end{cases} \quad (3.6)$$

Από αυτό το σημείο και ύστερα, το QUIC χρησιμοποιεί τις ίδιες εξισώσεις με το TCP, για τον υπολογισμό των μεταβλητών *smoothed_rtt* και *rtt_var*, ως εξής:

$$\begin{cases} smoothed\_rtt = (1-a)\cdot smoothed\_rtt + a \cdot adjusted\_rtt \\ rtt\_var = (1-\beta)\cdot rtt\_var + \beta \cdot |adjusted\_rtt - smoothed\_rtt| \end{cases} \quad (3.7)$$

Οι προεπιλεγμένες τιμές για τα α, β είναι 0.125 και 0.25 αντίστοιχα.

3.5.3 Ανίχνευση Απωλειών

Ένας αποστολέας QUIC χρησιμοποιεί πλαίσια ACK, με σκοπό να ανιχνεύσει τα χαμένα πακέτα. Αν ένα πακέτο χαθεί, το QUIC επιχειρεί να ανακάμψει από αυτήν την απώλεια αναμεταδίδοντας τα χαμένα πλαίσια μέσα σε ένα καινούριο πακέτο QUIC. Σύμφωνα με τις προδιαγραφές του QUIC, ένα πακέτο θεωρείται χαμένο, αν ικανοποιεί τις εξής προϋποθέσεις [35]:

- Το πακέτο έχει μεταδοθεί, δεν έχει επιβεβαιωθεί και ένα άλλο πακέτο με μεγαλύτερο packet number έχει επιβεβαιωθεί.

- Ο αριθμός πακέτου του είναι κατά *kPacketThreshold* μικρότερος από τον αριθμό ενός επιβεβαιωμένου πακέτου ή έχει σταλεί αρκετό χρόνο πριν στο παρελθόν.

Η επιβεβαίωση ενός πακέτου με μεγαλύτερο packet number αποτελεί ένδειξη ότι ένα πακέτο που έχει σταλεί αργότερα από το υπό εξέταση πακέτο έχει παραληφθεί σωστά. Η δεύτερη συνθήκη ορίζει πρακτικά δύο κατώφλια, τα οποία προσθέτουν στο QUIC κάποια ανοχή στην αναδιάταξη πακέτων και χρησιμοποιούνται ως εξής:

- **Κατώφλι Αναδιάταξης Πακέτων**

  Το πρωτόκολλο QUIC ορίζει ένα κατώφλι αναδιάταξης πακέτων, που συμβολίζεται ως *kPacketThreshold*. Έστω ότι ένας αποστολέας έχει στείλει ένα πακέτο με αριθμό *N*, για το οποίο δεν έχει λάβει ακόμα επιβεβαίωση. Σε περίπτωση που λάβει μία επιβεβαίωση για κάποιο άλλο πακέτο, το οποίο έχει αριθμό τουλάχιστον ίσο με *N+kPacketThreshold*, τότε ο αποστολέας θα θεωρήσει ότι το πακέτο με αριθμό *N* έχει χαθεί. Η προτεινόμενη τιμή για το κατώφλι *kPacketThreshold* είναι 3. Ο μηχανισμός αυτός είναι παρόμοιος με τα duplicate ACK του TCP. Όταν ένας αποστολέας TCP λάβει 3 duplicate τμήματα ACK, θεωρεί ότι το τμήμα με αριθμό ακολουθίας ίσο με τον αριθμό επιβεβαίωσης των duplicate ACK έχει χαθεί. Η μεταβλητή *kPacketThreshold* ονομάζεται κατώφλι αναδιάταξης πακέτων, γιατί επιτρέπει την αναδιάταξη πακέτων μέσα στο δίκτυο με μέγιστο «βάθος» αναδιάταξης ίσο με (*kPacketThreshold-1*). Αυτό σημαίνει ότι, αν ένα πακέτο με αριθμό *N* δεν έχει επιβεβαιωθεί, η λήψη επιβεβαιώσεων για τα πακέτα με αριθμούς {*N*, *N+1*,…, *N+kPacketThreshold-1*} δεν θα προκαλέσει τον χαρακτηρισμό του πακέτου *N* ως χαμένο. Επομένως, το QUIC είναι ανεκτικό στην αναδιάταξη πακέτων, των οποίων όμως οι αριθμοί ανήκουν σε αυτό το εύρος τιμών. Το μειονέκτημα αυτού του μηχανισμού είναι ότι, σε περιπτώσεις δικτύων που εμφανίζουν υψηλό βαθμό αναδιάταξης αλλά τα πακέτα παραδίδονται τελικά σωστά στον παραλήπτη, θα υπάρξει μείωση αποδοτικότητας γιατί αρκετές ανιχνεύσεις απωλειών πακέτων θα είναι ψευδείς (spurious).

- **Κατώφλι Χρόνου**

  Μόλις ένας αποστολέας λαμβάνει μία επιβεβαίωση για ένα πακέτο, το πρωτόκολλο QUIC χαρακτηρίζει ως χαμένα όλα τα πακέτα που δεν έχουν επιβεβαιωθεί, και έχουν σταλεί σε προγενέστερη χρονική στιγμή από τη στιγμή λήψης της επιβεβαίωσης. Η χρονική διαφορά των δύο αυτών στιγμών πρέπει να είναι τουλάχιστον ίση με ένα χρονικό κατώφλι $Time_{thr}$, το οποίο δίνεται από την εξίσωση:

  $$Time_{thr} = kTimeThreshold \cdot \max(smoothed\_rtt, latest\_rtt) \quad (3.8)$$

  Η προτεινόμενη για τιμή για τη μεταβλητή *kTimeThreshold* είναι 9/8.

Συνολικά, το QUIC καθορίζει τους δύο τρόπους που μόλις παρουσιάστηκαν, με τους οποίους γίνεται η ανίχνευση των απωλειών πακέτων. Καίτοι το QUIC εισάγει ορισμένα νέα χαρακτηριστικά, η ανίχνευση απωλειών εξακολουθεί να βασίζεται σε επιβεβαιώσεις και «λήξεις» χρόνου, όπως ακριβώς συνέβαινε και στο πρωτόκολλο TCP.

3.6 Έλεγχος Συμφόρησης

Ο έλεγχος συμφόρησης που προτείνεται για το πρωτόκολλο QUIC στο [35] βασίζεται στην έκδοση NewReno του TCP. Ωστόσο, η προδιαγραφή του QUIC υποστηρίζει τη χρήση και άλλων γνωστών ή νέων αλγορίθμων συμφόρησης. Επειδή το QUIC λειτουργεί πάνω από το UDP, οποιαδήποτε υλοποίηση αλγορίθμου ελέγχου συμφόρησης οφείλει να ακολουθεί τις οδηγίες που περιγράφονται στο [36].

Αντίστοιχα με το TCP, ο αλγόριθμος ελέγχου συμφόρησης του QUIC μεταβάλλει την τιμή του παραθύρου συμφόρησης (*cwnd*), ως απόκριση στη λήψη πλαισίων ACK ή στην ανίχνευση απώλειας πακέτου. Ορίζονται τρεις διακριτές καταστάσεις, στις οποίες μπορεί να μεταπέσει ένας αποστολέας QUIC:

- **Αργή Εκκίνηση**: Κάθε νέα σύνδεση QUIC εκκινεί από το στάδιο της αργής εκκίνησης και εξέρχεται από αυτό όταν ανιχνευθεί η πρώτη απώλεια πακέτου. Κατά την αργή εκκίνηση, όταν το QUIC λάβει ένα πλαίσιο ACK, αυξάνει το παράθυρο συμφόρησης κατά το συνολικό πλήθος bytes που επιβεβαιώθηκαν από το πλαίσιο ACK. Πρακτικά, το παράθυρο συμφόρησης διπλασιάζεται περίπου κάθε RTT, όπως ακριβώς και στην περίπτωση του TCP. Το QUIC εισέρχεται ξανά στη φάση της αργής εκκίνησης όταν ανιχνευθεί στο δίκτυο «επίμονη συμφόρηση», όπως εξηγείται στη συνέχεια.

- **Αποφυγή Συμφόρησης**: Με την ανίχνευση της πρώτης απώλειας πακέτου, το QUIC μεταπίπτει από το στάδιο της αργής εκκίνησης στο στάδιο της αποφυγής συμφόρησης. Στο στάδιο αποφυγής συμφόρησης, το QUIC χρησιμοποιεί την τεχνική AIMD (Additive Increase - Multiplicative Decrease), ώστε να αυξάνει γραμμικά το παράθυρο συμφόρησης. Όταν ανιχνευθεί μία νέα απώλεια, η τιμή του *cwnd* μειώνεται στο μισό, η τιμή του κατωφλίου αργή εκκίνησης (*ssthresh*) τίθεται ίσο με *cwnd* και το QUIC μεταπίπτει στο στάδιο της περιόδου ανάκαμψης.

- **Περίοδος Ανάκαμψης**: Όταν το QUIC εισέλθει στην περίοδο ανάκαμψης, συγκεντρώνει τα απαραίτητα πλαίσια που περιέχονταν στο χαμένο πακέτο, τα ενθυλακώνει σε ένα νέο πακέτο και το μεταδίδει. Το QUIC μεταπίπτει ξανά στο στάδιο της αποφυγής συμφόρησης όταν επιβεβαιωθεί ένα οποιοδήποτε πακέτο το οποίο στάλθηκε κατά το στάδιο της περιόδου ανάκαμψης. Συγκριτικά, το βήμα αυτό διαφέρει στο TCP, καθώς το TCP εξέρχεται από το στάδιο της ταχείας ανάκαμψης, μόνο όταν επιβεβαιωθεί το τμήμα που είχε αρχικά χαθεί. Αυτό επιτρέπει στο QUIC να επανέρχεται ταχύτερα σε σχέση με το TCP στη φάση αποφυγής συμφόρησης, και άρα να επανεκκινεί ταχύτερα τη γραμμική αύξηση του παραθύρου συμφόρησης.

Τέλος, στην περίπτωση όπου το QUIC λάβει ένα πλαίσιο ACK που υποδηλώνει απώλεια όλων των πακέτων που έχουν σταλεί μέσα σε ένα αρκετά μεγάλο χρονικό διάστημα, θεωρείται ότι το δίκτυο βρίσκεται σε κατάσταση «επίμονης συμφόρησης» (persistent congestion). Το χρονικό διάστημα που χρησιμοποιείται για την ανίχνευση επίμονης συμφόρησης δίνεται από την εξίσωση:

$$T_{persistent} = (smoothed\_rtt + 4 \cdot rtt\_var + \max\_ack\_delay) \cdot K \quad (3.9)$$

Η μεταβλητή *max_ack_delay* δηλώνει τη μέγιστη καθυστέρηση που μπορεί δηλωθεί στο πεδίο ACK Delay ενός πλαισίου ACK, ενώ η μεταβλητή K ονομάζεται *kPersistentCongestionThreshold*. Για παράδειγμα, ας υποτεθεί ότι: *smoothed_rtt = 1*, *rtt_var = 0*, *max_ack_delay = 0* και *kPersistentCongestionThreshold = 3*. Αν ένα πακέτο σταλθεί τη χρονική στιγμή *t = 0*, τότε η αλληλουχία πακέτων που απεικονίζεται στο Σχήμα 3.10, θα ανιχνευόταν ως επίμονη συμφόρηση:

t=0	Send Pkt #1 (App Data)
t=1	Send Pkt #2 (PTO 1)
t=3	Send Pkt #3 (PTO 2)
t=7	Send Pkt #4 (PTO 3)
t=8	Recv ACK of Pkt #4

Σχήμα 3.10 *Περίπτωση επίμονης συμφόρησης στο QUIC* [35]

Τα πρώτα τρία πακέτα θεωρούνται χαμένα, αφού τη χρονική στιγμή *t = 8* λαμβάνεται μία επιβεβαίωση για το τέταρτο πακέτο. Η περίοδος συμφόρησης υπολογίζεται ως η χρονική διαφορά των στιγμών αποστολής του πρώτου και του τελευταίου χαμένου πακέτου: $T_{congestion} = 3 - 0 = 3$. Το κατώφλι της επίμονης συμφόρησης είναι: $T_{persistent} = (1 + 0 + 0) \cdot 3 = 3$. Επομένως, επειδή η περίοδος συμφόρησης είναι ίση με το κατώφλι, το QUIC θεωρεί ότι το δίκτυο αντιμετωπίζει επίμονη συμφόρηση. Στην περίπτωση αυτή, το παράθυρο συμφόρησης γίνεται ίσο με το μέγιστο μέγεθος ενός πακέτου QUIC, δηλαδή η μεταβλητή *cwnd* μειώνεται στην ελάχιστη τιμή της, ενώ το QUIC μεταπίπτει στο στάδιο της αργής εκκίνησης. Η αντίδραση αυτή του QUIC είναι λειτουργικά ίδια με την αντίδραση του TCP σε λήξη του χρονομετρητή *TimeoutInterval* (RTO), κατά την οποία το παράθυρο συμφόρησης μειώνεται στην τιμή *cwnd = 1 MSS* και το πρωτόκολλο TCP εισέρχεται στη φάση της αργής εκκίνησης.

3.7 Ασφάλεια Συνδέσεων QUIC

Το QUIC ακολουθεί την τάση του σύγχρονου ψηφιακού κόσμου, όπου απαιτείται ολοένα και περισσότερη ασφάλεια στην επικοινωνία μέσω του Διαδικτύου. Προς αυτήν την κατεύθυνση, ο σχεδιασμός του QUIC ενσωματώνει εξ ορισμού τις λειτουργικότητες του πρωτοκόλλου ασφαλείας TLS 1.3, υπό την έννοια ότι ο χρήστης δεν μπορεί σε καμία περίπτωση να το απενεργοποιήσει. Το QUIC προσφέρει υπηρεσίες ταυτοποίησης και κρυπτογράφησης για την πλειοψηφία των πακέτων (συμπεριλαμβανομένης της επικεφαλίδας) που αποστέλλονται στο δίκτυο, με την εξαίρεση ορισμένων πεδίων τις επικεφαλίδας. Αυτό αποτελεί σημαντική διαφοροποίηση σε σχέση με το TCP, το οποίο κρυπτογραφεί μόνο τα δεδομένα εφαρμογής που μεταφέρει, με τις επικεφαλίδες TCP να παραμένουν ορατές μέσα στο δίκτυο.

3.7.1 Εγκατάσταση Σύνδεσης QUIC

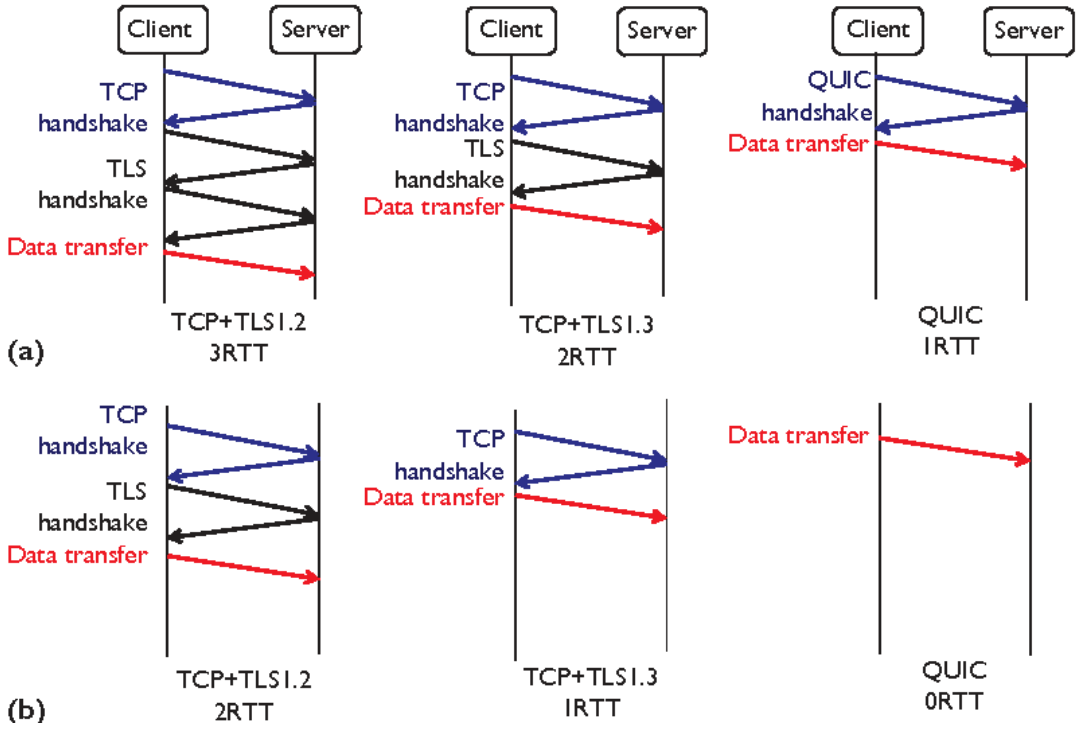
Η εγκατάσταση μίας σύνδεσης QUIC μεταξύ δύο τερματικών γίνεται μέσω χειραψίας, που παρουσιάζει παρόμοια χαρακτηριστικά με την αντίστοιχη του TCP. Στη γενική περίπτωση, ανταλλάσσονται συνολικά τέσσερα δεδομενογράμματα UDP, μέχρι να ολοκληρωθεί πλήρως η χειραψία QUIC, όπως φαίνεται στο Σχήμα 3.11. Κάθε δεδομενόγραμμα UDP μπορεί να περιέχει διαφορετικού τύπου πακέτα QUIC (Initial, Handshake, 1-RTT), τα οποία είναι κρυπτογραφημένα με διαφορετικά κλειδιά. Ωστόσο, συνενώνονται στο ίδιο δεδομενόγραμμα UDP, ώστε να μειωθεί ο αριθμός των δεδομενογραμμάτων UDP που ανταλλάζονται στη χειραψία QUIC.

Σχήμα 3.11 *Η χειραψία QUIC* [37]

Μία χειραψία QUIC αρχίζει με το δεδομενόγραμμα “QUIC Client Hello”, στο οποίο περιέχεται ένα πακέτο QUIC τύπου Initial. Το πακέτο Initial ενθυλακώνει ένα πλαίσιο CRYPTO με το μήνυμα “TLS Client Hello”. Στη συνέχεια, ο εξυπηρετητής αποστέλλει το δεδομενόγραμμα “QUIC Server Hello”, το οποίο περιέχει ένα πακέτο τύπου Initial και ένα πακέτο τύπου Handshake. Το πακέτο Initial περιέχει σε ένα πλαίσιο CRYPTO το μήνυμα “TLS Server Hello”, ενώ το πακέτο Handshake περιέχει πληροφορίες σχετικά με την ταυτοποίηση του εξυπηρετητή (π.χ. server certificate). Έπειτα, ο πελάτης αποστέλλει το δεδομενόγραμμα “Initial Completion”, το οποίο επιβεβαιώνει στον εξυπηρετητή την ολοκλήρωση της χειραψίας από την πλευρά του πελάτη. Επιπλέον, στο στάδιο αυτό ο πελάτης έχει επίσης διαθέσιμο το μυστικό κλειδί (1-RTT key), άρα μπορεί να στείλει το πρώτο πακέτο QUIC με δεδομένα εφαρμογής (1-RTT data). Η χειραψία ολοκληρώνεται με το δεδομενόγραμμα “Handshake Completion”. Ύστερα από αυτό το σημείο, τα δύο άκρα της σύνδεσης ανταλλάσσουν δεδομένα με το κοινό μυστικό κλειδί που παράχθηκε κατά τη χειραψία QUIC.

Όπως έχει ήδη αναφερθεί, το πρωτόκολλο QUIC σχεδιάστηκε ώστε να ελαχιστοποιεί το χρόνο που απαιτείται για την εγκατάσταση της σύνδεσης. Πράγματι, σύμφωνα με τα ανωτέρω, η ροή δεδομένων εφαρμογής εκκινεί από τον πελάτη προς τον εξυπηρετητή, ύστερα από χρονικό διάστημα ίσο με 1 RTT. Στην περίπτωση όπου ο πελάτης έχει επικοινωνήσει ξανά στο παρελθόν με τον εξυπηρετητή, είναι δυνατή η αποστολή δεδομένων εφαρμογής μαζί με το πρώτο δεδομενόγραμμα “QUIC Client Hello”, επιτυγχάνοντας πρακτικά την εγκατάσταση της σύνδεσης σε χρόνο 0 RTT. Η βελτίωση στο χρόνο εγκατάστασης της σύνδεσης που πετυχαίνει το QUIC συγκριτικά με τις

τωρινές υλοποιήσεις του πρωτοκόλλου TCP (σε συνδυασμό με το TLS), απεικονίζεται στο Σχήμα 3.12. Η περίπτωση (a) αντιστοιχεί σε τερματικά που επικοινωνούν μεταξύ τους για πρώτη φορά, ενώ η περίπτωση (b) αντιστοιχεί σε τερματικά, που έχουν επικοινωνήσει ξανά στο παρελθόν (επανεκκινούμενες συνδέσεις).

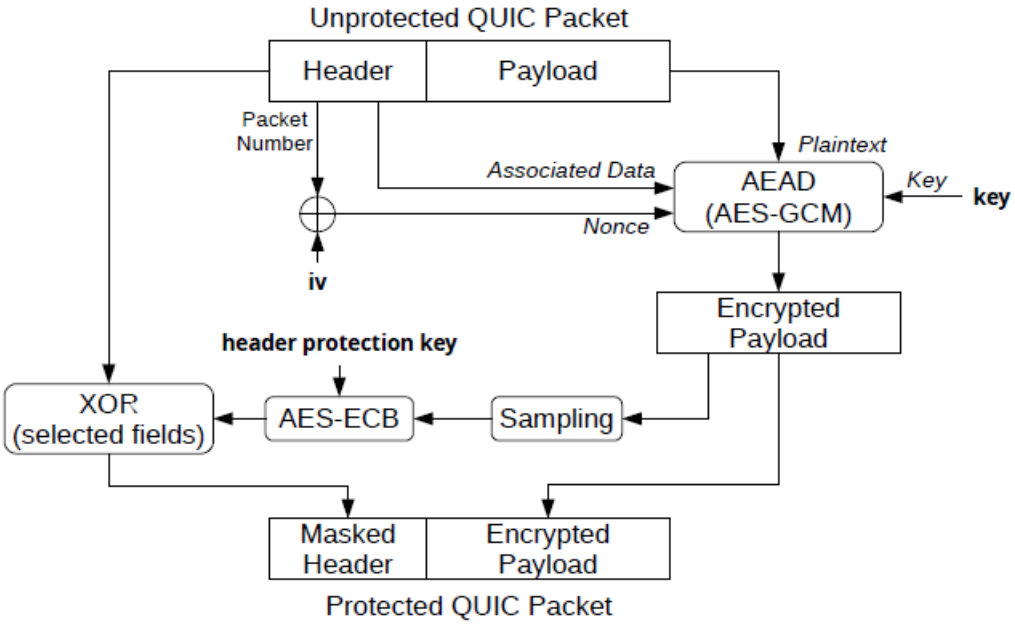


Σχήμα 3.12 *Χρόνος εγκατάστασης σύνδεσης QUIC vs (TCP+TLS)* [38]

3.7.2 Ασφάλεια Πακέτων και Επικεφαλίδων QUIC

Μόλις ο πελάτης και ο εξυπηρετητής ανταλλάξουν το κοινό μυστικό κλειδί που θα χρησιμοποιηθεί για την κρυπτογράφηση των δεδομένων, το φορτίο και ο αριθμός πακέτου κάθε πακέτου QUIC κρυπτογραφούνται με τη διαδικασία που παρουσιάζεται συνοπτικά στο Σχήμα 3.13. Η πλήρης ανάλυσή της μπορεί να βρεθεί στο [39].

Αρχικά, η επικεφαλίδα απομονώνεται από το φορτίο του πακέτου QUIC. Έπειτα, το φορτίο του πακέτου QUIC κρυπτογραφείται σύμφωνα με τον αλγόριθμο κρυπτογράφησης που έχει συμφωνηθεί μεταξύ του αποστολέα και του παραλήπτη κατά την εγκατάσταση της σύνδεσης. Ως είσοδοι στον αλγόριθμο δίνονται το φορτίο του πακέτου, το μυστικό κλειδί και μία τυχαία τιμή (nonce). Στη συνέχεια, δειγματοληπτείται ένα μέρος του παραγόμενου κρυπτοκειμένου (ciphertext), το οποίο εισέρχεται στον αλγόριθμο κρυπτογράφησης της επικεφαλίδας, μαζί με το αντίστοιχο κλειδί. Τέλος, εκτελείται μία πράξη XOR μεταξύ της αρχικής και της κρυπτογραφημένης επικεφαλίδας, που έχει ως αποτέλεσμα την προστασία του πεδίου Packet Number.



Σχήμα 3.13 *Ο μηχανισμός προστασίας των πακέτων QUIC* [40]

TCP	QUIC
Υλοποιημένο στο λειτουργικό σύστημα	Υλοποιημένο στο χώρο χρήστη με χρήση του UDP ως υπόστρωμα
Μοναδική μορφή επικεφαλίδας TCP με κανένα πεδίο κρυπτογραφημένο	Δύο μορφές επικεφαλίδας QUIC με ορισμένα πεδία κρυπτογραφημένα
Αρίθμηση τμημάτων TCP με αριθμούς ακολουθίας, οι οποίοι επαναλαμβάνονται σε περιπτώσεις αναμεταδόσεων	Αρίθμηση πακέτων QUIC με ένα μοναδικό αύξοντα αριθμό, ο οποίος δεν επαναλαμβάνεται
Τα δεδομένα εφαρμογής περιέχονται στα τμήματα TCP	Τα δεδομένα εφαρμογής περιέχονται στα πλαίσια QUIC, που ενθυλακώνονται σε πακέτα QUIC
Μια σύνδεση TCP αναγνωρίζεται μοναδικά από μια τετράδα τιμών (4-tuple multiplexing)	Μια σύνδεση QUIC αναγνωρίζεται μοναδικά από το Connection ID
Επηρεάζεται από αλλαγές στις παραμέτρους του στρώματος IP	Δεν επηρεάζεται από αλλαγές στις παραμέτρους του στρώματος IP
Δεν επεκτείνει την πολύπλεξη των HTTP streams στο στρώμα μεταφοράς	Επεκτείνει την πολύπλεξη των HTTP streams στο στρώμα μεταφοράς, με χρήση των QUIC streams
Ο χρόνος εγκατάστασης σύνδεσης κυμαίνεται από 1RTT έως 3RTT	Ο χρόνος εγκατάστασης σύνδεσης κυμαίνεται από 0RTT έως 1RTT

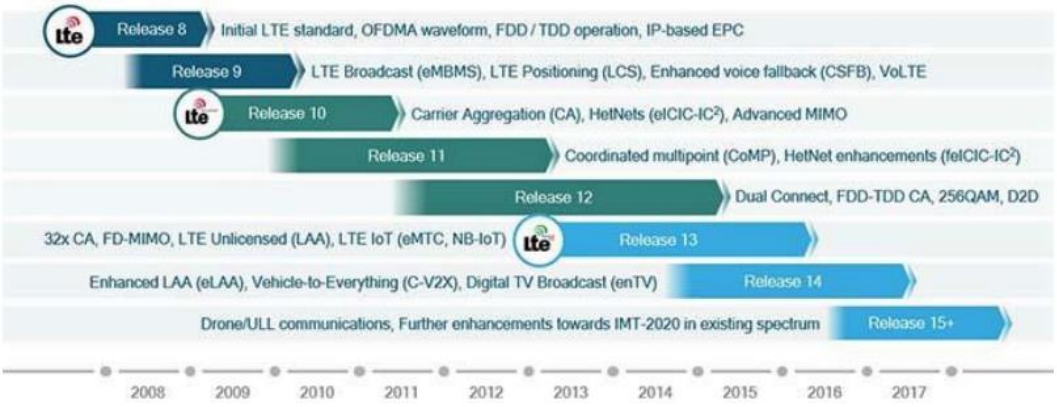
Πίνακας 3.3 *Σχεδιαστικές διαφορές των πρωτοκόλλων TCP και QUIC*

Κεφάλαιο 4. Long Term Evolution (LTE)

Τα συστήματα LTE (Long Term Evolution) αποτελούν την εξέλιξη της τρίτης γενιάς των συστημάτων κινητών επικοινωνιών (3G - UMTS). Η ανάπτυξη των τεχνικών προδιαγραφών των συστημάτων LTE αποτέλεσε έργο της οργάνωσης 3GPP (Third Generation Partnership Project), σε συνεργασία με το ETSI (European Telecommunications Standards Institute), με τις εργασίες να εκκινούν το 2004. Η πρώτη ολοκληρωμένη έκδοση του LTE (Release 8) δημοσιεύτηκε το Μάρτιο του 2009, με επόμενη έκδοση να ακολουθεί λίγους μήνες αργότερα (Release 9) [41]. Η ανάπτυξη του νέου προτύπου LTE ωθήθηκε τόσο από την αύξηση των συνδρομητών κινητής τηλεφωνίας, όσο και από τη δημιουργία νέων εφαρμογών που καταναλώνουν μεγάλο όγκο από mobile data (π.χ. Web Browsing 2.0, mobile video streaming, VoIP). Συνεπώς, το LTE σχεδιάστηκε με γνώμονα την ικανοποίηση νέων, αυστηρότερων απαιτήσεων:

- Υψηλότεροι ρυθμοί μετάδοσης, της τάξης των δεκάδων Mbps.
- Χαμηλή καθυστέρηση από-άκρο-σε-άκρο, της τάξης των λίγων δεκάδων milliseconds.
- Αποδοτικότερη χρησιμοποίηση του διαθέσιμου φάσματος συχνοτήτων, ώστε να υποστηρίζονται ταυτόχρονα περισσότεροι χρήστες.
- Απλούστευση της αρχιτεκτονικής του δικτύου, ώστε να μειωθούν τα λειτουργικά κόστη των τηλεπικοινωνιακών παρόχων.

Τέλος, επειδή ο κύριος στόχος των συστημάτων LTE ήταν η υποστήριξη κινητών υπηρεσιών δεδομένων (mobile data services), υιοθετήθηκε εξ ολοκλήρου η τεχνολογία της μεταγωγής πακέτου αντί της μεταγωγής κυκλώματος που ίσχυε στα δίκτυα 3G. Ως εκ τούτου, τα συστήματα LTE κάνουν αποκλειστική χρήση του πρωτοκόλλου IP για τη διακίνηση της πληροφορίας και αποκαλούνται “all-IP networks”.



Σχήμα 4.1 *Η εξέλιξη των εκδόσεων της 3GPP για τα συστήματα LTE*

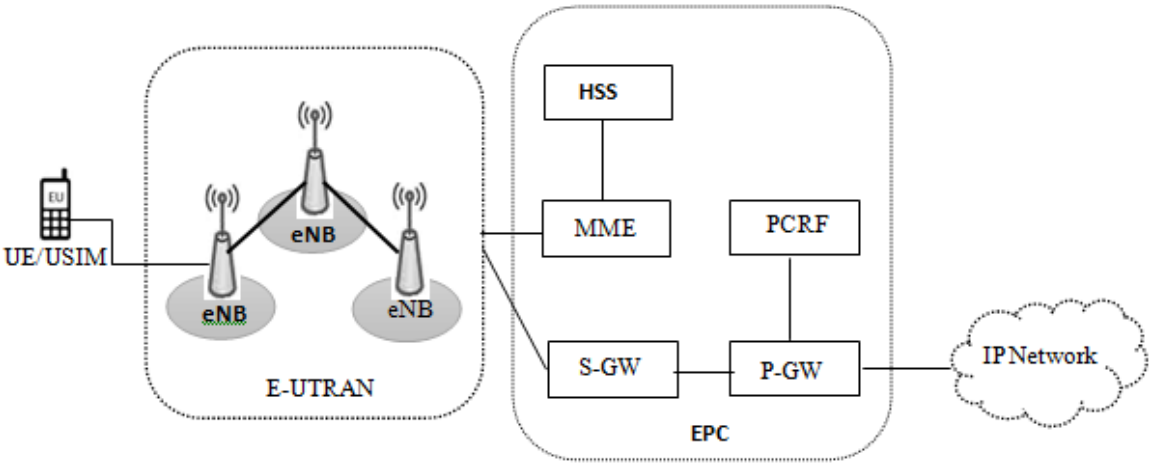
[Πηγή: https://www.qualcomm.com/news/onq/2017/understanding-3gpp-starting-basics]

Η τεχνολογία LTE γνώρισε παγκοσμίως ευρεία αποδοχή, ως η τέταρτη γενιά των συστημάτων κινητών επικοινωνιών (4G). Η 3GPP συνέχισε την εξέλιξη του LTE με την δημοσίευση των εκδόσεων 9 έως 14. Οι εκδόσεις 10 έως 12 χαρακτηρίστηκαν ως LTE -

Advanced (LTE-A), ενώ οι επόμενες εκδόσεις αποκαλούνται LTE - Advanced pro. Οι νέες εκδόσεις δεν επιφέρουν αλλαγές στη βασική αρχιτεκτονική του δικτύου. Ωστόσο, αξιοποιούν νέες τεχνολογίες ή τεχνικές που επιτυγχάνουν υψηλότερους ρυθμούς μετάδοσης. Σήμερα, το LTE αποτελεί την πλέον διαδεδομένη τεχνολογία δικτύου κινητών επικοινωνιών, με ποσοστό γεωγραφικής κάλυψης μεγαλύτερο του 80% σε αρκετές ανεπτυγμένες χώρες [42].

4.1 Αρχιτεκτονική Δικτύου LTE

Σε αντιστοιχία με τα κυψελωτά δίκτυα 2G και 3G, ένα δίκτυο LTE αποτελείται από ένα δίκτυο ραδιοπρόσβασης (Radio Access Network - RAN) και ένα δίκτυο κορμού (Core Network - CN). Το RAN του LTE ονομάζεται E-UTRAN (Evolved UTRAN) και είναι υπεύθυνο για την επικοινωνία μεταξύ των τερματικών και των σταθμών βάσης. Το δίκτυο κορμού καλείται EPC (Evolved Packet Core) και συνδέει τους σταθμούς βάσης με εξωτερικά δίκτυα (π.χ. το Internet). Η συνολική αρχιτεκτονική ενός δικτύου LTE απεικονίζεται στο Σχήμα 4.2.

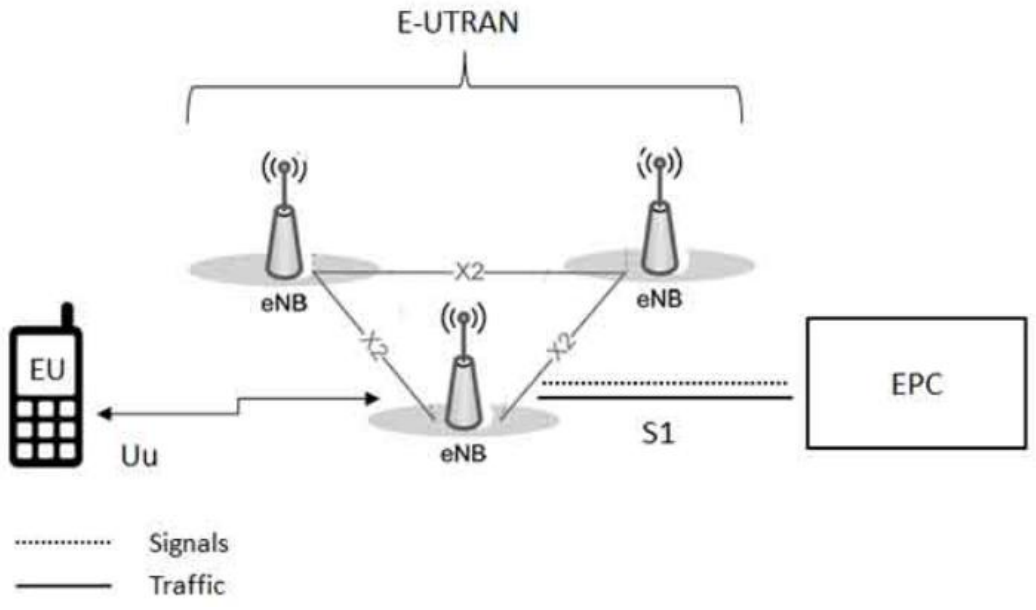


Σχήμα 4.2 *Αρχιτεκτονική δικτύου LTE*

[Πηγή: https://www.researchgate.net/figure/LTE-Network-architecture_fig1_318502441]

4.1.1 Δίκτυο Ραδιοπρόσβασης (E-UTRAN)

Το E-UTRAN αποτελείται από τα τερματικά των χρηστών, που ονομάζονται UE (User Equipment), και από τους σταθμούς βάσης, που ονομάζονται eNB (evolved Node B). Σε κάθε χρονική στιγμή, ένα UE βρίσκεται σε μία κυψέλη του δικτύου και επικοινωνεί με κάποιον eNB μέσω της διεπαφής Uu. Οι eNBs συνδέονται μεταξύ τους με τη διεπαφή X2, ενώ επικοινωνούν με τον EPC μέσω της διεπαφής S1. Η διεπαφή Uu μεταφέρει τα δεδομένα χρηστών και σηματοδοσίας στις κατευθύνσεις Uplink (UL) και Downlink (DL). Η κατεύθυνση UL αντιστοιχεί σε μεταφορά δεδομένων από ένα UE προς έναν eNB ενώ η κατεύθυνση DL το αντίστροφο. Η διεπαφή X2 μεταφέρει κυρίως δεδομένα σηματοδοσίας και προωθεί πακέτα δεδομένων μεταξύ γειτονικών eNBs σε περιπτώσεις μεταπομπής ενός UE. Τέλος, η διεπαφή S1 μεταφέρει δεδομένα χρήστη και μηνύματα ελέγχου μεταξύ του E-UTRAN και του EPC.



Σχήμα 4.3 *Αρχιτεκτονική E-UTRAN*

[Πηγή: https://www.tutorialspoint.com/lte/lte_network_architecture.htm]

Ο eNB αποτελεί το θεμελιώδες τμήμα του δικτύου E-UTRAN όπου υλοποιούνται οι εξής βασικές λειτουργίες:

- Δυναμική κατανομή των ραδιοπόρων στα UEs, που ονομάζεται Διαχείριση Ραδιοπόρων (Radio Resource Management - RRM).

- Έλεγχος των ραδιοκομιστών (radio bearers), οι οποίοι αποτελούν ουσιαστικά μία σύνδεση μεταξύ δύο σημείων, στην οποία το δίκτυο εγγυάται μία συγκεκριμένη ποιότητα υπηρεσίας (Quality of Service - QoS). Για παράδειγμα, ένας ραδιοκομιστής μπορεί να ορίζεται μεταξύ ενός UE και του E-UTRAN, ώστε να προσφέρει στο UE ένα ελάχιστο εγγυημένο ρυθμό μετάδοσης (Guaranteed Bit Rate - GBR).

- Έλεγχος ραδιοπρόσβασης (Radio Admission Control) και έλεγχος κινητικότητας σύνδεσης (Connection Mobility) των UEs.

- Αποστολή μηνυμάτων σηματοδοσίας ελέγχου πάνω από τη διεπαφή.

- Κρυπτογράφηση των δεδομένων σε επίπεδο χρήστη και σε επίπεδο ελέγχου.

- Συμπίεση και κρυπτογράφηση των επικεφαλίδων του πρωτοκόλλου IP.

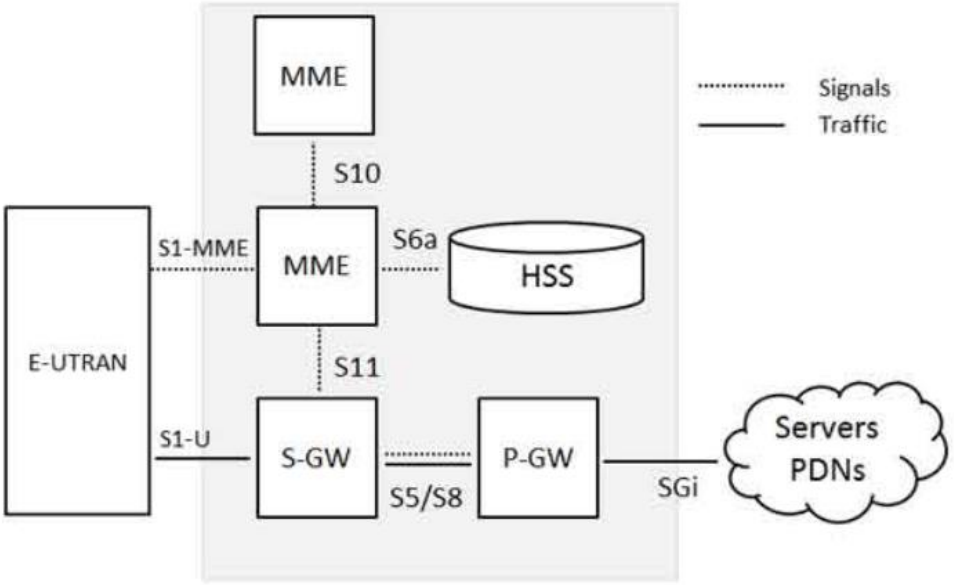
- Χρονοπρογραμματισμός (scheduling) των ραδιοπόρων και μετάδοση μηνυμάτων αναζήτησης (paging) και ευρυεκπομπής (broadcast).

- Συλλογή και αναφορά μετρήσεων για την κινητικότητα και το χρονοπρογραμματισμό των ραδιοπόρων.

- Δρομολόγηση των δεδομένων των χρηστών προς το δίκτυο κορμού EPC.

4.1.2 Δίκτυο κορμού (EPC)

Στις βασικές μονάδες του δικτύου κορμού EPC που απεικονίζονται στο Σχήμα 4.4, συμπεριλαμβάνονται η μονάδα διαχείρισης κινητικότητας (Mobility Management Entity - MME), η πύλη εξυπηρέτησης (Serving Gateway - S-GW), η πύλη δικτύου πακέτων (Packet Gateway - P-GW) και ο οικείος εξυπηρετητής συνδρομητών (Home Subscriber Server - HSS).

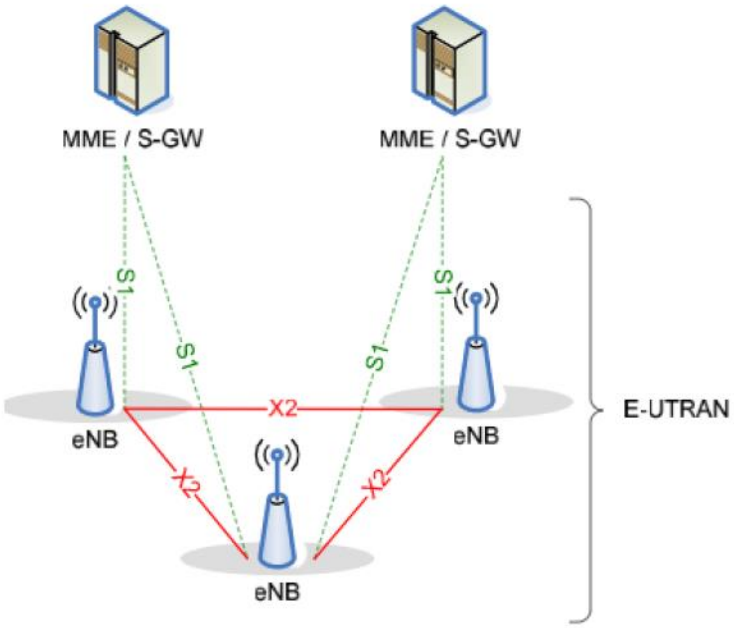


Σχήμα 4.4 *Αρχιτεκτονική EPC*

[Πηγή: https://www.tutorialspoint.com/lte/lte_network_architecture.htm]

Το E-UTRAN συνδέεται με το δίκτυο κορμού μέσω της διεπαφής S1. Η S1 χωρίζεται σε S1-MME και S1-U για την επικοινωνία ενός eNB με την MME και τη S-GW αντίστοιχα. Η διεπαφή S1-MME μεταφέρει μηνύματα σηματοδοσίας που ανήκουν στο στρώμα ελέγχου (C-Plane), ενώ από τη διεπαφή S1-U διέρχονται δεδομένα του στρώματος χρήστη (U-Plane). Ένας eNB μπορεί να είναι συνδεδεμένος με περισσότερες της μίας MME ή S-GW, όπως αντίστοιχα μία MME ή S-GW μπορεί να επικοινωνεί ταυτόχρονα με πολλούς eNBs (βλ. Σχήμα 4.5).

Η κεντρική πύλη εξυπηρέτησης S-GW λειτουργεί ως δρομολογητής, καθώς προωθεί πακέτα δεδομένων του U-Plane που προέρχονται από τους eNBs, προς την P-GW και αντίστροφα (UL & DL κατεύθυνση). Παράλληλα, η S-GW λειτουργεί ως σημείο άγκυρας (anchor point), σε περιπτώσεις μεταπομπής ή αλλαγής τεχνολογίας πρόσβασης (π.χ. από 4G σε 3G) για ένα UE. Επιπλέον, η S-GW πραγματοποιεί την τηλε-ειδοποίηση (paging) ενός UE που βρίσκεται σε αδρανή (idle) κατάσταση, όταν δέχεται δεδομένα στη ζεύξη DL που προορίζονται για αυτό το UE. Τέλος, η S-GW διαχειρίζεται και αποθηκεύει πληροφορίες σχετικά με τους κομιστές IP (IP bearers) των UE που εξυπηρετεί, όπως για παράδειγμα το εξωτερικό δίκτυο δεδομένων (Packet Data Network - PDN) με το οποίο επικοινωνεί κάποιο UE ή την κλάση υπηρεσίας που έχει οριστεί για αυτό (QCI - QoS Class Identifier).



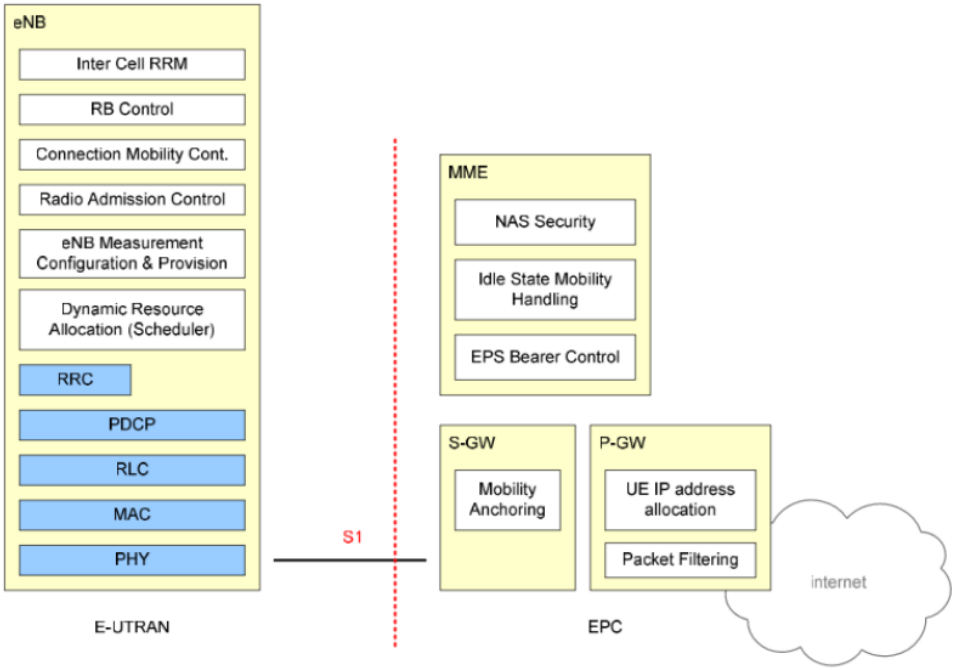
Σχήμα 4.5 *Σύνδεση του E-UTRAN με τις μονάδες MME και S-GW* [41]

Η κεντρική πύλη του δικτύου πακέτων δεδομένων (P-GW) συνδέει το δίκτυο κορμού EPC με εξωτερικά δίκτυα IP, όπως το Internet, μέσω της διεπαφής SGi. Η P-GW εξασφαλίζει τη συνδεσιμότητα των UEs με εξωτερικά δίκτυα, καθώς εκχωρεί στα UEs διευθύνσεις και προθέματα IP. Η μονάδα P-GW λειτουργεί ως κεντρική πύλη του δικτύου LTE, καθώς από αυτήν διέρχεται όλη η τηλεπικοινωνιακή κίνηση από και προς τα UEs. Επομένως, η P-GW είναι υπεύθυνη για την εφαρμογή της πολιτικής χρεώσεων (charging policy), το φιλτράρισμα όλων των πακέτων IP και την εξασφάλιση της ποιότητας υπηρεσίας (QoS) που αντιστοιχεί στους διάφορους κομιστές. Συμπληρωματικά, η P-GW λειτουργεί ως άγκυρα (anchor) για την κινητικότητα ενός UE μεταξύ δικτύων 3GPP και non-3GPP (π.χ. Wi-MAX).

Η μονάδα MME διαχειρίζεται όλες τις πληροφορίες του C-Plane για ένα UE, καθώς αποτελεί τον κύριο κόμβο ελέγχου του E-UTRAN σε ένα δίκτυο LTE. Κύρια ευθύνη της MME είναι η επεξεργασία των μηνυμάτων σηματοδοσίας για την κινητικότητα και την ασφάλεια σύνδεσης στο E-UTRAN, που αποστέλλονται μεταξύ ενός UE και του EPC. Η MME συμμετέχει, επίσης, στην ενεργοποίηση, συντήρηση ή απενεργοποίηση κομιστών και επιλέγει την S-GW που θα εξυπηρετήσει ένα UE, είτε κατά την αρχική εισαγωγή του UE στο δίκτυο είτε σε περίπτωση μεταπομπής. Παράλληλα, η μονάδα MME πραγματοποιεί την πιστοποίηση (authentication) ενός UE και διαχειρίζεται τα κλειδιά ασφαλείας για την κρυπτογράφηση των δεδομένων του C-Plane. Τέλος, η MME αποτελεί το τερματικό άκρο του στρώματος μη πρόσβασης (Non Access Stratum - NAS) της στοίβας πρωτοκόλλων του LTE, η οποία αναλύεται στην §4.2

Ο HSS αποτελεί κεντρική βάση δεδομένων, η οποία περιέχει στοιχεία για όλους τους συνδρομητές του παρόχου του συγκεκριμένου δικτύου LTE. Στα στοιχεία αυτά συμπεριλαμβάνονται το προφίλ QoS κάθε UE, καθώς και πληροφορίες για τις P-GW και MME με τις οποίες το UE μπορεί να συνδεθεί. Ωστόσο, οι πληροφορίες που σχετίζονται με τις πολιτικές χρεώσεων κάθε συνδρομητή αποθηκεύονται σε διακριτή μονάδα, που ονομάζεται PCRF (Policy Control and Charging Rules Function).

Συνολικά, κάθε δομική μονάδα του δικτύου LTE επιτελεί ένα ανεξάρτητο σύνολο λειτουργιών, επιτυγχάνοντας έτσι το λειτουργικό διαχωρισμό των κόμβων του δικτύου (βλ. Σχήμα 4.6). Με μπλε χρώμα επισημαίνονται τα στρώματα της στοίβας πρωτοκόλλων.

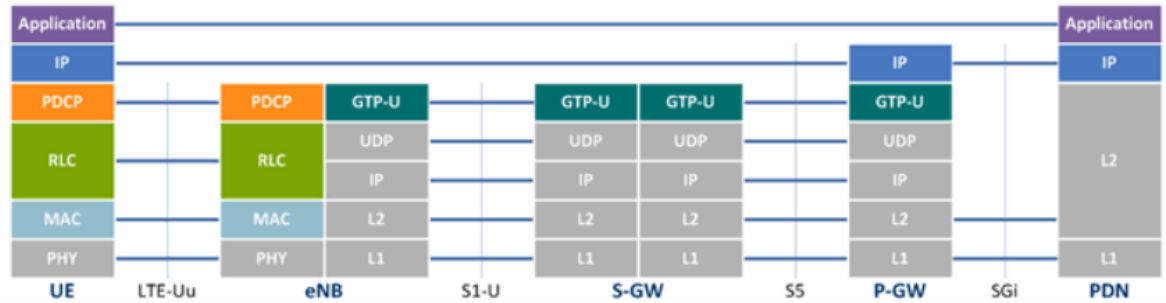


Σχήμα 4.6 *Λειτουργικός διαχωρισμός μεταξύ E-UTRAN και EPC του δικτύου LTE* [41]

4.2 Στοίβα Πρωτοκόλλων LTE

Τα δεδομένα που μεταφέρει ένα δίκτυο LTE χωρίζονται σε δεδομένα επιπέδου χρήστη (U-Plane) και σε δεδομένα επιπέδου ελέγχου (C-Plane). Για κάθε τύπο δεδομένων, χρησιμοποιείται μία σειρά από πρωτόκολλα για την επικοινωνία του UE με τους eNBs και τις μονάδες του EPC. Συγκεκριμένα, για την επικοινωνία μεταξύ του UE και του eNB, τα στρώματα είναι το PDCP (Packet Data Convergence Protocol), το RLC (Radio Link Protocol), το MAC (Medium Access Control) και το φυσικό στρώμα PHY (Physical Layer), τα οποία τερματίζονται στον eNB. Επιπλέον, σε επίπεδο ελέγχου, χρησιμοποιούνται τα πρωτόκολλα RRC (Radio Resource Control) και NAS (Non Access Stratum), με την ιδιαιτερότητα ότι το NAS λειτουργεί μεταξύ του UE και της MME. Πέρα από αυτά τα πρωτόκολλα, χρησιμοποιούνται και ορισμένα ακόμη για την επικοινωνία του eNB με άλλες μονάδες του EPC, όπως αναφέρεται στη συνέχεια.

Κάθε πακέτο IP που προέρχεται από κάποιο εξωτερικό δίκτυο (π.χ. το Internet), προκειμένου να δρομολογηθεί μέσα στο δίκτυο κορμού προς την S-GW και τον eNB, ενθυλακώνεται σε ένα πακέτο του πρωτοκόλλου GTP-U (GPRS Tunneling Protocol for U-Plane), όπως απεικονίζεται στο Σχήμα 4.7. Στο C-Plane, ο eNB επικοινωνεί με την MME μέσω μηνυμάτων S1-AP (S1 Application Protocol), για όποια πληροφορία αφορά τη διαχείριση της κινητικότητα των UEs. Ως πρωτόκολλο μεταφοράς επιλέχθηκε το SCTP (Stream Control Transmission Protocol) [43], επειδή διαθέτει μηχανισμό αξιόπιστης παράδοσης, που είναι απαραίτητος κατά την ανταλλαγή μηνυμάτων σηματοδοσίας.



Σχήμα 4.7 *Στοίβα πρωτοκόλλων για το U-Plane*

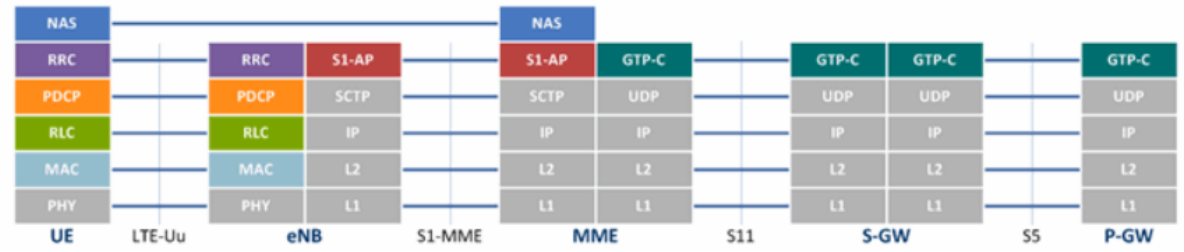
[Πηγή: https://www.netmanias.com/en/post/techdocs/5904/lte-network-architecture]

4.2.1 Non-Access Stratum - NAS

Το πρωτόκολλο NAS αποτελεί το ανώτατο στρώμα της στοίβας πρωτοκόλλων του C-Plane και χρησιμοποιείται για την επικοινωνία μηνυμάτων ελέγχου μεταξύ του UE και του EPC. Από την πλευρά του EPC, το τερματικό σημείο του NAS είναι η μονάδα MME. Το NAS υποστηρίζει λειτουργίες για διαχείριση κινητικότητας (EPS Mobility Management - EMM) και διαχείριση συνόδου (EPS Session Management - ESM). Η λειτουργία EMM περιλαμβάνει την ανίχνευση της γεωγραφικής θέσης ενός UE, μέσω ενημέρωσης του κωδικού περιοχής εντοπισμού (tracking area code - TAC), ενώ η λειτουργία ESM είναι υπεύθυνη για την ενεργοποίηση ή απενεργοποίηση των κομιστών ενός UE.

4.2.2 Radio Resource Control - RRC

Το πρωτόκολλο RRC, που ανήκει στο C-Plane της διεπαφής Uu, διαχειρίζεται όλα τα θέματα που αφορούν την ασύρματη ζεύξη μεταξύ UE και eNB. Το RRC πραγματοποιεί την ευρυεκπομπή (broadcast) των πληροφοριών του συστήματος (System Information) και την αναζήτηση (paging) των UEs. Επιπλέον, το RRC ελέγχει την καθιέρωση και την αποδέσμευση των radio bearers, ενώ διαχειρίζεται τις λειτουργίες σχετικά με την κινητικότητα (π.χ. μεταπομπή). Το RRC ορίζει δύο βασικές καταστάσεις, στις οποίες μπορεί να βρίσκεται ένα UE. Όταν ένα UE είναι σε κατάσταση RRC_IDLE, δεν μεταδίδει ούτε δέχεται δεδομένα και το RRC στην πλευρά του eNB πραγματοποιεί κυρίως τις λειτουργίες της αναζήτησης και της επιλογής κυψέλης (cell selection). Στην κατάσταση RRC_CONNECTED, το UE αναφέρει στον eNB μετρήσεις για την ποιότητα του σήματος λήψης, ενώ ταυτόχρονα είναι εφικτή η μεταφορά δεδομένων από και προς αυτό, μέσω της P-GW.



Σχήμα 4.8 *Στοίβα πρωτοκόλλων για το C-Plane*

[Πηγή: https://www.netmanias.com/en/post/techdocs/5904/lte-network-architecture]

4.2.3 Packet Data Convergence Protocol - PDCP

Το PDCP στο U-Plane χειρίζεται πακέτα IP, ενώ στο C-Plane μεταφέρει τις πληροφορίες του στρώματος RRC. Το PDCP πραγματοποιεί συμπίεση των επικεφαλίδων IP, ενώ υλοποιεί τις λειτουργίες της κρυπτογράφησης και της προστασίας της ακεραιότητας των δεδομένων. Επιπλέον, το PDCP είναι υπεύθυνο για την σειριακή παράδοση δεδομένων στα ανώτερα στρώματα (με χρήση αριθμών ακολουθίας PDCP), καθώς και για την ανίχνευση αντιγράφων στα δεδομένα που δέχεται από τα κατώτερα στρώματα.

4.2.4 Radio Link Control - RLC

Η κύρια λειτουργία του στρώματος RLC είναι οργάνωση των δεδομένων που δέχεται από το PDCP σε κατάλληλο μέγεθος, ώστε αυτά να μεταδοθούν μέσω της ραδιοεπαφής του LTE. Για το σκοπό αυτό, το RLC μπορεί να πραγματοποιεί συνένωση (concatenation) ή τεμαχισμό (segmentation) στα δεδομένα του PDCP. Επιπλέον, στο στρώμα RLC γίνεται αναδιάταξη των δεδομένων κατώτερων στρωμάτων, τα οποία έχουν ληφθεί στο δέκτη με λάθος σειρά. Τέλος, ορίζονται τρεις πιθανοί τρόποι λειτουργίας για το RLC: Acknowledged Mode (AM), Unacknowledged Mode (UM) και Transparent Mode (TM). Ανάλογα με τον τρόπο λειτουργίας, το RLC μπορεί επιπλέον να αναμεταδίδει δεδομένα για διόρθωση σφαλμάτων, με χρήση της τεχνικής ARQ (Automatic Repeat Request).

4.2.5 Medium Access Control - MAC

Το στρώμα MAC είναι υπεύθυνο για την πολύπλεξη των δεδομένων μεταξύ του στρώματος RLC και του φυσικού στρώματος. Όπως αναλύεται εκτενέστερα στην §4.4, το MAC επικοινωνεί με το RLC και το PHY, μέσω λογικών καναλιών και καναλιών μεταφοράς, αντίστοιχα. Τα λογικά κανάλια καθορίζουν το είδος της πληροφορίας που μεταφέρει το RLC, ενώ τα κανάλια μεταφοράς καθορίζουν τον τρόπο με τον οποίο θα μεταδοθεί η πληροφορία πάνω στη ραδιοεπαφή. Οι λειτουργίες του στρώματος MAC συνοψίζονται ως εξής:

- Απεικόνιση των λογικών καναλιών σε κανάλια μεταφοράς
- Χρονοπρογραμματισμός (scheduling) των πόρων του eNB για μετάδοση δεδομένων, σύμφωνα με την προτεραιότητα του κάθε UE.
- Υλοποίηση της διαδικασίας τυχαίας πρόσβασης των UEs (Random Access Procedure).
- Διόρθωση σφαλμάτων με την τεχνική HARQ (Hybrid Automatic Repeat Request).
- Επίτευξη συγχρονισμού των UEs για μετάδοση στη ζεύξη UL.

4.2.6 Physical Layer - PHY

Το φυσικό στρώμα είναι υπεύθυνο για τη μετάδοση των δεδομένων του U-Plane και του C-Plane πάνω από τη ραδιοεπαφή, διαχειρίζεται δηλαδή το διαθέσιμο φάσμα συχνοτήτων που χρησιμοποιεί ο σταθμός βάσης eNB. Το στρώμα PHY υλοποιεί τις

τεχνικές διαμόρφωσης που έχουν επιλεχθεί για τα συστήματα LTE, πριν το τελικό σήμα διαβιβαστεί στις αναλογικές βαθμίδες (π.χ. ενισχυτές, κεραίες κ.ά.) για μετάδοση στον αέρα. Οι διαδικασίες του φυσικού στρώματος αναλύονται στην §4.3.

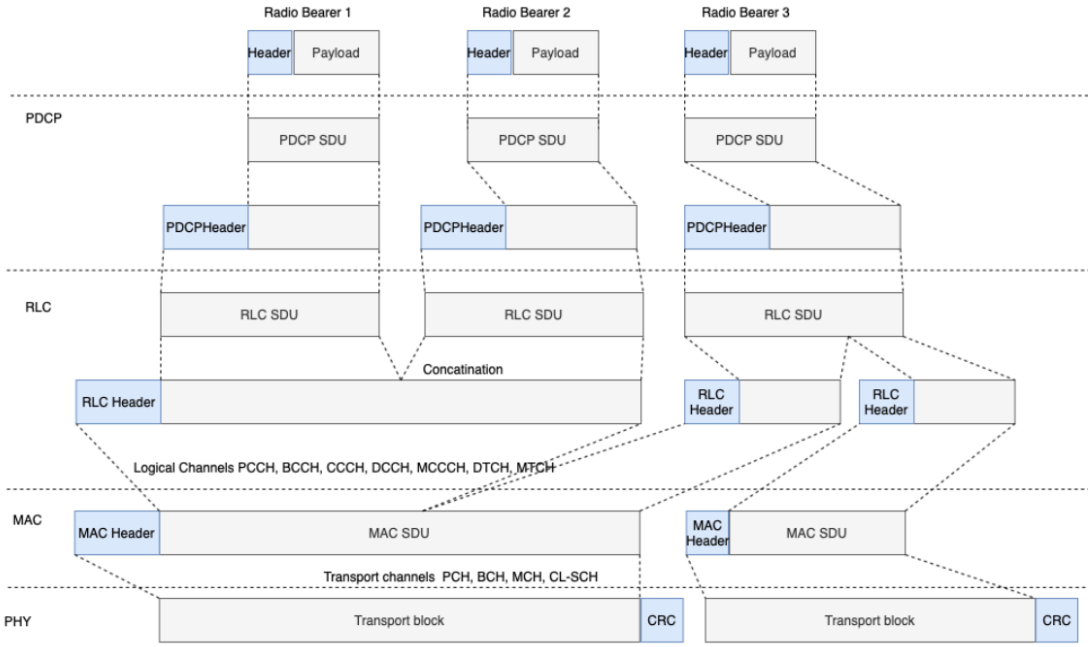
4.2.7 Ροή Δεδομένων στη Στοίβα Πρωτοκόλλων

Στο Σχήμα 4.9 απεικονίζεται ένα λογικό διάγραμμα ροής των δεδομένων, καθώς αυτά επεξεργάζονται από τα διάφορα στρώματα της στοίβας πρωτοκόλλων του LTE. Ένα πακέτο που εισέρχεται ως είσοδος σε ένα επίπεδο της στοίβας καλείται SDU (Service Data Unit), ενώ ένα πακέτο που εξέρχεται από ένα επίπεδο της στοίβας (κατόπιν επεξεργασίας), ονομάζεται PDU (Protocol Data Unit). Ακολουθώντας προσέγγιση από πάνω προς τα κάτω (top - down approach), η ροή δεδομένων έχει ως εξής:

- Πακέτα IP που καταφθάνουν από το στρώμα IP παραδίδονται ως PDCP SDUs στο στρώμα PDCP. Το PDCP συμπιέζει την επικεφαλίδα IP, πραγματοποιεί κρυπτογράφηση των δεδομένων, προσθέτει επικεφαλίδα PDCP και διαβιβάζει τις PDCP PDUs στο στρώμα RLC.

- Το στρώμα RLC συνενώνει ή τεμαχίζει τις RLC SDUs, ώστε να έχουν μέγεθος κατάλληλο για μετάδοση πάνω από τη ραδιοεπαφή. Επιπλέον, προσθέτει μία επικεφαλίδα RLC και παραδίδει τις RLC PDUs στο στρώμα MAC.

- Το στρώμα MAC οργανώνει τις RLC PDUs σε MAC SDUs, σύμφωνα με το χρονοπρογραμματισμό και την προτεραιότητα κάθε πακέτου. Στις MAC SDUs προστίθεται μία επικεφαλίδα MAC και μεταφέρονται στο στρώμα PHY, για να μεταδοθούν ως TBs (Transport Blocks) στη διεπαφή Uu μεταξύ UE και eNB.

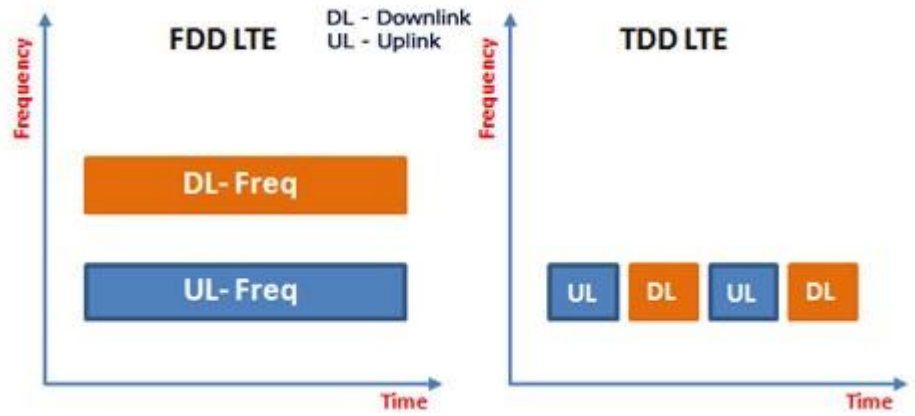


Σχήμα 4.9 *Λογικό διάγραμμα ροής δεδομένων στη στοίβα πρωτοκόλλων του LTE*
[Πηγή: https://www.prodevelopertutorial.com/lte-rlc-layer/]

4.3 Φυσικό Στρώμα LTE

Ο σχεδιασμός του φυσικού στρώματος του LTE πραγματοποιήθηκε με γνώμονα τις νέες απαιτήσεις για υψηλότερους ρυθμούς μετάδοσης, αποδοτικότερη διαχείριση του φάσματος και αυξημένη χωρητικότητα σε χρήστες. Προς ικανοποίηση αυτών των προδιαγραφών, υιοθετήθηκε η τεχνική OFDM (Orthogonal Frequency Division Multiplexing), ως ο κύριος τρόπος μετάδοσης πληροφορίας στη φυσική ασύρματη ζεύξη. Η τεχνική OFDM εμφανίστηκε πρώτη φορά στα μέσα του 1960. Ωστόσο, δεν μπορούσε να υλοποιηθεί λόγω των περιορισμένων δυνατοτήτων των ηλεκτρικών κυκλωμάτων. Η τεχνολογική ανάπτυξη στους κλάδους της ηλεκτρονικής και της ψηφιακή επεξεργασίας σήματος επέτρεψαν την υλοποίηση της OFDM στην πράξη, με αποτέλεσμα να χρησιμοποιείται σήμερα σε αρκετά πρότυπα επικοινωνιών, όπως τα 802.11 (Wi-fi) και DVB (Digital Video Broadcasting) [44].

Ένα δίκτυο LTE διαθέτει δύο είδη λειτουργίας στο φυσικό επίπεδο, αντίστοιχα με τον τρόπο που διαχωρίζονται οι μεταδόσεις στις ζεύξεις UL και DL. Στη λειτουργία FDD (Frequency Division Duplexing), κάθε κατεύθυνση χρησιμοποιεί διαφορετική συχνότητα για τη μετάδοση δεδομένων. Αντίθετα, στη λειτουργία TDD (Time Division Duplexing), οι μεταδόσεις UL και DL χρησιμοποιούν την ίδια ζώνη συχνοτήτων και διαχωρίζονται στο πεδίο του χρόνου. Αν και ο τρόπος TDD χρησιμοποιεί μικρό εύρος συχνοτήτων, η πλειοψηφία των τηλεπικοινωνιακών παρόχων έχουν επιλέξει για τα LTE δίκτυά τους τη λειτουργία FDD, με αποτέλεσμα να είναι σήμερα ο κυρίαρχος τρόπος λειτουργίας των δικτύων LTE. Για το λόγο αυτό, η παρούσα εργασία επικεντρώνεται στον τρόπο λειτουργίας FDD, με τον αναγνώστη να παραπέμπεται στο [45] για μία επισκόπηση των χαρακτηριστικών του LTE-TDD.



Σχήμα 4.10 *Δυνατοί τρόποι λειτουργίας: LTE - FDD και LTE - TDD*

[Πηγή: https://www.differencebetween.com/difference-between-fdd-lte-and-tdd-lte]

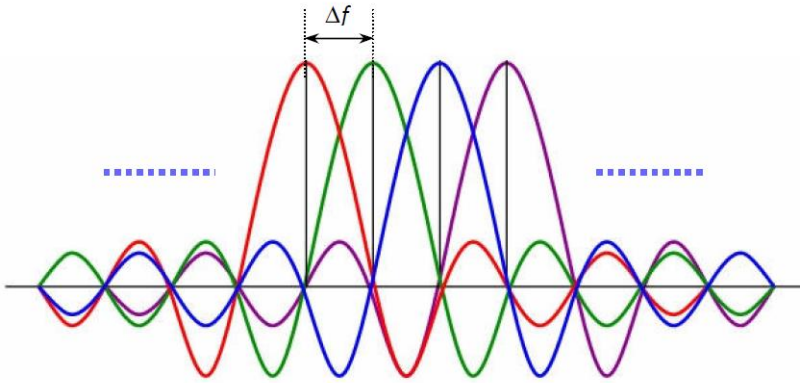
4.3.1 Τεχνικές Πολλαπλής Πρόσβασης OFDMA

Ως σχήμα πολλαπλής πρόσβασης στην κατεύθυνση DL χρησιμοποιείται η συμβατική OFDMA (Orthogonal Frequency Division Multiple Access). Η τεχνική αυτή βασίζεται στην τεχνολογία OFDM, κατά την οποία το συνολικά διαθέσιμο εύρος ζώνης του συστήματος, διαχωρίζεται σε μικρότερα υποφέροντα (sub-carriers) στενής ζώνης. Τα υποφέροντα αυτά απέχουν σταθερό διάστημα Δf και είναι μεταξύ τους ορθογώνια, όπως

απεικονίζεται στο Σχήμα 4.11. Δυο κυματομορφές καλούνται ορθογώνιες, όταν το σημείο μεγιστοποίησης της μίας συμπίπτει με το σημείο μηδενισμού της άλλης. Η ορθογωνιότητα δύο συναρτήσεων εκφράζεται μαθηματικά ως εξής:

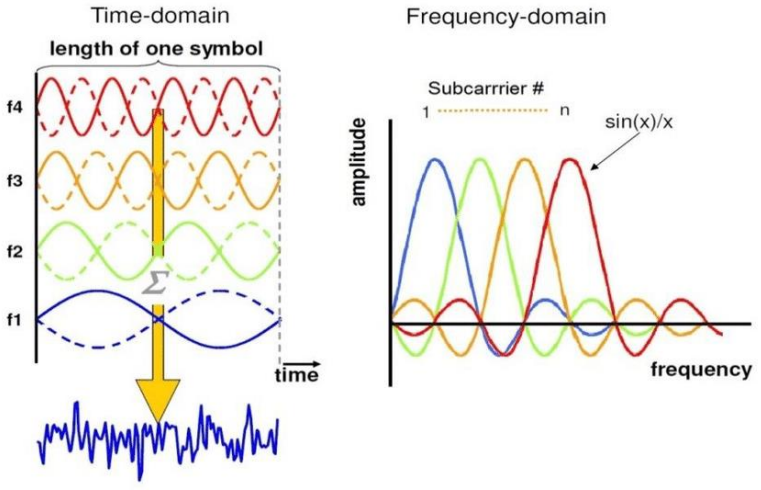
$$\frac{1}{T}\int_0^T f_i(t)\cdot f_j(t)\,dt = \begin{cases} 0, & if\ i \neq j \\ 1, & if\ i = j \end{cases} \tag{4.1}$$

Η ιδιότητα αυτή επιτρέπει στα υποφέρονται να επικαλύπτονται στο πεδίο της συχνότητας, αυξάνοντας την αποδοτικότητα της χρήσης του φάσματος. Επιπλέον, κάθε υποφέρον αντιμετωπίζει ανεξάρτητες διαλείψεις από το δίαυλο (flat fading), καθώς το εύρος ζώνης κάθε υποφέροντος είναι μικρότερο από το εύρος ζώνης συνοχής του διαύλου (Coherence Bandwidth - B_C).



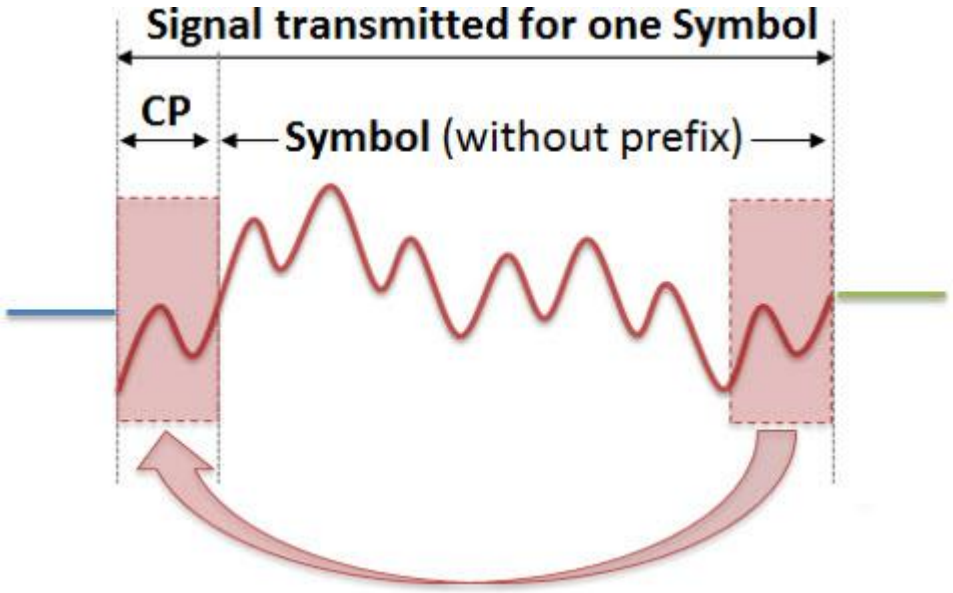
Σχήμα 4.11 *Τα υποφέροντα που χρησιμοποιεί η OFDM στο πεδίο της συχνότητας* [46]

Στο πεδίο του χρόνου, τα αδιαμόρφωτα υποφέροντα είναι ημίτονα της ίδιας χρονικής διάρκειας αλλά διαφορετικής συχνότητας, όπως απεικονίζεται στο Σχήμα 4.12. Τα αδιαμόρφωτα αυτά υποφέροντα διαμορφώνονται με κάποιο σχήμα διαμόρφωσης και αθροίζονται, όπως θα εξηγηθεί στη συνέχεια. Το αποτέλεσμα είναι ένα OFDM σύμβολο διάρκειας T_u, η οποία ονομάζεται και ωφέλιμη διάρκεια συμβόλου.



Σχήμα 4.12 *Αναπαράσταση συμβόλου OFDM στο πεδίο του χρόνου* [47]

Ένας τυπικός ασύρματος δίαυλος επικοινωνίας εμφανίζει χρονική διασπορά. Ένας δέκτης λαμβάνει πολλαπλά αντίγραφα ενός σήματος εκπομπής με ποικίλες χρονικές καθυστερήσεις (delay spread), εξαιτίας του φαινομένου της πολυδιαδρομικής διάδοσης λόγω ανακλάσεων του σήματος. Ως αποτέλεσμα αυτής της χρονικής διασποράς, δύο διαδοχικά OFDM σύμβολα μπορεί να επικαλύπτονται στο χρόνο, οπότε προκαλείται διασυμβολική παρεμβολή (Inter Symbol Interference - ISI). Επιπλέον, προκαλείται μερική απώλεια της ορθογωνιότητας στο πεδίο της συχνότητας, οπότε εμφανίζεται και παρεμβολή μεταξύ γειτονικών υποφερόντων (Inter Carrier Interference - ICI). Προς αποφυγή αυτών των φαινομένων, προστίθεται στην αρχή κάθε OFDM συμβόλου ένα χρονικό διάστημα προστασίας T_g. Το διάστημα αυτό, που ονομάζεται cyclic prefix (CP), είναι μία επανάληψη ενός τμήματος από το τέλος του OFDM συμβόλου (βλ. Σχήμα 4.13). Η συνολική διάρκεια συμβόλου γίνεται $T_s = T_g + T_u$, προσδίδοντας έτσι στο OFDM σύμβολο ανοχή στη χρονική διασπορά που εισάγει ο ασύρματος δίαυλος.



Σχήμα 4.13 *Εισαγωγή Cyclic Prefix σε ένα OFDM σύμβολο*
[Πηγή: https://www.telecomhall.net/t/what-is-cp-cyclic-prefix-in-lte/6369]

Συνοπτικά, τα πλεονεκτήματα της μεθόδου OFDM ως σχήμα πολλαπλής πρόσβασης είναι τα εξής:

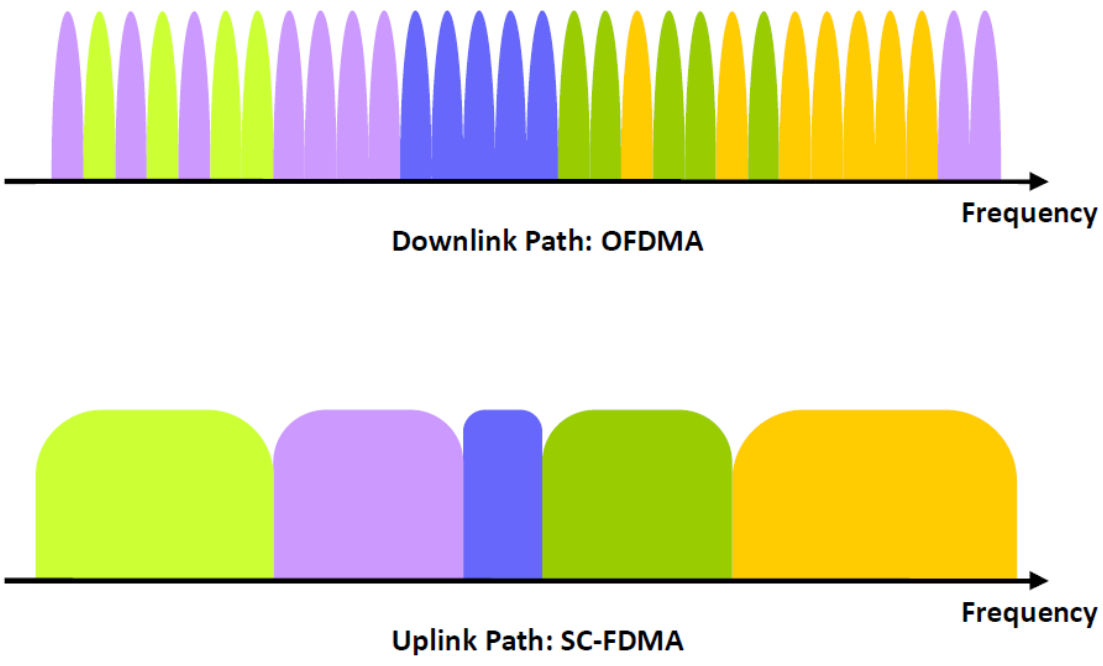
- Η μεγάλη ωφέλιμη διάρκεια συμβόλου T_u (που προκύπτει από το διαμοιρασμό του συνολικού φάσματος σε μικρότερα τμήματα) καθώς και η χρήση του cyclic prefix, αυξάνουν την ανθεκτικότητα του συστήματος στο φαινόμενο της πολυδιαδρομικής διάδοσης και περιορίζουν τις παρεμβολές ISI και ICI.

- Ευέλικτη διαχείριση του φάσματος, μέσω δυναμικά προσαρμοστικής εκχώρησης ραδιοπόρων στο χρόνο και τη συχνότητα.

- Αύξηση φασματικής απόδοσης (spectral efficiency), εξαιτίας της ορθογωνιότητας των υποφερόντων.

- Βελτιστοποίηση του ρυθμού μετάδοσης κάθε χρήστη μίας κυψέλης, μέσω της εκπομπής των δεδομένων του χρήστη στα υποφέροντα που εμφανίζουν την μικρότερη απόσβεση (fading). Το χαρακτηριστικό αυτό αποτελεί θεμελιώδες τμήμα της τεχνικής OFDMA.

Βέβαια, παρά τα αρκετά πλεονεκτήματα της συμβατικής OFDMA, υπάρχουν και ορισμένα μειονεκτήματα. Ένα από αυτά είναι ότι εμφανίζει υψηλό λόγο μέγιστης προς μέση ισχύ (Peak to Average Power Ratio - PAPR). Οι υψηλές τιμές PAPR οφείλονται στην τυχαία προσθετική συμβολή πολλών υποφερόντων, καθώς αυτά προστίθενται για να δημιουργήσουν ένα OFDM σύμβολο. Το πρόβλημα αυτό επιλύεται στην πλευρά του σταθμού βάσης eNB, με χρήση ειδικών τεχνικών γραμμικοποίησης (linearization) των ενισχυτών ισχύος. Ωστόσο, οι τεχνικές αυτές δεν μπορούν να εφαρμοστούν στα κινητά τερματικά UE, καθώς θα αύξαναν πολύ το κόστος των συσκευών. Επομένως, για τη ζεύξη UL προτάθηκε και υιοθετήθηκε η τεχνική Single Carrier FDMA (SC - FDMA), που αποτελεί παραλλαγή της OFDMA [48]. Στην τεχνική SC-FDMA, ένα OFDM σύμβολο «εξαπλώνεται» σε εύρος πολλαπλών υποφερόντων, μειώνοντας έτσι την τιμή PAPR [49]. Η διαφορά των δύο τεχνικών απεικονίζεται στο Σχήμα 4.14, με τους χρήστες να επισημαίνονται με διαφορετικά χρώματα. Στην κατεύθυνση DL, παρατηρείται ότι μπορούν να ανατεθούν σε ένα χρήστη πολλαπλά υποφέροντα, που δεν είναι απαραίτητα διαδοχικά. Αντίθετα, στην κατεύθυνση UL, σε κάθε χρήστη ανατίθεται συγκεκριμένη ζώνη συχνοτήτων, το εύρος της οποίας καθορίζεται από τον eNB, σύμφωνα με τις ανάγκες ρυθμού μετάδοσης κάθε χρήστη.

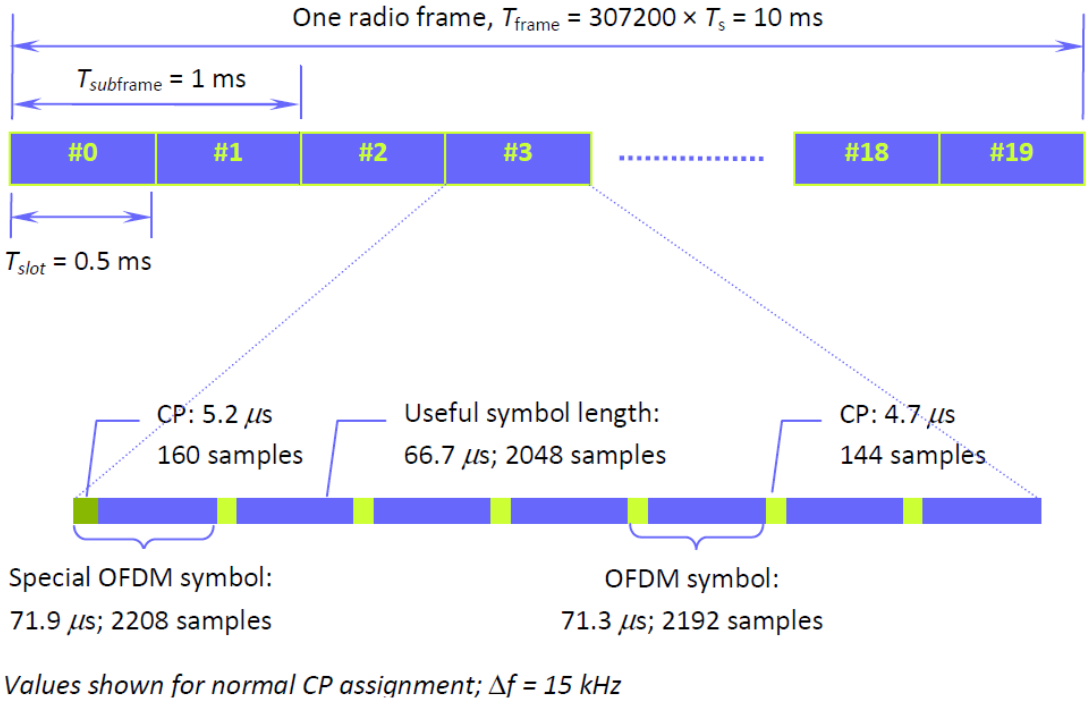


Σχήμα 4.14 *Κατανομή εύρους ζώνης στις τεχνικές OFDMA και SC-FDMA* [46]

4.3.2 Δομή Πλαισίου FDD

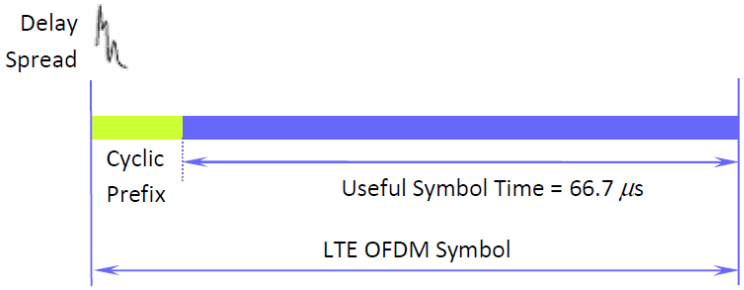
Στα συστήματα LTE, τα υποφέροντα απέχουν μεταξύ τους διάστημα $\Delta f = 15kHz$, επομένως η ωφέλιμη διάρκεια συμβόλου είναι $T_u = 1/\Delta f \approx 66.7\mu s$. Στο πεδίο του χρόνου, η πληροφορία οργανώνεται σε πλαίσια LTE (LTE frames). Ένα πλαίσιο διαρκεί 10ms,

όπως απεικονίζεται στο Σχήμα 4.15. Κάθε πλαίσιο χωρίζεται σε δέκα υποπλαίσια (sub-frames), διάρκειας 1ms. Ο χρονοπρογραμματισμός (scheduling) τη μετάδοσης δεδομένων στο φυσικό στρώμα γίνεται σε επίπεδο υποπλαισίων. Κάθε υποπλαίσιο χωρίζεται περαιτέρω σε δύο σχισμές (slots), διάρκειας 0.5ms η κάθε μία. Τέλος, κάθε σχισμή περιέχει έξι ή εφτά OFDM σύμβολα, αναλόγως με το αν χρησιμοποιείται κανονικό (normal) ή εκτεταμένο (extended) cyclic prefix.



Σχήμα 4.15 *Δομή πλαισίου FDD με χρήση κανονικού cyclic prefix* [46]

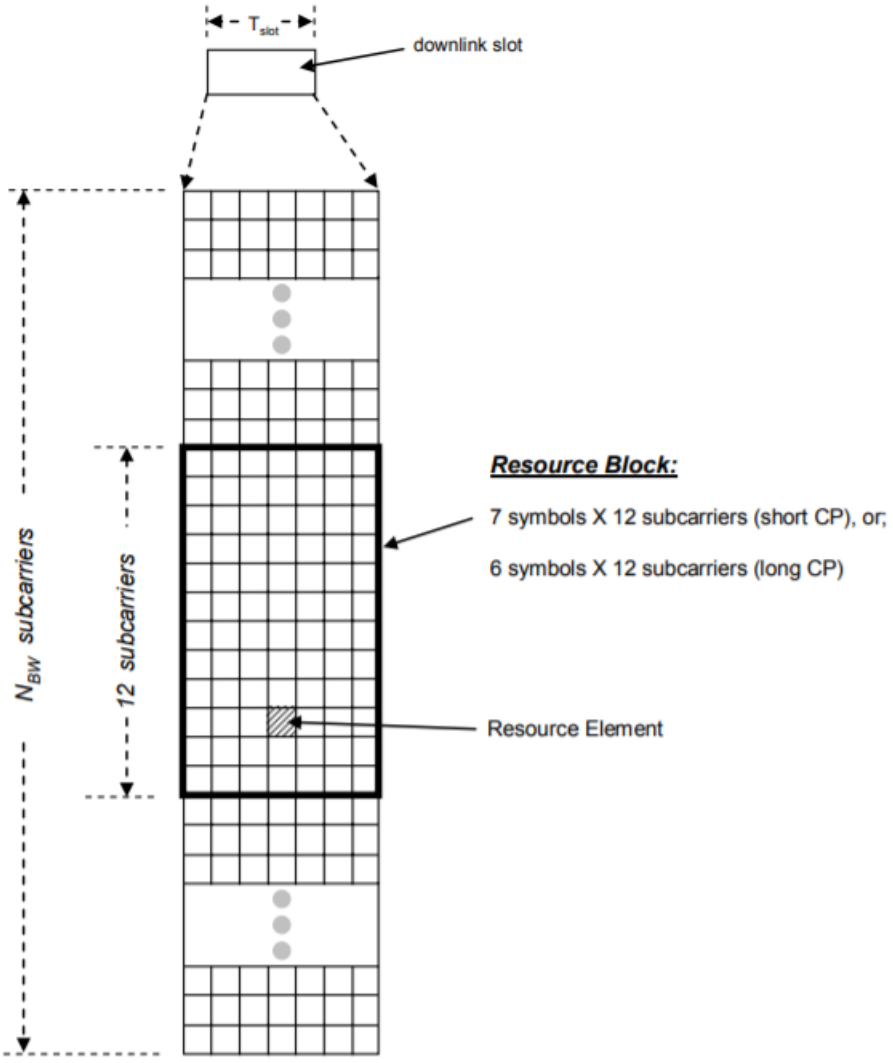
Όπως αναφέρθηκε, η ωφέλιμη διάρκεια συμβόλου είναι $T_u \approx 66.7\mu s$. Στην περίπτωση χρήσης κανονικού CP, το πρώτο OFDM σύμβολο μίας σχισμής έχει cyclic prefix διάρκειας $T_{CP} \approx 5.2\mu s$, ενώ τα υπόλοιπα έξι σύμβολα έχουν cyclic prefix διάρκειας $T_{CP} \approx 4.7\mu s$ (βλ. Σχήμα 4.15). Στην περίπτωση χρήσης εκτεταμένου CP, αυτό έχει διάρκεια $T_{CP\text{-}e} \approx 16.7\mu s$. Τα cyclic prefixes δεν μεταφέρουν ωφέλιμη πληροφορία, οπότε η χρήση τους μειώνει την χωρητικότητα του φυσικού στρώματος για μεταφορά δεδομένων (περίπου κατά 7.5% στην περίπτωση κανονικού CP). Το κανονικό CP χρησιμοποιείται σε αστικές περιοχές (urban areas), ενώ το εκτεταμένο CP χρησιμοποιείται σε κυψέλες με μεγάλη ακτίνα κάλυψης, που συνήθως βρίσκονται σε αγροτικές περιοχές (rural areas).



Σχήμα 4.16 *Δομή OFDM συμβόλου* [46]

4.3.3 Σχήμα μετάδοσης DL

Τα συστήματα LTE - FDD υποστηρίζουν έξι τιμές για το συνολικό εύρος ζώνης των ζεύξεων DL και UL, από 1.4MHz έως 20MHz ανά κατεύθυνση. Το εύρος ζώνης χωρίζεται σε υποφέροντα που απέχουν μεταξύ τους $\Delta f = 15kHz$, ενώ ο συνολικός αριθμός τους εξαρτάται από το μέγεθος του εύρους ζώνης. Το σήμα εκπομπής από ένα σταθμό βάσης eNB στη ζεύξη DL μπορεί να αναπαρασταθεί ως ένα πλέγμα χρόνου - συχνότητας, που ονομάζεται πλέγμα ραδιοπόρων (Resource Grid). Ένα τέτοιο πλέγμα απεικονίζεται στο Σχήμα 4.17, για τη χρονική διάρκεια μίας σχισμής (slot).

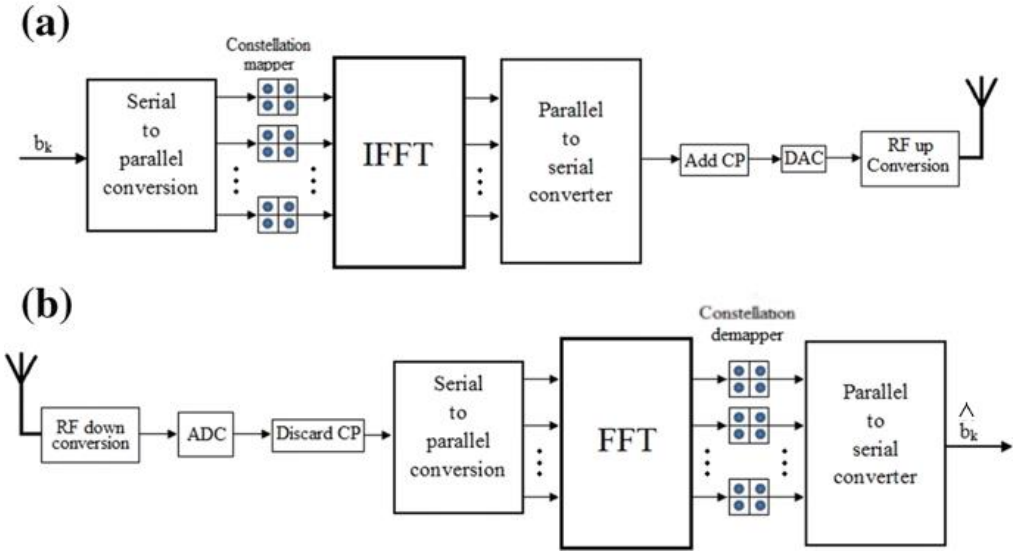


Σχήμα 4.17 *LTE DL Resource Grid* [50]

Κάθε «κουτί» του πλέγματος αναπαριστά ένα RE (Resource Element), το οποίο αποτελεί τη μικρότερη διακριτή μονάδα που μεταφέρει δεδομένα. Ένα RE αντιπροσωπεύει ένα υποφέρον σε διάρκεια ενός OFDM συμβόλου. Η μικρότερη μονάδα που μπορεί να ανατεθεί σε ένα χρήστη από τον eNB είναι το RB (Resource Block), που αποτελείται από 12 υποφέροντα σε διάρκεια μίας σχισμής (0.5ms). Επομένως, το ελάχιστο εύρος ζώνης που ανατίθεται σε ένα χρήστη είναι $12 \times 15 = 180kHz$. Τα RBs που

ανατίθενται σε ένα χρήστη δεν είναι απαραίτητο να είναι διαδοχικά στο χρόνο ή τη συχνότητα. Αυτό αποφασίζεται σε κάθε περίπτωση από τον eNB.

Όπως συμβαίνει σχεδόν με όλα τα συστήματα επικοινωνιών, ένας αριθμός από υποφέροντα δεν μεταφέρουν πληροφορία και χρησιμοποιούνται ως διαστήματα φύλαξης (guard bands). Στα συστήματα LTE, το υποφέρον της DC συνιστώσας, καθώς και συγκεκριμένος αριθμός από υποφέροντα στα δύο άκρα του συνολικού εύρους ζώνης χρησιμοποιούνται ως υποφέροντα φύλαξης (guard sub-carriers). Συνολικά, το διαθέσιμο εύρος ζώνης του διαύλου χρησιμοποιείται σε περίπου σε ποσοστό 90-91%.



Σχήμα 4.18 *Δομικό διάγραμμα υλοποίησης OFDMA στο LTE* [51]

Στο Σχήμα 4.18 απεικονίζεται το δομικό διάγραμμα (block diagram) της υλοποίησης της τεχνικής OFDMA, στα συστήματα LTE για (a) εκπομπή και (b) λήψη. Τα βασικά βήματα στην αλυσίδα εκπομπής είναι τα εξής (στην πλευρά λήψης του UE η διαδικασία αντιστρέφεται και παράγεται η εκτίμηση ($\hat{b}_k$) του αρχικού bit stream):

- Ένα εισερχόμενο σειριακό bit stream (b_k) μετατρέπεται σε παράλληλα bit streams μέσω του μετατροπέα S/P (Serial to Parallel). Έπειτα, κάθε παράλληλο bit stream διαμορφώνεται ανεξάρτητα από τα υπόλοιπα με κάποιο δυνατό σχήμα διαμόρφωσης. Στο LTE χρησιμοποιούνται τα σχήματα διαμόρφωσης QPSK, 16QAM ή 64QAM. Η έξοδος κάθε μονάδας Constellation mapper είναι ένας μιγαδικός αριθμός X_k (μιγαδικό σύμβολο) που αντιπροσωπεύει την θέση του ψηφιακού συμβόλου στον αστερισμό του σχήματος διαμόρφωσης που επιλέχθηκε για το *κ-οστό* παράλληλο bit stream.

- Στη συνέχεια, κάθε μιγαδικό σύμβολο X_k αντιστοιχίζεται σε ένα υποφέρον για μετάδοση, με χρήση του μετασχηματισμού IFFT (Inverse Fast Fourier Transform) μεγέθους N. Οι δυνατές τιμές του μεγέθους N κυμαίνονται από 128 έως 2048 και εξαρτώνται από το διαθέσιμο εύρος ζώνης. Όπως έχει αναφερθεί, ένα OFDM σύμβολο αποτελεί το άθροισμα όλων των υποφερόντων (βλ. Σχήμα 4.12), τα οποία έχουν διαμορφωθεί ανεξάρτητα με κάποιο σχήμα διαμόρφωσης. Πρακτικά λοιπόν, η έξοδος του μετασχηματισμού IFFT είναι ένα δειγματοληπτημένο στο

χρόνο (time sampled) OFDM σύμβολο, που έχει μήκος *N* δείγματα. Η συχνότητα δειγματοληψίας (sampling frequency) ενός OFDM συμβόλου είναι ίση με $f_s = \Delta f \times N = 15kHz \times N$. Το σήμα στην έξοδο του IFFT δίνεται από την εξίσωση:

$$v(t) = \sum_{k=0}^{N-1} X_k \cdot e^{j2\pi kt/T_u}, \qquad 0 \le t \le T_u \tag{4.2}$$

- Έπειτα, τα *N* δείγματα του συμβόλου OFDM μετασχηματίζονται σε μία σειριακή ακολουθία και ορισμένα από τα τελευταία δείγματα της ακολουθίας επαναλαμβάνονται στην αρχή της. Το βήμα αυτό αντιστοιχεί πρακτικά στην εισαγωγή του cyclic prefix.

- Τέλος, το ψηφιακό σήμα του OFDM συμβόλου μετατρέπεται σε αναλογικό, μέσω ενός μετατροπέα DAC (Digital to Analog Converter) και εκπέμπεται από τη μονάδα RF στον ασύρματο δίαυλο σε κάποια κεντρική συχνότητα f_c. Το τελικό σήμα εκπομπής δίνεται από την εξίσωση:

$$s(t) = \boldsymbol{\mathcal{R}}\{v(t) \cdot e^{j2\pi f_c t}\} \tag{4.3}$$

Ως παράδειγμα κατανόησης, δίνεται ο υπολογισμός του μέγιστου ρυθμού μετάδοσης, για εύρος ζώνης 20MHz. Στη διάρκεια μίας σχισμής (0.5ms), ένα υποφέρον μεταφέρει 7 OFDM σύμβολα (για κανονικό CP), και τα συνολικά υποφέροντα είναι 1200. Αν υποτεθεί ότι όλα τα σύμβολα έχουν προκύψει από διαμόρφωση 64QAM, τότε κάθε OFDM σύμβολο κωδικοποιείται από 6 bits. Επομένως, ο μέγιστος ρυθμός μετάδοσης είναι:

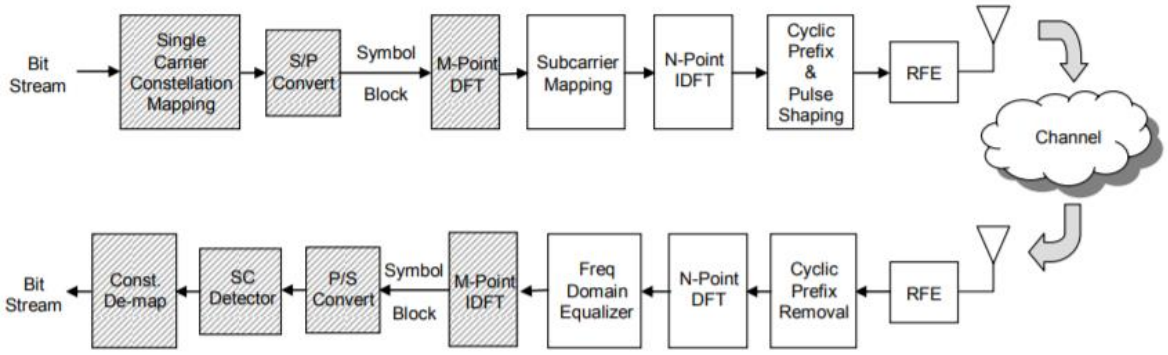
$$R = \frac{7\frac{symbols}{sub-carrier} \times 1200\ (sub-carriers) \times 6\frac{bits}{symbol}}{0.5ms} = 100.8\ Mbps \tag{4.4}$$

Διαθέσιμο Εύρος Ζώνης (MHz)	**1.4**	**3**	**5**	**10**	**15**	**20**
Διάρκεια πλαισίου (ms)	10					
Διάρκεια υποπλαισίου (ms)	1					
Εύρος υποδιαύλου (kHz)	15					
Συχνότητα δειγματοληψίας (MHz)	1.92	3.84	7.68	15.36	23.04	30.72
Μέγεθος FFT	128	256	512	1024	1536	2048
Πλήθος RBs	6	15	25	50	75	100
Πλήθος υποφερόντων	72	180	300	600	900	1200
Χρησιμοποιούμενο εύρος ζώνης (MHz)	1.08	2.7	4.5	9	13.5	18
Αποδοτικότητα εύρους ζώνης διαύλου	77.1%	90%	90%	90%	90%	90%
Πλήθος OFDM συμβόλων ανά σχισμή	7/6 (κανονικό/εκτεταμένο CP)					
Διάρκεια κανονικού CP (μs)	5.2 (πρώτο σύμβολο) / 4.69 (επόμενα 6 σύμβολα)					
Διάρκεια εκτεταμένου CP (μs)	16.67					
Μέγιστος Ρυθμός Μετάδοσης (Mbps) (SISO & κανονικό CP)	6.048	15.12	25.2	50.4	75.2	100.8

Πίνακας 4.1 *Παράμετροι φυσικού στρώματος LTE*

4.3.4 Σχήμα μετάδοσης UL

Η μετάδοση στην κατεύθυνση UL ακολουθεί παρόμοιο σχήμα με την κατεύθυνση DL. Οι τιμές των παραμέτρων είναι ίδιες με αυτές που παρουσιάζονται στον Πίνακα 4.1. Όπως αναφέρθηκε στην §4.3.1, σε ένα χρήστη στην κατεύθυνση UL ανατίθενται RBs που είναι διαδοχικά στη συχνότητα (βλ. Σχήμα 4.14). Η ουσιώδης διαφορά έγκειται στο ότι στην κατεύθυνση UL χρησιμοποιείται η τεχνική SC-FDMA. Στο Σχήμα 4.19 απεικονίζεται το block diagram της υλοποίησης της τεχνικής SC-FDMA στα συστήματα LTE. Ορισμένες λειτουργικές μονάδες είναι ίδιες με τις αντίστοιχες στην κατεύθυνση DL. Τα βασικά βήματα στην αλυσίδα εκπομπής είναι:



Σχήμα 4.19 *Δομικό διάγραμμα υλοποίησης SC-FDMA στο LTE* [50]

- Ένα εισερχόμενο bit stream (b_k) διαμορφώνεται ως μία ακολουθία μιγαδικών συμβόλων, σύμφωνα με κάποιο σχήμα διαμόρφωσης. Τα διαθέσιμα σχήματα διαμόρφωσης για την κατεύθυνση UL είναι τα BPSK, QPSK ή 16QAM και επιλέγονται με βάση τις συνθήκες του διαύλου. Κατόπιν, ο μετατροπέας S/P μετατρέπει την ακολουθία συμβόλων από σειριακή σε παράλληλη, ώστε τα σύμβολα να δοθούν ως είσοδοι στον μετασχηματισμό FFT.

- Με τη χρήση του μετασχηματισμού FFT μεγέθους M, τα σύμβολα εισόδου αναπαρίστανται ως M διακριτά δείγματα στη συχνότητα. Πρακτικά, η πληροφορία που προέρχεται από ένα χρήστη διαμοιράζεται σε M υποφέροντα, για αυτό και πολλές φορές η τεχνική SC-FDMA αναφέρεται στη βιβλιογραφία ως DFTS-OFDM (DFT-Spread OFDM).

- Έπειτα, τα M δείγματα του μετασχηματισμού FFT αντιστοιχίζονται στα υποφέροντα που έχουν επιλεγεί για μετάδοση στη ζεύξη UL, και οδηγούνται ως είσοδοι στο μετασχηματισμό IFFT. Υπενθυμίζεται ότι τα υποφέροντα αυτά είναι διαδοχικά στο πεδίο της συχνότητας.

- Ο μετασχηματισμός IDFT έχει ίδια λειτουργικότητα με τη ζεύξη DL, καθώς παράγει N δείγματα ενός SC-FDMA συμβόλου στο πεδίο του χρόνου (N time samples). Επιπλέον, παρόμοια με τη ζεύξη DL, προστίθεται στο SC-FDMA σύμβολο ένα cyclic prefix.

- Τέλος, το ψηφιακό σήμα του SC-FDMA συμβόλου μετατρέπεται σε αναλογικό και εκπέμπεται από τη μονάδα RF στον ασύρματο δίαυλο.

Στη πλευρά λήψης του eNB, η ανωτέρω διαδικασία αντιστρέφεται και παράγεται μία εκτίμηση του bit stream που εκπέμφθηκε ($\hat{b}_k$). Υπενθυμίζεται ότι, συγκριτικά με την OFDMA, η τεχνική SC-FDMA επιτυγχάνει χαμηλότερες τιμές PAPR, που είναι βασική απαίτηση για μετάδοση στη ζεύξη UL.

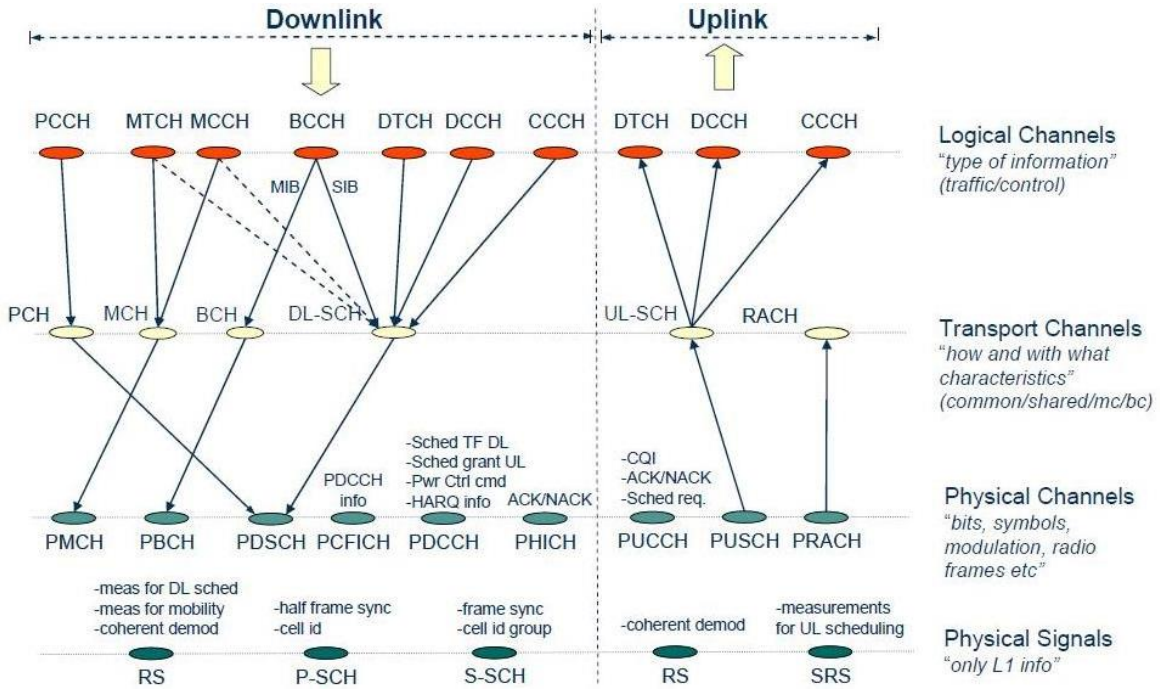
4.4 Κανάλια LTE

Στα συστήματα LTE, η ροή της πληροφορίας μεταξύ των διαφορετικών στρωμάτων της στοίβας πρωτοκόλλων του LTE οργανώνεται σε κανάλια (channels). Τα κανάλια αυτά διαφοροποιούνται και διακρίνονται σε υποκατηγορίες, αντίστοιχα με το είδος της πληροφορίας που μεταφέρουν, την κατεύθυνση της ζεύξης που λειτουργούν και τη χρησιμότητά τους. Όλα τα κανάλια που ορίζονται στο LTE απεικονίζονται στο Σχήμα 4.20, και ανήκουν σε μία από τις εξής τρεις κύριες κατηγορίες:

- **Λογικά Κανάλια (Logical Channels)**: Καθορίζουν το είδος της πληροφορίας που μεταφέρεται (π.χ. δεδομένα χρήστη, μήνυμα ελέγχου κ.ά.) και ανήκουν στο στρώμα RLC.

- **Κανάλια Μεταφοράς (Transport Channels)**: Καθορίζουν τον τρόπο με τον οποίο μεταφέρεται κάθε είδος πληροφορίας (π.χ. ευρυεκπομπή) και ανήκουν στο στρώμα MAC.

- **Φυσικά Κανάλια (Physical Channels)**: Καθορίζουν τη θέση μέσα στο πλέγμα ραδιοπόρων, στην οποία μεταδίδεται κάθε είδος πληροφορίας, και ανήκουν στο στρώμα PHY.



Σχήμα 4.20 *Η απεικόνιση μεταξύ των διαφορετικών κατηγοριών καναλιών του LTE*
[Πηγή: http://worldtechie.blogspot.com/2013/05/lte-channel-structure-and-mapping.html]

4.4.1 Λογικά Κανάλια

Τα λογικά κανάλια χωρίζονται σε κανάλια κίνησης (traffic channels), που μεταφέρουν δεδομένα του U-Plane, και σε κανάλια ελέγχου (control channels), που μεταφέρουν δεδομένα του C-Plane. Επιπλέον, ορισμένα λογικά κανάλια είναι εκχωρημένα αποκλειστικά σε ένα UE (dedicated logical channels) και άλλα χρησιμοποιούνται από περισσότερα UEs (common logical channels). Τα είδη των λογικών καναλιών συνοψίζονται στον Πίνακα 4.2.

Λογικό Κανάλι		**Είδος Πληροφορίας**	**Κατεύθυνση ζεύξης**
DTCH	Dedicated Traffic Channel	Δεδομένα U-Plane	UL, DL
DCCH	Dedicated Control Channel	Δεδομένα C-Plane	UL, DL
CCCH	Common Control Channel	Δεδομένα C-Plane	UL, DL
PCCH	Paging Control Channel	Μηνύματα τηλε-ειδοποίησης	DL
BCCH	Broadcast Control Channel	Μηνύματα ευρυεκπομπής	DL

Πίνακας 4.2 *Είδη λογικών καναλιών LTE*

Το κανάλι BCCH χρησιμοποιείται για την ευρυεκπομπή (broadcast) ορισμένων κρίσιμων παραμέτρων μίας κυψέλης προς όλα τα UE που βρίσκονται εντός της περιοχής κάλυψης της κυψέλης. Οι πληροφορίες αυτές είναι απαραίτητες (π.χ. εύρος ζώνης), προκειμένου να μπορέσει ένα UE να προσαρμόσει τις δικές του ρυθμίσεις και να συνδεθεί στον eNB. Το κανάλι PCCH μεταφέρει μηνύματα τηλε-ειδοποίησης (paging), που απευθύνονται σε UE των οποίων η θέση δεν είναι γνωστή μέσα στο δίκτυο. Το κανάλι CCCH χρησιμοποιείται από UEs τα οποία επικοινωνούν για πρώτη φορά με κάποια κυψέλη, ώστε να συνδεθούν στο στρώμα RRC.

4.4.2 Κανάλια Μεταφοράς

Κάθε λογικό κανάλι αντιστοιχίζεται σε κάποιο κανάλι μεταφοράς, με την πλειοψηφία των λογικών καναλιών να απεικονίζονται στα μεριζόμενα κανάλια (shared channels). Τα κανάλια μεταφοράς παρουσιάζονται στον Πίνακα 4.3.

Κανάλι Μεταφοράς		**Είδος Πληροφορίας**	**Κατεύθυνση ζεύξης**
UL-SCH	Uplink Shared Channel	Δεδομένα και σηματοδοσία άνω ζεύξης	UL
RACH	Random Access Channel	Αιτήματα τυχαίας πρόσβασης	UL
DL-SCH	Downlink Shared Channel	Δεδομένα και σηματοδοσία κάτω ζεύξης	DL
PCH	Paging Channel	Μηνύματα τηλε-ειδοποίησης	DL
BCH	Broadcast Channel	Μηνύματα ευρυεκπομπής	DL

Πίνακας 4.3 *Είδη καναλιών μεταφοράς LTE*

Το κανάλι PCH μεταφέρει τα μηνύματα paging του λογικού καναλιού PCCH. Το κανάλι BCH μεταφέρει το κρισιμότερο τμήμα του λογικού καναλιού BCCH, το οποίο ονομάζεται MIB (Master Information Block). Οι υπόλοιπες πληροφορίες του λογικού καναλιού BCCH, που ονομάζονται SIBs (System Information Blocks), μεταφέρονται στο κανάλι DL-SCH. Σε ένα UE εκχωρούνται ραδιοπόροι από τον eNB, ώστε να μεταφέρει δεδομένα στην ζεύξη UL. Ωστόσο, ορισμένες ενέργειες ενός χρήστη εμπεριέχουν έναν βαθμό τυχαιότητας (π.χ. πραγματοποίησης τηλεφωνικής κλήσης). Επομένως, ένα UE χρησιμοποιεί εκπομπές τυχαίας πρόσβασης (random access), προκειμένου να ζητήσει την εκχώρηση ραδιοπόρων από τον eNB. Τα αιτήματα τυχαίας πρόσβασης μεταφέρονται στο κανάλι RACH.

Τα κανάλια μεταφοράς, εκτός από το είδος της πληροφορίας που μεταφέρουν, διαφέρουν και στην αντιμετώπιση των σφαλμάτων. Για παράδειγμα, στα κανάλια UL-SCH και DL-SCH εφαρμόζονται οι τεχνικές ARQ και HARQ. Επιπλέον, χρησιμοποιείται σχήμα προσαρμοστικής διαμόρφωσης και κωδικοποίησης (Adaptive Coding and Modulation - ACM), ώστε να προσαρμόζεται η μετάδοση στις συνθήκες του διαύλου. Αντίθετα, στα υπόλοιπα κανάλια μεταφοράς χρησιμοποιείται κωδικοποίηση FEC (Forward Error Correction) με σταθερό ρυθμό κώδικα, ο οποίος συνήθως είναι χαμηλός, προκειμένου να εξασφαλίζεται μεγάλη ανοχή σε σφάλματα.

4.4.3 Φυσικά Κανάλια Δεδομένων

Τα φυσικά κανάλια δεδομένων απεικονίζουν τα δεδομένα που λαμβάνουν από το στρώμα MAC σε συγκεκριμένες θέσεις του πλέγματος ραδιοπόρων. Για κάθε φυσικό κανάλι δεδομένων ορίζονται τα υποφέροντα, τα OFDM σύμβολα και το σχήμα διαμόρφωσης, που θα χρησιμοποιηθούν για τη μετάδοση της πληροφορίας κάθε καναλιού. Τα φυσικά κανάλια δεδομένων συνοψίζονται στον Πίνακα 4.4.

Φυσικό Κανάλι Δεδομένων		**Είδος Πληροφορίας**	**Κατεύθυνση ζεύξης**
PUSCH	Physical Uplink Shared Channel	Δεδομένα και σηματοδοσία άνω ζεύξης	UL
PRACH	Physical Random-Access Channel	Αιτήματα τυχαίας πρόσβασης	UL
PDSCH	Physical Downlink Shared Channel	Δεδομένα και σηματοδοσία κάτω ζεύξης	DL
PBCH	Physical Broadcast Channel	Μηνύματα ευρυεκπομπής	DL

Πίνακας 4.4 *Είδη φυσικών καναλιών δεδομένων LTE*

Τα μεριζόμενα φυσικά κανάλια PDSCH και PUSCH μεταφέρουν τα δεδομένα από τα κανάλια μεταφοράς DL-SCH και UL-SCH αντίστοιχα, αλλά και τα μηνύματα paging του καναλιού μεταφοράς PCH. Τα κανάλια PDSCH και PUSCH χρησιμοποιούν την τεχνική ACM, ώστε να προσαρμόζεται η μετάδοση με βάση τις συνθήκες του ασύρματου διαύλου. Το κανάλι PBCH μεταφέρει τις πληροφορίες που περιέχονται στο MIB, που είναι απαραίτητες σε ένα UE προκειμένου να συνδεθεί στην κυψέλη. Το PBCH είναι ειδικά σχεδιασμένο, ώστε ένα UE να μπορεί να διαβάσει τις πληροφορίες του MIB, ακόμα και αν δεν γνωρίζει το συνολικό εύρος ζώνης που χρησιμοποιείται. Για το σκοπό αυτό, το

PBCH μεταφέρεται πάντα στα 72 κεντρικά υποφέροντα (γύρω από τη φέρουσα συχνότητα). Επιπλέον, επειδή οι πληροφορίες που μεταφέρει το PBCH είναι κρίσιμες, το κανάλι αυτό διαμορφώνεται με το πιο εύρωστο δυνατό σχήμα, δηλαδή το QPSK. Το κανάλι PRACH μεταφέρει τις πληροφορίες του καναλιού μεταφοράς RACH.

4.4.4 Φυσικά Κανάλια Ελέγχου

Στο φυσικό στρώμα χρησιμοποιούνται ορισμένες πληροφορίες ελέγχου, οι οποίες είναι απαραίτητες για την σωστή αποκωδικοποίηση ενός πλαισίου LTE στον παραλήπτη, ενώ υποστηρίζουν επίσης βασικές λειτουργίες του φυσικού στρώματος. Τα πεδία ελέγχου του φυσικού στρώματος συνοψίζονται στον Πίνακα 4.5.

Πεδίο Ελέγχου		**Είδος Πληροφορίας**	**Κατεύθυνση ζεύξης**
UCI	Uplink Control Information	Επιβεβαίωση HARQ	UL
		Δείκτης ποιότητας διαύλου (CQI)	
		Δείκτης πίνακα προ-κωδικοποίησης (PMI)	
		Δείκτης τάξης διαύλου (RI)	
		Αιτήματα Χρονοπρογραμματισμού (SR)	
DCI	Downlink Control Information	Εντολές Χρονοπρογραμματισμού για την κάτω ζεύξη	DL
		Εκχωρήσεις ραδιοπόρων για την άνω ζεύξη	
		Εντολές ελέγχου ισχύος των UE	
CFI	Control Format Indicator	Μέγεθος περιοχής ελέγχου (1/2/3 OFDM σύμβολα στην αρχή κάθε υποπλαισίου)	DL
HI	Hybrid ARQ Indicator	Επιβεβαίωση HARQ	DL

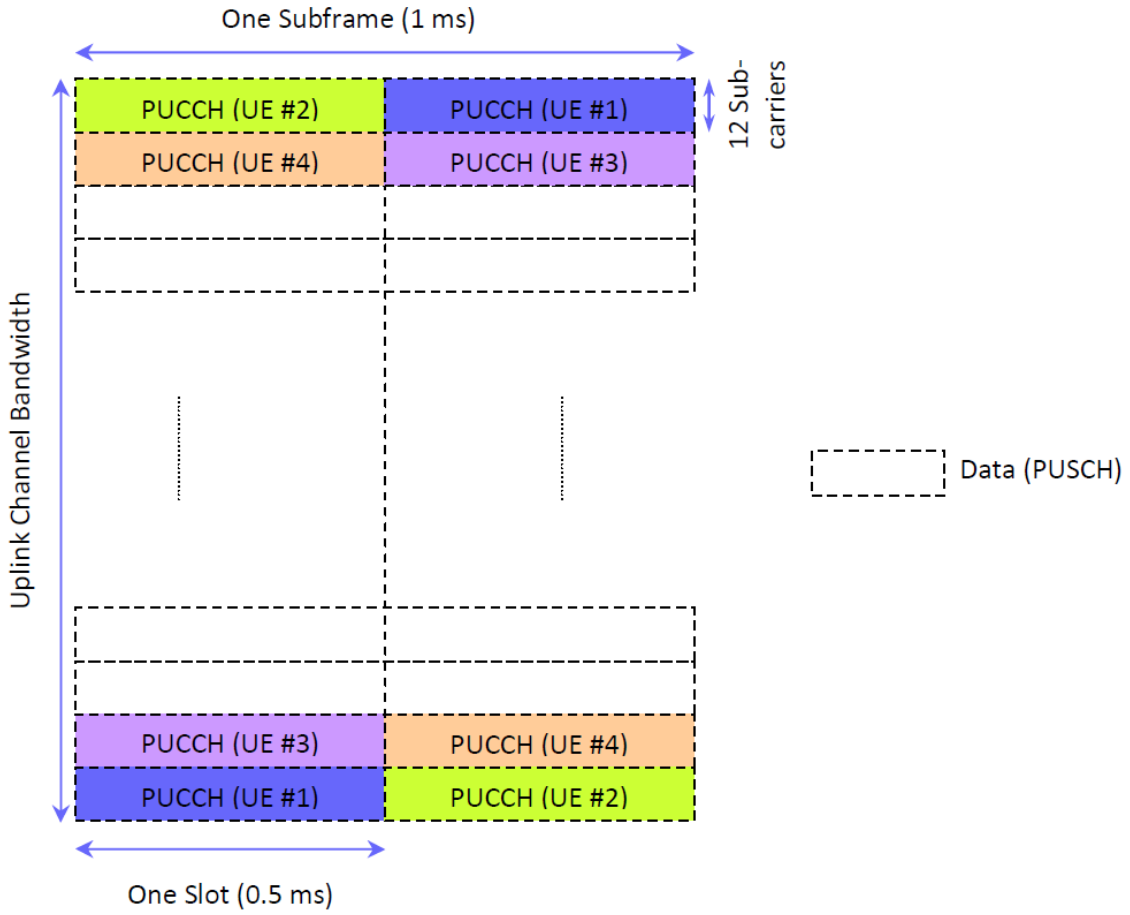
Πίνακας 4.5 *Πεδία ελέγχου φυσικού στρώματος LTE*

Τα φυσικά κανάλια ελέγχου χρησιμοποιούνται για την μεταφορά των πληροφοριών ελέγχου. Συνεπώς, τα φυσικά κανάλια ελέγχου PCFICH, PHICH και PDCCH (βλ. Πίνακα 4.6) μεταφέρουν τις πληροφορίες ελέγχου CFI, HI και DCI αντίστοιχα. Το κανάλι PCFICH χρησιμοποιεί το σχήμα διαμόρφωσης QPSK, καθώς είναι απαραίτητο το πεδίο CFI να μεταφερθεί χωρίς λάθη στο UE. Το κανάλι PHICH μεταφέρει τα μηνύματα HARQ του eNB προς το UE και χρησιμοποιεί το σχήμα διαμόρφωσης BPSK. Οι πληροφορίες του καναλιού PDCCH προορίζονται αποκλειστικά για το UE και χρησιμοποιείται το σχήμα διαμόρφωσης QPSK. Το πλήθος των OFDM συμβόλων που χρησιμοποιούνται σε κάθε υποπλαίσιο για τη μετάδοση του PDCCH, καθορίζεται από το PCFICH.

Στην περίπτωση της άνω ζεύξης, το κανάλι PUCCH χρησιμοποιείται μόνο όταν το UE εκπέμπει αποκλειστικά πληροφορίες ελέγχου. Αν ταυτόχρονα εκπέμπει και δεδομένα, τότε όλη η πληροφορία μεταφέρεται από το κανάλι PUSCH. Το κανάλι PUCCH χρησιμοποιεί την περιοχή συχνοτήτων στα άκρα του εύρους ζώνης, όπως απεικονίζεται στο Σχήμα 4.21. Στη διάρκεια ενός υποπλαισίου, αναθέτονται σε ένα χρήστη δύο RBs, τα οποία βρίσκονται σε αντιδιαμετρικές θέσεις του πλέγματος, ώστε να επιτυγχάνεται διαφορισμός (diversity) στο πεδίο της συχνότητας. Στο κανάλι PUCCH χρησιμοποιούνται τα σχήματα διαμόρφωσης BPSK και QPSK.

Φυσικό Κανάλι Ελέγχου		Είδος Πληροφορίας	Κατεύθυνση ζεύξης
PUCCH	Physical Uplink Control Channel	UCI	UL
PCFICH	Physical Control Format Indicator Channel	CFI	DL
PHICH	Physical Hybrid ARQ Indicator Channel	HI	DL
PDCCH	Physical Downlink Control Channel	DCI	DL

Πίνακας 4.6 *Είδη φυσικών καναλιών ελέγχου LTE*



Σχήμα 4.21 *Η δομή του φυσικού καναλιού ελέγχου PUCCH* [46]

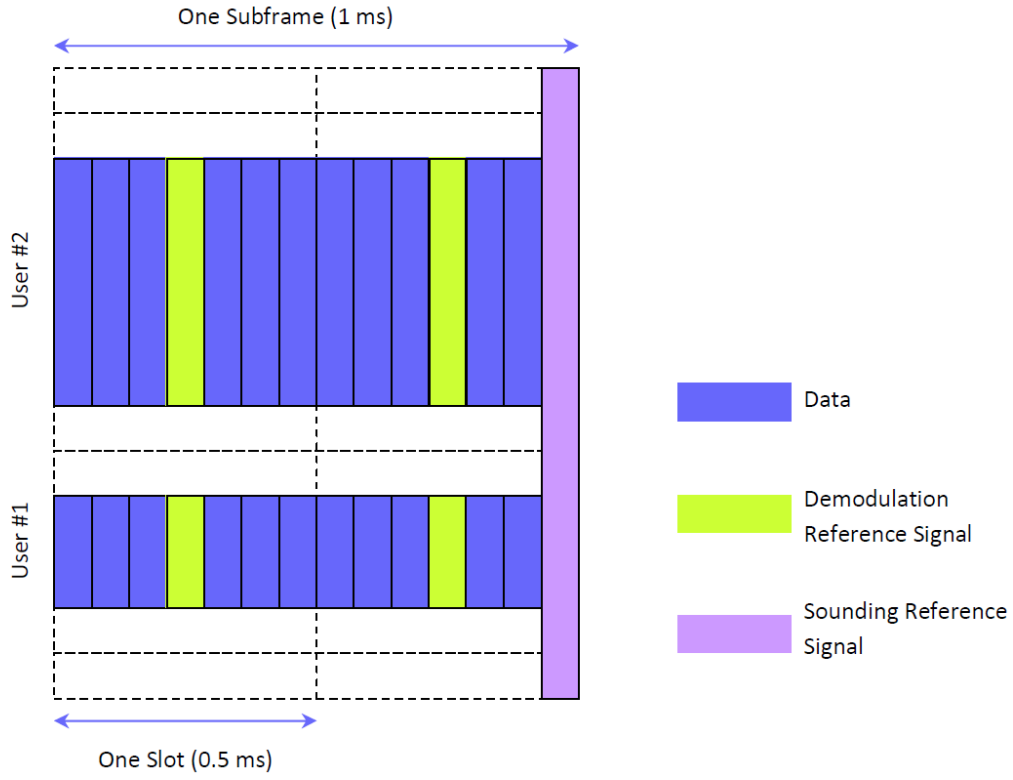
4.5 Φυσικά Σήματα LTE

Εκτός από τα πεδία ελέγχου που χρησιμοποιούνται στο φυσικό στρώμα, στα συστήματα LTE ορίζονται και ορισμένα φυσικά σήματα, τα οποία υποστηρίζουν αποκλειστικά τις λειτουργίες του φυσικού στρώματος. Τα φυσικά σήματα συνοψίζονται στον Πίνακα 4.7. Στην άνω ζεύξη υπάρχουν δύο σήματα αναφοράς: το DRS (Demodulation Reference Signal) και το SRS (Sounding Reference Signal). Το σήμα DRS εκπέμπεται στο φυσικό κανάλι PUSCH, και χρησιμοποιείται για την εκτίμηση της επίδρασης του διαύλου στη ζώνη συχνοτήτων που εκπέμπει το UE (βλ. Σχήμα 4.22).

Φυσικό Σήμα		Χρησιμότητα	Κατεύθυνση ζεύξης
DRS	Demodulation Reference Signal	Εκτίμηση διαύλου	UL
SRS	Sounding Reference Signal	Εκτίμηση διαύλου για το συνολικό εύρος ζώνης	UL
PSS	Primary Synchronization Signal	Cell Acquisition	DL
SSS	Secondary Synchronization Signal	Cell Acquisition	DL
RS	Cell-Specific Reference Signal	Εκτίμηση διαύλου	DL

Πίνακας 4.7 *Φυσικά σήματα LTE*

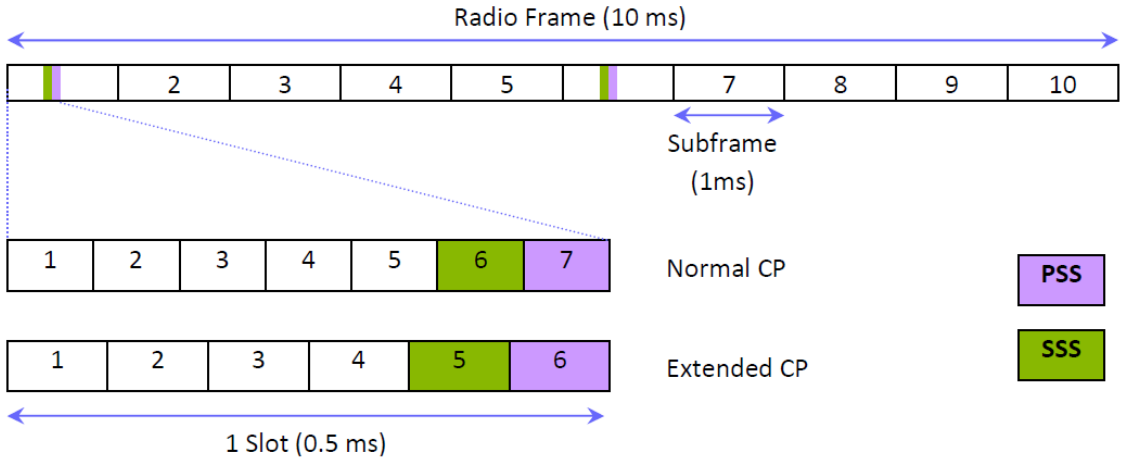
Προκειμένου ο eNB να έχει γνώση της επίδρασης του διαύλου σε όλο το διαθέσιμο εύρος ζώνης, χρησιμοποιείται το σήμα SRS. Το SRS καταλαμβάνει μεγαλύτερο εύρος συχνοτήτων, συγκριτικά με το DRS, όπως απεικονίζεται στο Σχήμα 4.22. Η χρήση του SRS δίνει τη δυνατότητα στον eNB να καθορίζει δυναμικά το χρονοπρογραμματισμό στην άνω ζεύξη, σύμφωνα με το ποια ζώνη συχνοτήτων εμφανίζει καλύτερη ποιότητα σήματος λήψης, για μία δεδομένη χρονική στιγμή (frequency selective UL scheduling).



Σχήμα 4.22 *Φυσικά σήματα DRS και SRS της ζεύξης UL* [46]

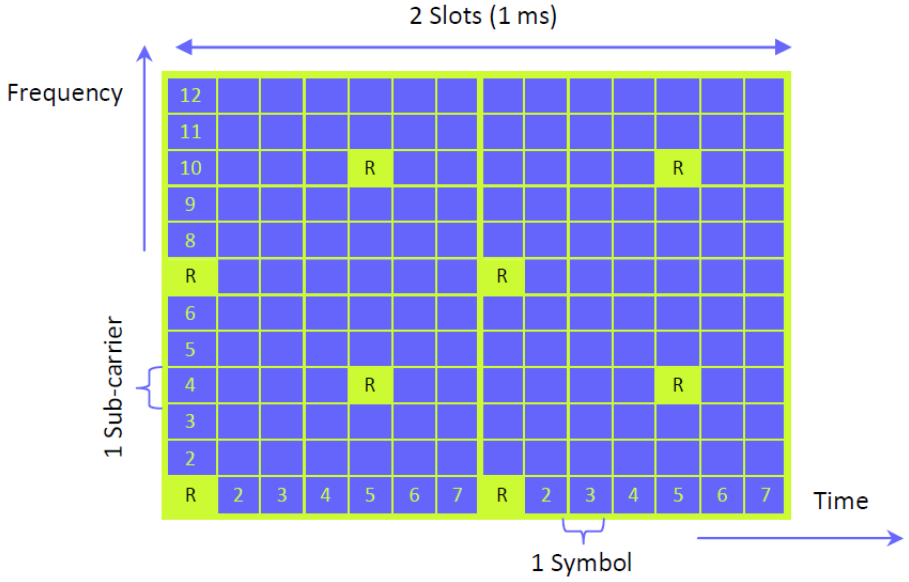
Προκειμένου ένα UE να συνδεθεί σε κάποια κυψέλη ενός δικτύου LTE, χρησιμοποιεί δύο ειδικά σήματα συγχρονισμού που εκπέμπονται από τον eNB: το πρωτεύον (PSS) και το δευτερεύον (SSS) σήμα συγχρονισμού. Με την ανίχνευση αυτών των σημάτων, το UE αποκτά σημαντικές παραμέτρους του συστήματος, όπως την ταυτότητα της κυψέλης (Physical Cell Identity - PCI), το είδος του cyclic prefix και τον τρόπο πρόσβασης (FDD/TDD). Στο στάδιο αυτό το UE αποκωδικοποιεί επίσης το κανάλι PBCH, λαμβάνοντας τις πληροφορίες που περιέχονται στο MIB. Τα σήματα συγχρονισμού

αποστέλλονται δύο φορές μέσα σε κάθε πλαίσιο LTE, στην 1η και 11η σχισμή αντίστοιχα (βλ. Σχήμα 4.23). Στο πεδίο της συχνότητας, τα PSS και SSS καταλαμβάνουν πάντα τα 62 κεντρικά υποφέροντα γύρω από τη φέρουσα συχνότητα, προκειμένου ένα UE να μπορεί να συνδεθεί σε κάποια κυψέλη, χωρίς να γνωρίζει εξαρχής το συνολικό εύρος ζώνης που χρησιμοποιείται [52].



Σχήμα 4.23 *Σήματα συγχρονισμού PSS και SSS στη διάρκεια ενός πλαισίου LTE* [46]

Τέλος, τα σήματα αναφοράς στην κάτω ζεύξη (RS) χρησιμοποιούνται για την εκτίμηση του διαύλου και τη σύμφωνη αποδιαμόρφωση (demodulation) στα UEs. Τα σήματα αναφοράς τοποθετούνται σε συγκεκριμένες θέσεις στο χρόνο και τη συχνότητα, όπως απεικονίζεται στο Σχήμα 4.24. Κάθε RB του πλέγματος ραδιοπόρων περιέχει τέσσερα σήματα αναφοράς. Ένα UE πραγματοποιεί γραμμική παρεμβολή μεταξύ όλων των σημάτων αναφοράς της ίδιας χρονικής στιγμής, ώστε να αποκτήσει μία εκτίμηση της επίδρασης του διαύλου στο συνολικά χρησιμοποιούμενο εύρος ζώνης.



Σχήμα 4.24 *Σήματα αναφοράς (RS) σε ένα RB στην κατεύθυνση DL* [46]

Για την καλύτερη κατανόηση του τρόπου χρήσης της τεχνικής OFDMA, των μετασχηματισμών IFFT/FFT, της διαμόρφωσης/αποδιαμόρφωσης OFDM συμβόλων και των σημάτων αναφοράς RS, ο αναγνώστης παραπέμπεται στο [53], που περιέχει μία κομψή και απλή υλοποίηση των ανωτέρω θεμάτων στη γλώσσα προγραμματισμού *Python*.

Κεφάλαιο 5. Mobile Optimization

Η ευρεία ανάπτυξη των συστημάτων LTE, όπως αυτά παρουσιάστηκαν στο Κεφάλαιο 4, επέφερε ανάλογη αύξηση στους χρήστες των ασύρματων δικτύων. Σύμφωνα με στοιχεία της Cisco, ο αριθμός των συνδρομητών κινητής τηλεφωνίας παγκοσμίως το 2018 ήταν 5.1 δισεκατομμύρια [1], η πλειοψηφία των οποίων συνδέεται στο Internet και καταναλώνει δεδομένα κινητής τηλεφωνίας (mobile data). Ωστόσο, παρά την έκρηξη στη δημοφιλία των κινητών συσκευών, τα πρωτόκολλα μεταφοράς του Internet, και ειδικά το TCP, παραμένουν άρρηκτα συνδεδεμένα με τα ενσύρματα δίκτυα για τα οποία αρχικά σχεδιάστηκαν. Καίτοι το TCP λειτουργεί και πάνω από ένα ασύρματο δίκτυο, τα δύο είδη δικτύων παρουσιάζουν ουσιώδεις διαφορές στις ιδιότητες και τα χαρακτηριστικά τους. Για παράδειγμα, οι τυχαίες και απότομες διακυμάνσεις του διαθέσιμου bandwidth λόγω διαλείψεων μικρής ή μεγάλης κλίμακας αποτελούν εγγενές χαρακτηριστικό μόνο των ασύρματων δικτύων. Ως εκ τούτου, σύμφωνα με πρόσφατες μελέτες, οι συνδέσεις TCP σε ένα ασύρματο δίκτυο δεν αξιοποιούν στο μέγιστο το διαθέσιμο bandwidth, με αποτέλεσμα να μην επιτυγχάνεται βέλτιστη επίδοση του TCP [54].

Το γεγονός ότι το TCP εμφανίζει χαμηλό βαθμό χρησιμοποίησης (utilization) των διαθέσιμων πόρων του ασύρματου δικτύου οδηγεί σε δύο αρνητικά επακόλουθα. Πρώτον, επειδή η πλειοψηφία των mobile εφαρμογών (π.χ. video streaming) λειτουργούν πάνω από το TCP, επηρεάζεται και η επίδοση των εφαρμογών, με αποτέλεσμα μεγαλύτερους χρόνους εμφάνισης ιστοσελίδας (Page Load Time - PLT) και περισσότερες παύσεις σε video συνεχούς ροής (video stalls). Δεύτερον, καθώς μία κυψέλη ενός δικτύου εξυπηρετεί πολλούς χρήστες, η αδυναμία αξιοποίησης όλου του διαθέσιμου bandwidth σε δεδομένη χρονική στιγμή μετατοπίζει φόρτο εργασίας (workload) σε επόμενη χρονική στιγμή, στην οποία συνυπάρχει και μελλοντικός φόρτος εργασίας. Το αποτέλεσμα αυτό ισοδυναμεί με μείωση της χωρητικότητας του δικτύου (network capacity) [55], για την οποία οι τηλεπικοινωνιακοί πάροχοι δαπανούν μεγάλα χρηματικά ποσά προς αναβάθμιση των υποδομών του δικτύου.

Στο [56] γίνεται ευρεία επισκόπηση των λύσεων που έχουν προταθεί και υλοποιηθεί, προκειμένου να βελτιωθεί η επίδοση του TCP στο περιβάλλον των δικτύων κινητής επικοινωνίας. Στην §5.1 παρουσιάζεται η πλέον διαδεδομένη τεχνική για τη βελτιστοποίηση του TCP, γνωστή ως TCP Acceleration.

5.1 TCP Acceleration

Το πρωτόκολλο TCP δεν διαθέτει πληροφορίες για το είδος του δικτύου (ενσύρματο, ασύρματο) και για το διαθέσιμο εύρος ζώνης μιας σύνδεσης TCP. Εξαιτίας της έλλειψης αυτών των πληροφοριών, το TCP προσαρμόζεται αργά στις μεταβολές του δικτύου, όπως θα αναλυθεί στη συνέχεια. Όταν το TCP λειτουργεί στη βέλτιστη κατάσταση, το παράθυρο συμφόρησης είναι ίσο με το γινόμενο του διαθέσιμου εύρους ζώνης (bandwidth) και της καθυστέρησης (RTT) της σύνδεσης TCP. Το γινόμενο αυτό συμβολίζεται ως BDP (Bandwidth - Delay Product) και δίνεται από την εξίσωση:

$$BDP\ (bits) = Bandwidth\left(\frac{bits}{seconds}\right) \cdot RTT(\text{seconds}) \quad (5.1)$$

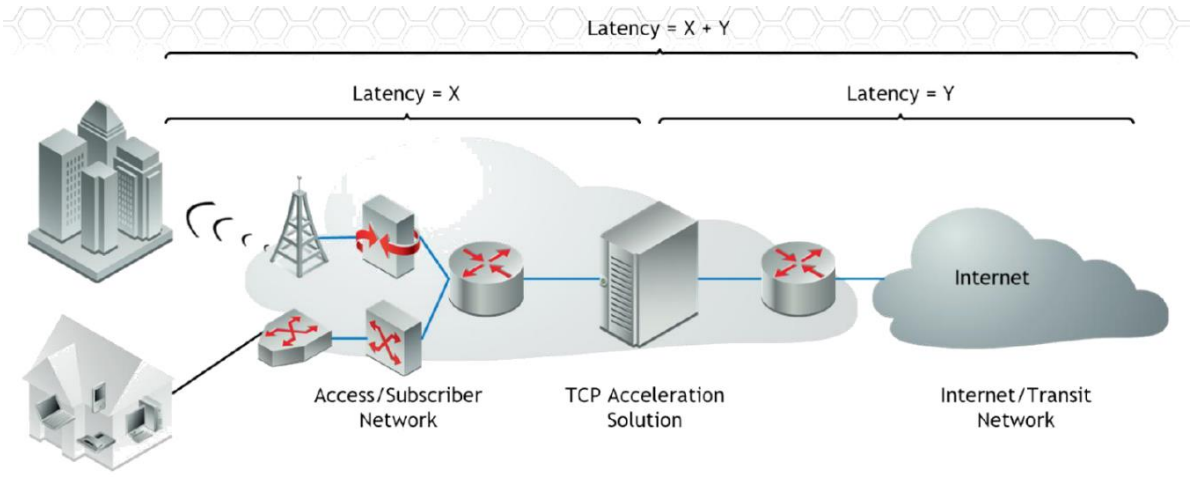
Το γινόμενο BDP αποτελεί τη μέγιστη ποσότητα δεδομένων που μπορεί να διακινείται στην εκάστοτε σύνδεση TCP, χωρίς η ποσότητα αυτή να έχει επιβεβαιωθεί. Εξαιτίας της έλλειψης πληροφοριών περί των χαρακτηριστικών του δικτύου, υπάρχουν περιπτώσεις όπου το παράθυρο συμφόρησης είναι αρκετά μικρότερο από το BDP και περιπτώσεις όπου είναι μεγαλύτερο από το BDP. Κάποια χρονική στιγμή για παράδειγμα, είναι ενδεχόμενο το δίκτυο να έχει μεγαλύτερο εύρος ζώνης διαθέσιμο αλλά η διαδικασία προσαρμογής σε αυτό να είναι αργή, οπότε υποχρησιμοποιείται το διαθέσιμο εύρος ζώνης. Παράλληλα, το TCP πραγματοποιεί συντηρητική αντιμετώπιση όταν υπάρχουν απώλειες πακέτων καθώς θεωρεί ότι είναι ενδείξεις συμφόρησης. Αυτό έχει αρνητική επίδραση σε κάποιο συνδρομητή κινητής τηλεφωνίας, ο οποίος λόγω μειωμένης ισχύος λήψης σε δεδομένη στιγμή δέχεται μείωση της ταχύτητας μετάδοσής του, η οποία, στη συνέχεια, καθυστερεί να αυξηθεί μέχρι την τιμή του διαθέσιμου εύρους ζώνης. Αντίστοιχα, υπάρχουν περιπτώσεις όπου το TCP πραγματοποιεί αποστολή δεδομένων με αυξημένους ρυθμούς που προκαλούν υπερχείλιση των ενταμιευτών στους διάφορους κόμβους του δικτύου. Η υπερχείλιση των ενταμιευτών των διαφόρων κόμβων μπορεί να οφείλεται είτε σε απότομη αύξηση του επιπέδου της δικτυακής κίνησης είτε σε ξαφνική αποστολή μεγάλου όγκου δεδομένων σε μικρό χρονικό διάστημα (micro-burst). Η συσσώρευση δεδομένων στους ενταμιευτές προκαλεί απόρριψη πακέτων, που παρερμηνεύεται από το TCP ως συμφόρηση στο δίκτυο. Συνεπώς, εξαιτίας της υπερχείλισης των ενταμιευτών λόγω των δύο προαναφερθέντων φαινομένων, μειώνονται οι ταχύτητες μετάδοσης των χρηστών χωρίς, στη πραγματικότητα, να έχει προκληθεί συμφόρηση στο δίκτυο.

Η τεχνική TCP Acceleration αποσκοπεί στην αύξηση του μέσου throughput των συνδέσεων TCP ενός ασύρματου δικτύου, χωρίς να χρειάζεται οποιαδήποτε αλλαγή στο λειτουργικό (OS) των τερματικών συστημάτων. Για το σκοπό αυτό, χρησιμοποιείται ειδική μονάδα, που ονομάζεται TCP Accelerator και τοποθετείται στο δίκτυο του τηλεπικοινωνιακού παρόχου. Πρέπει να τονιστεί ότι ο TCP Accelerator τοποθετείται σε ειδική θέση εντός του δικτύου, ώστε να παρακολουθεί και να διαχειρίζεται όλη την κίνηση TCP που διέρχεται μέσω αυτού. Ο TCP Accelerator λειτουργεί ως διακομιστής μεσολάβησης (proxy server), καθώς διαιρεί μία σύνδεση TCP σε δύο διακριτά τμήματα: το ένα είναι μεταξύ του δικτύου πρόσβασης (access network) και του TCP Accelerator, ενώ το άλλο είναι μεταξύ του TCP Accelerator και εξωτερικών δικτύων (π.χ. Internet). Ο TCP Accelerator επικοινωνεί με τον TCP server, που βρίσκεται σε κάποιο εξωτερικό δίκτυο, εκ μέρους του TCP client και το αντίστροφο. Η τοποθέτηση του TCP Accelerator στο δίκτυο κορμού του τηλεπικοινωνιακού παρόχου απεικονίζεται στο Σχήμα 5.1.

5.1.1 Επιτάχυνση της Φάσης Αργής Εκκίνησης

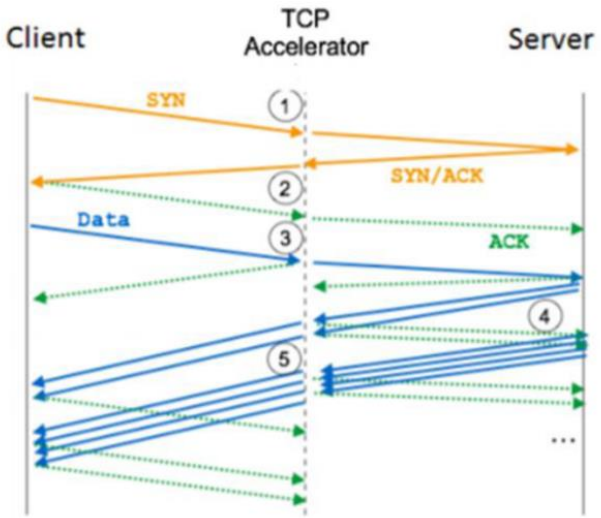
Όπως παρουσιάστηκε στο Κεφάλαιο 2, ένας αποστολέας TCP αυξάνει το παράθυρο συμφόρησης *cwnd*, άρα και το ρυθμό μετάδοσης δεδομένων *R*, όταν λαμβάνει επιβεβαιώσεις ACK για τμήματα TCP που έχουν σταλεί. Για παράδειγμα, στη φάση αποφυγής συμφόρησης, το παράθυρο συμφόρησης αυξάνεται γραμμικά κατά 1*MSS* κάθε RTT. Στην περίπτωση όπου η σύνδεση TCP παρουσιάζει υψηλή από-άκρο-σε-άκρο καθυστέρηση (end to end delay), ο ρυθμός μετάδοσης δεδομένων στον αποστολέα TCP

αυξάνεται με αργό ρυθμό, με αποτέλεσμα οι διαθέσιμοι ραδιοπόροι του δικτύου να μην αξιοποιούνται στο έπακρο.



Σχήμα 5.1 *Τοποθέτηση TCP Accelerator εντός του δικτύου του παρόχου* [57]

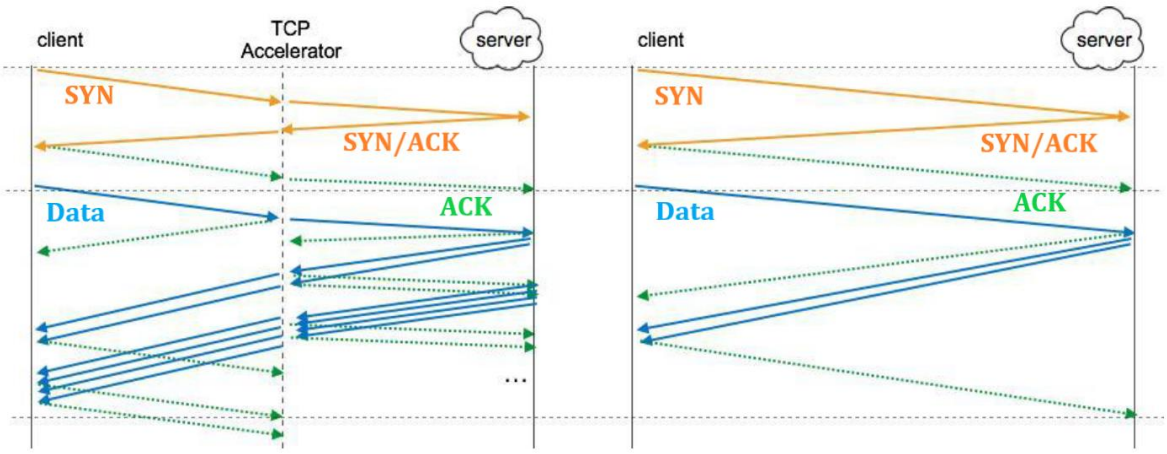
Ο χρόνος προσαρμογής του ρυθμού μετάδοσης μιας σύνδεσης TCP στο διαθέσιμο εύρος ζώνης είναι καθοριστικός για την ποιότητα υπηρεσίας των χρηστών ενός δικτύου. Η τεχνική TCP Acceleration συμβάλλει στη μείωση του χρόνου αυτού μέσω της πρώιμης αποστολής επιβεβαιώσεων ACK από τον TCP Accelerator προς τον TCP server, εκ μέρους του TCP client. Ο TCP Accelerator επιτυγχάνει ταχύτερη αύξηση του ρυθμού μετάδοσης στη σύνδεση TCP, καθώς πρακτικά χωρίζει τη συνολική καθυστέρηση σε δύο μικρότερες και ανεξάρτητες μεταξύ τους καθυστερήσεις (*Latency X* και *Latency Y* σύμφωνα με το Σχήμα 5.1). Επειδή ο TCP server λαμβάνει πρώιμες επιβεβαιώσεις ACK από τον TCP Accelerator, αποστέλλει νέα δεδομένα νωρίτερα επιτυγχάνοντας έτσι την αύξηση του παραθύρου συμφόρησης *cwnd*. Συνεπώς, ο TCP Accelerator επιταχύνει πρακτικά τη φάση της αργής εκκίνησης του TCP μέσω της μείωσης του χρόνου επιβεβαίωσης των τμημάτων TCP. Με την παρέμβαση του TCP Accelerator, η διαδικασία αργής εκκίνησης τροποποιείται όπως φαίνεται στο Σχήμα 5.2.



Σχήμα 5.2 *Επιτάχυνση της φάσης της αργής εκκίνησης του TCP* [58]

Με αναφορά στο Σχήμα 5.2, ο TCP Accelerator παρακολουθεί τη διαδικασία τριπλής χειραψίας ανάμεσα στον πελάτη και τον εξυπηρετητή, προωθώντας τα πακέτα που φθάσουν σε εκείνον στην αντίστοιχη πλευρά της σύνδεσης TCP (1-2). Μόλις ολοκληρωθεί η διαδικασία αυτή, η σύνδεση έχει εγκατασταθεί και ο TCP Accelerator έχει όλες τις απαραίτητες πληροφορίες για να εκπροσωπεί τον πελάτη στην επικοινωνία του με τον εξυπηρετητή και αντίστροφα. Συνεπώς, ο TCP Accelerator μπορεί να επιβεβαιώνει τη λήψη δεδομένων εκ μέρους του παραλήπτη, είτε αυτός είναι ο πελάτης είτε αυτός είναι ο εξυπηρετητής (3). Τα πακέτα επιβεβαίωσης ACK δημιουργούνται στον TCP Accelerator και περιέχουν στην επικεφαλίδα πληροφορίες διαμορφωμένες από εκείνον (π.χ. παράθυρο λήψης, αριθμοί ακολουθίας). Επιπλέον, το πακέτο δεδομένων που αποστέλλεται από τον πελάτη στο σημείο αναφοράς (3), προωθείται στον εξυπηρετητή αφού αποθηκευτεί προσωρινά στον ενταμιευτή του TCP Accelerator. Η προσωρινή αποθήκευση των ανεπιβεβαίωτων πακέτων πραγματοποιείται καθώς μπορεί να χρειαστεί η αναμετάδοσή τους, η οποία στην περίπτωση αυτή θα γίνει με πολύ μικρότερη καθυστέρηση σε σχέση με την αναμετάδοσή τους από τον εξυπηρετητή. Καθώς οι επιβεβαιώσεις φθάνουν ταχύτερα στον εξυπηρετητή, πυροδοτούν την αποστολή δεδομένων με υψηλότερο ρυθμό (4), οπότε η διαδικασία της αργής εκκίνησης επιταχύνεται σημαντικά. Η αποστολή δεδομένων μπορεί να γίνεται ενόσω δεν έχει επιβεβαιωθεί η λήψη των προηγούμενων πακέτων (5), καθώς ο TCP Accelerator διαθέτει αρκετά μεγάλο παράθυρο συμφόρησης για να προσεγγίζει το διαθέσιμο εύρος ζώνης.

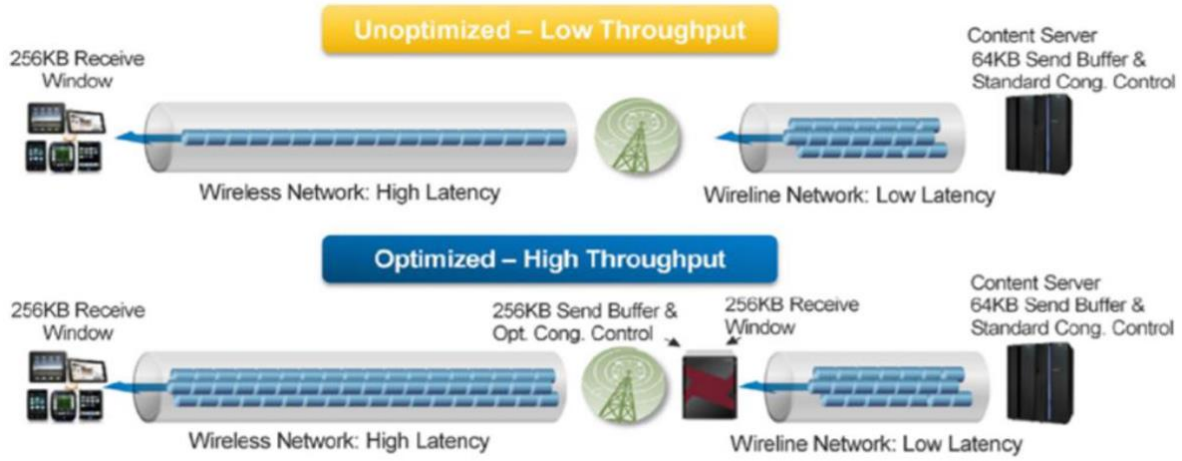
Στο Σχήμα 5.3 απεικονίζεται η σύγκριση της φάσης της αργής εκκίνησης με και χωρίς χρήση TCP Accelerator. Συγκρίνοντας τα δύο διαγράμματα για το ίδιο χρονικό διάστημα, όπως αυτό καθορίζεται από την τελευταία οριζόντια διακεκομμένη γραμμή, παρατηρείται ότι στην περίπτωση της επιταχυνόμενης σύνδεσης TCP, ο client λαμβάνει έξι τμήματα TCP που περιέχουν δεδομένα εφαρμογής. Αντίθετα, στην περίπτωση της συμβατικής σύνδεσης TCP, ο client λαμβάνει στο ίδιο χρονικό διάστημα μόνο δύο τμήματα TCP με δεδομένα εφαρμογής. Γίνεται, επομένως, εμφανής, η επιτάχυνση της φάσης της αργής εκκίνησης με τη χρήση του TCP Accelerator.



Σχήμα 5.3 *Σύγκριση της φάσης αργής εκκίνησης με και χωρίς χρήση TCP Accelerator* [57]

Στα ενσύρματα δίκτυα, η συνεισφορά του TCP Accelerator είναι σημαντική αλλά όχι καθοριστική. Τα ενσύρματα δίκτυα χαρακτηρίζονται από μικρή καθυστέρηση, σταθερό εύρος ζώνης, ενώ η κυριότερη αιτία απώλειας πακέτων είναι η συμφόρηση στο δίκτυο.

Τα παράθυρα συμφόρησης των εξυπηρετητών είναι συνήθως μικρά, ενώ τα παράθυρα λήψης των τερματικών είναι αρκετά μεγαλύτερα. Επομένως, στα ενσύρματα δίκτυα, η ασυμφωνία μεταξύ των παραθύρων συμφόρησης (εξυπηρετητή) και λήψης (τερματικών) δεν επηρεάζει σημαντικά τη μετάδοση δεδομένων, λόγω των ικανοποιητικών συνθηκών μετάδοσης. Αντίθετα, τα ασύρματα δίκτυα χαρακτηρίζονται από υψηλή καθυστέρηση, μεταβλητό εύρος ζώνης ενώ η απώλεια πακέτων μπορεί να οφείλεται είτε σε συμφόρηση του δικτύου είτε σε χαμηλή ισχύ λήψης στο δέκτη είτε σε συμφόρηση στην κυψέλη. Χωρίς την παρεμβολή του TCP Accelerator, η ασυμφωνία μεταξύ του παραθύρου συμφόρησης των εξυπηρετητών (μικρά παράθυρα) και του παραθύρου λήψης των τερματικών (μεγάλα παράθυρα) έχει ως αποτέλεσμα την υποβαθμισμένη ποιότητα υπηρεσίας, ιδιαίτερα στην περίπτωση εφαρμογών video streaming όπου απαιτείται υψηλό throughput και χαμηλή καθυστέρηση. Ο TCP Accelerator επιτυγχάνει τη βελτιστοποίηση της λειτουργίας και των δύο τμημάτων της σύνδεσης TCP (ασύρματο και ενσύρματο) και εξομοιώνει τα παράθυρα συμφόρησης και λήψης σε κάθε πλευρά. Συνεπώς, η λήψη δεδομένων από το χρήστη γίνεται με υψηλότερο ρυθμό και η αρνητική επίδραση της αυξημένης καθυστέρησης και του μεταβλητού εύρους ζώνης περιορίζεται σημαντικά. Στο Σχήμα 5.4 παρουσιάζεται σχηματικά η συμβολή του TCP Accelerator στη μετάδοση δεδομένων στα ασύρματα δίκτυα [58].

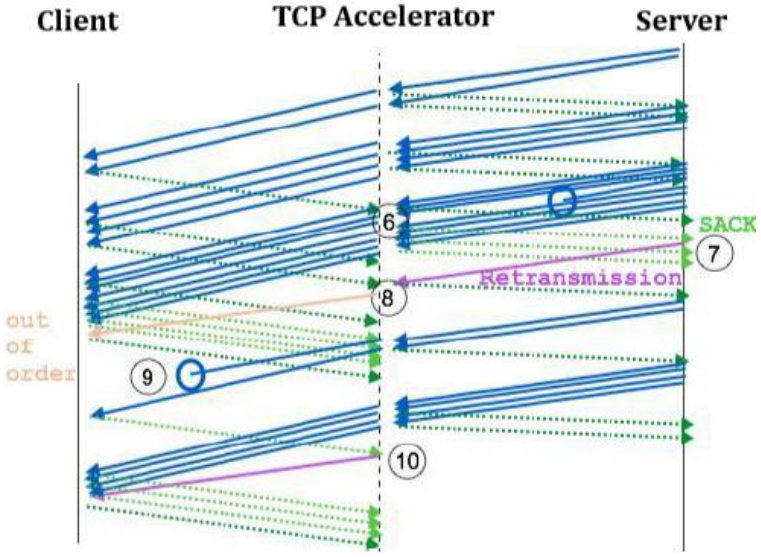


Σχήμα 5.4 *Συμβολή TCP Accelerator στην αύξηση της ρυθμαπόδοσης στα ασύρματα δίκτυα*

5.1.2 Αποδοτικότερη Διαχείριση Απωλειών Πακέτων

Όταν ο ρυθμός μετάδοσης του εξυπηρετητή (που καθορίζεται από το μέγεθος του παραθύρου συμφόρησης) φθάσει το διαθέσιμο εύρος ζώνης, η τιμή του πρέπει να διατηρηθεί, εφόσον υποστηρίζεται από το δίκτυο. Η διατήρηση του εύρους ζώνης αποσκοπεί στην καλύτερη δυνατή αξιοποίηση των διαθέσιμων τηλεπικοινωνιακών πόρων και την παροχή βέλτιστης ποιότητας υπηρεσίας του χρήστη. Όταν ανιχνεύσει μια απώλεια πακέτου, ο αλγόριθμος συμφόρησης του TCP (βλ. §2.6.3) μειώνει το παράθυρο συμφόρησης, καθώς θεωρεί ότι η απώλεια προκλήθηκε εξαιτίας του υψηλού φορτίου κίνησης στο δίκτυο. Ωστόσο, οι απώλειες πακέτων είναι συχνό φαινόμενο στα ασύρματα δίκτυα και σπάνια οφείλονται σε υψηλό φορτίο κίνησης. Αντίθετα, οι απώλειες πακέτων προκαλούνται κυρίως από τυχαίες διακυμάνσεις της ισχύος του σήματος λήψης στα UEs εξαιτίας διαλείψεων στον ασύρματο δίαυλο. Ο TCP Accelerator διασφαλίζει ότι δεν θα

υπάρχουν άσκοπες ελαττώσεις του ρυθμού μετάδοσης του TCP server. Αυτό επιτυγχάνεται με την απόκρυψη προς την πλευρά του server, των απωλειών που συμβαίνουν στο ασύρματο τμήμα της σύνδεσης TCP. Η λειτουργία αυτή απεικονίζεται στο Σχήμα 5.5, όπου τα μπλε βέλη αντιστοιχούν στα τμήματα TCP που αποστέλλει ο server. Αν κάποιο πακέτο χαθεί στο ενσύρματο τμήμα της σύνδεσης TCP, ο TCP Accelerator αποκρίνεται άμεσα με την αποστολή ενός τμήματος SACK και ο server αναμεταδίδει το χαμένο τμήμα TCP (γεγονότα 6 και 7). Αν ένα πακέτο χαθεί στο ασύρματο τμήμα, ο client αποστέλλει μία επιβεβαίωση SACK και ο TCP Accelerator αναμεταδίδει το χαμένο τμήμα TCP από τον buffer που διατηρεί (γεγονότα 9 και 10), χωρίς η απώλεια να γίνει αντιληπτή από τον TCP server [59]. Συνεπώς, ο TCP Accelerator παρεμποδίζει τη μείωση του παραθύρου συμφόρησης, άρα και του ρυθμού μετάδοσης στον εξυπηρετητή. Ταυτόχρονα, μειώνονται και οι άσκοπες αναμεταδόσεις πακέτων, ώστε το σύνολο των αναμεταδόσεων να ανταποκρίνεται ικανοποιητικά στο σύνολο των πραγματικά απολεσθέντων πακέτων.



Σχήμα 5.5 *Διαχείριση απωλειών πακέτων από τον TCP Accelerator* [60]

5.1.3 Packet Pacing

Η μεταβλητότητα του διαθέσιμου εύρους ζώνης που παρατηρείται κατά κύριο λόγο στα ασύρματα δίκτυα και η αργή προσαρμογή του TCP στις αλλαγές του εύρους ζώνης καθιστά τη χρήση του TCP Accelerator επιτακτική. Για παράδειγμα, όταν το διαθέσιμο εύρος ζώνης μειώνεται, ο αποστολέας πρέπει άμεσα να μειώσει το ρυθμό αποστολής του, προς αποφυγή σφαλμάτων μετάδοσης. Αντίθετα, όταν το διαθέσιμο εύρος ζώνης αυξηθεί, ο αποστολέας πρέπει, εφόσον αυτό είναι απαραίτητο για την παροχή της καλύτερης δυνατής ποιότητας υπηρεσίας στο χρήστη, να το χρησιμοποιήσει αμέσως. Ο TCP Accelerator, έχοντας γνώση της κατάστασης του δικτύου μέσω των μετρήσεων που πραγματοποιεί στις συνδέσεις των χρηστών, μπορεί να συμβάλει στην υλοποίηση των δύο προαναφερθεισών διαδικασιών. Ο ρυθμός με τον οποίο ο TCP Accelerator λαμβάνει τις επιβεβαιώσεις του client είναι μια ένδειξη για το εκτιμώμενο εύρος ζώνης στη σύνδεσή του. Έτσι, ανάλογα με το ρυθμό άφιξης των επιβεβαιώσεων σε εκείνον, ο TCP Accelerator καθορίζει το ρυθμό αποστολής δεδομένων στο client και τον προσαρμόζει

σε αλλαγές του διαθέσιμου εύρους ζώνης. Ο μηχανισμός αυτός έχει γίνει γνωστός ως Packet Pacing [61].

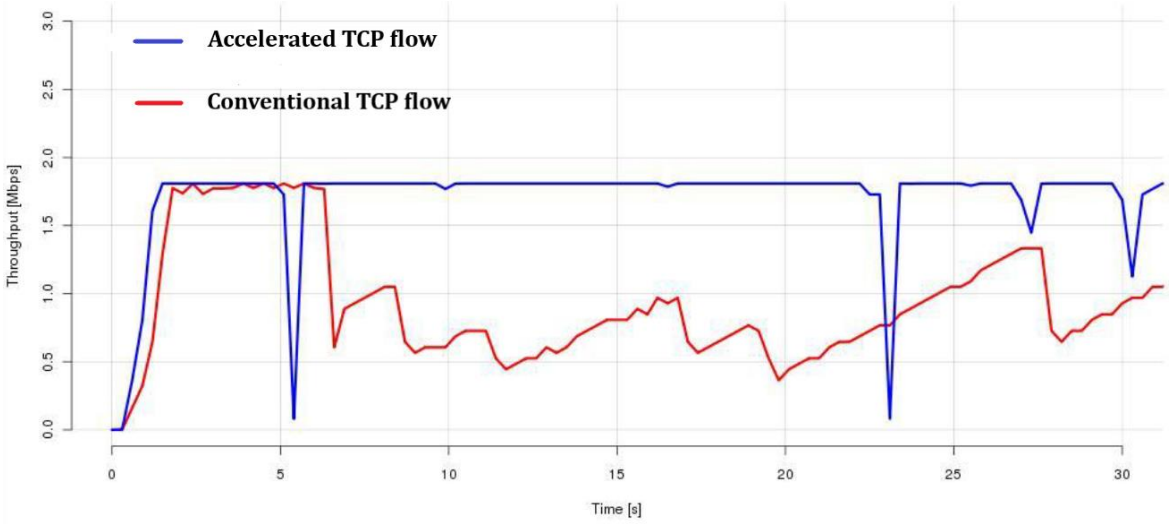
Στο Σχήμα 5.6 απεικονίζεται το throughput από δύο ροές TCP, που λειτουργούν εντός ενός ασύρματου δικτύου. Παρατηρείται ότι η επιταχυνόμενη (accelerated) σύνδεση TCP επιτυγχάνει μεγαλύτερο throughput, εμφανίζει λιγότερες διακυμάνσεις και ανακάμπτει ταχύτερα ύστερα από απώλειες πακέτων, συγκριτικά με μία συμβατική (conventional) σύνδεση TCP. Επιπλέον, όπως απεικονίζεται στο Σχήμα 5.7, η επιταχυνόμενη σύνδεση TCP παρουσιάζει μικρότερη μέση τιμή και διακύμανση του χρόνου RTT συγκριτικά με τη συμβατική σύνδεση TCP. Τέλος, η επιταχυνόμενη σύνδεση TCP εμφανίζει χαμηλότερο ποσοστό αναμεταδόσεων πακέτων από τη συμβατική σύνδεση TCP (βλ. Σχήμα 5.8). Τα οφέλη της χρήσης του TCP Accelerator συνοψίζονται ως εξής:

- Υψηλότεροι ρυθμοί μετάδοσης και αυξημένες επιδόσεις σε επίπεδο εφαρμογής (application performance) που οδηγούν σε βελτιωμένη ποιότητα υπηρεσίας (QoS).

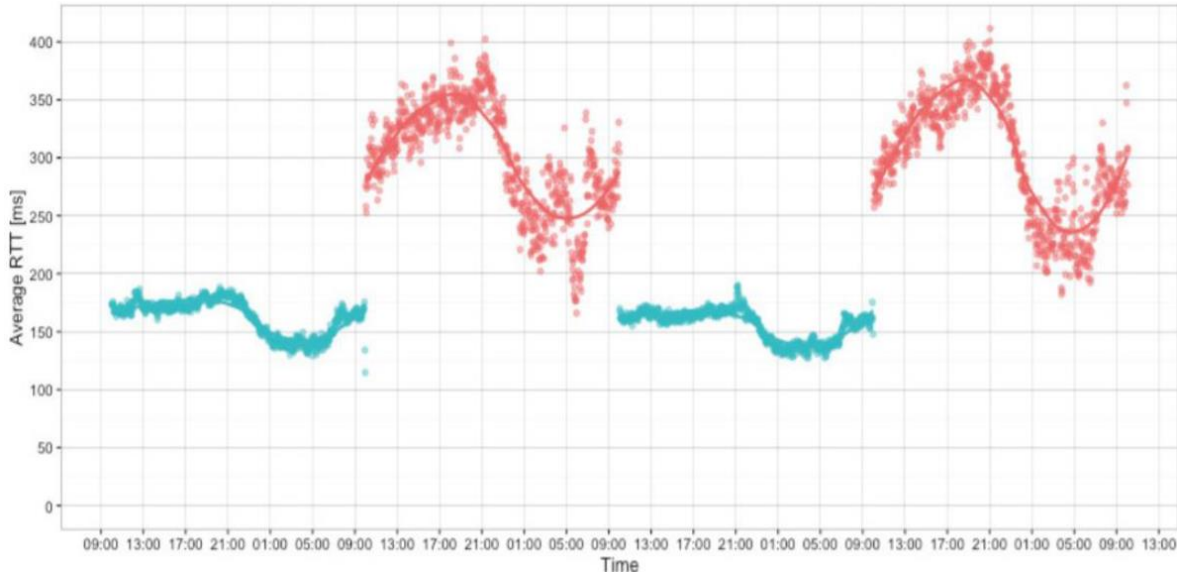
- Βελτιωμένη επίδοση του δικτύου, καθώς επιτυγχάνεται υψηλότερο ποσοστό χρησιμοποίησης (utilization) των διαθέσιμων πόρων και αύξηση της χωρητικότητας (capacity).

- Αύξηση των εσόδων του τηλεπικοινωνιακού παρόχου, καθώς συνδρομητές που απολαμβάνουν υψηλό QoS τείνουν να καταναλώνουν περισσότερα mobile data.

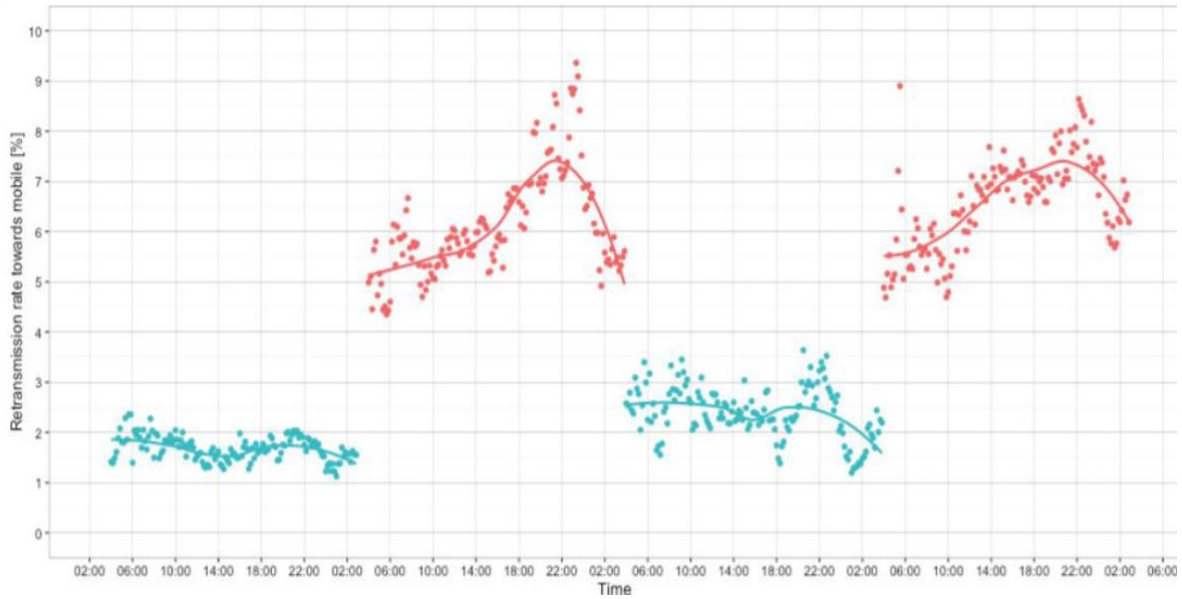
- Υψηλότερες τιμές του δείκτη ROI (Return on Investment), καθώς η αποδοτικότερη χρήση των υπαρχουσών υποδομών του δικτύου μεταθέτει το κόστος της αναβάθμισής τους στο μέλλον.



Σχήμα 5.6 *Βελτίωση του throughput που επιτυγχάνει η τεχνική TCP Acceleration* [60]



Σχήμα 5.7 *Μέσοι χρόνοι RTT με ενεργοποιημένο (μπλε) και απενεργοποιημένο (κόκκινο) TCP Accelerator* [60]



Σχήμα 5.8 *Ποσοστό αναμεταδόσεων με ενεργοποιημένο (μπλε) και απενεργοποιημένο (κόκκινο) TCP Accelerator* [60]

5.2 Αξιοποίηση Πληροφορίας του Δικτύου RAN

Οι TCP Accelerators που χρησιμοποιούνται σήμερα εμφανίζουν ορισμένους περιορισμούς. Για παράδειγμα, ο TCP Accelerator δεν λαμβάνει πληροφορίες περί της κατάσταση της κυψέλης όπου ανήκει ένα UE. Η βέλτιστη διαχείριση της δικτυακής κίνησης που παράγουν οι χρήστες προϋποθέτει ότι έχει γίνει προηγουμένως βελτιστοποίηση της κατανομής των πόρων σε κάθε κυψέλη, λαμβάνοντας υπόψη το πλήθος των UE που η κάθε κυψέλη εξυπηρετεί. Σε τρέχουσες εκδόσεις των εμπορικά διαθέσιμων TCP Accelerators, αν ένα UE ανήκει σε μια κυψέλη με υψηλή συμφόρηση λόγω πολυάριθμων ταυτοχρόνων χρηστών, ο TCP Accelerator θα διαπιστώσει χαμηλό εύρος ζώνης στη σύνδεσή του (όπως και σε αυτές των υπόλοιπων UE που ανήκουν στη συγκεκριμένη κυψέλη) και θα μειώσει το ρυθμό μετάδοσης του. Ωστόσο, με κατάλληλη κατανομή των τηλεπικοινωνιακών πόρων στις κυψέλες, θα μπορούσε να αποφευχθεί η μείωση του ρυθμού μετάδοσης των χρηστών που ανήκουν σε κυψέλες με υψηλή συμφόρηση. Συγκεκριμένα, η αποδοτική κατανομή των πόρων στις κυψέλες προϋποθέτει τη διάθεση περισσότερου εύρους ζώνης στις κυψέλες με πολυάριθμους ενεργούς χρήστες και τη μείωση του εύρους ζώνης των κυψελών με μικρό πλήθος χρηστών ως αντιστάθμισμα.

Κάθε τηλεπικοινωνιακός πάροχος διαθέτει βάσεις δεδομένων σε κόμβους του ασύρματου δικτύου πρόσβασης RAN (επί ή πλησίον του σταθμού βάσης eNB), οι οποίες διαθέτουν μετρήσεις για διάφορες παραμέτρους που αφορούν τη λειτουργία του eNB. Ο TCP Accelerator θα μπορούσε να αξιοποιήσει αυτές τις μετρήσεις για περαιτέρω βελτιστοποίηση της λειτουργίας των συνδέσεων TCP, με εγκατάσταση στο δίκτυο RAN εξυπηρετητών MEC (Mobile Edge Computing). Συγκεκριμένα, ένας εξυπηρετητής MEC μπορεί να επικοινωνεί με τις βάσεις δεδομένων και να επεξεργάζεται τις διαθέσιμες μετρήσεις, ώστε να διαπιστώνει σε πραγματικό χρόνο (real time) το επίπεδο συμφόρησης των κυψελών. Με κριτήριο το ποσοστό της κίνησης των δεδομένων μέσα στη κυψέλη και τις δραστηριότητες των συνδρομητών, ο εξυπηρετητής MEC μπορεί περιοδικά να παρέχει πληροφορία στον TCP Accelerator για το μέγεθος συμφόρησης της κυψέλης. Η επικοινωνία του TCP Accelerator με τον εξυπηρετητή MEC μπορεί να χρησιμοποιεί το πρωτόκολλο RADIUS (Remote Authentication Dial-In User Service) [62], καθώς αυτό ήδη υποστηρίζεται από TCP Accelerators πολλών εταιριών [63]. Στη συνέχεια, ο TCP Accelerator μέσω κατάλληλου αλγορίθμου θα κατανέμει το εύρος ζώνης στους σταθμούς βάσης των κυψελών αντίστοιχα με το μέγεθος της συμφόρησης.

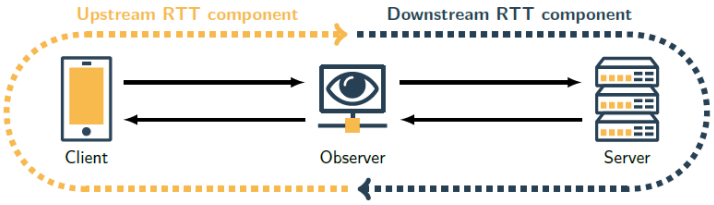
Μια άλλη πληροφορία του δικτύου RAN που θα μπορούσε να χρησιμοποιηθεί από το ασύρματο τμήμα μιας σύνδεσης TCP είναι η ισχύς λήψης των χρηστών. Στην περίπτωση όπου ένας χρήστης βρίσκεται μακριά από το σταθμό βάσης και το τερματικό του λειτουργεί υπό χαμηλή ισχύ λήψης, ο ρυθμός μετάδοσης προς το UE πρέπει να μειωθεί μέχρι η ισχύς λήψης του να επανέλθει σε τιμή μεγαλύτερη από κάποιο ελάχιστο κατώφλι, το οποίο θα ορίζει ο TCP Accelerator. Ωστόσο, η διαδικασία μείωσης του ρυθμού μετάδοσής του πυροδοτείται μόνο έπειτα από απώλεια σημαντικού αριθμού πακέτων με αποτέλεσμα την άσκοπη χρήση του εύρους ζώνης, εξαιτίας των αναμεταδόσεων χαμένων πακέτων. Παράλληλα, η διαδικασία αύξησης της ταχύτητας μετάδοσης προς το χρήστη όταν το τερματικό του αποκτήσει ικανοποιητική ισχύ λήψης καθυστερεί ακόμα και με τη χρήση του TCP Accelerator. Ο εξυπηρετητής MEC θα μπορούσε να συμβάλει στην επίλυση των προαναφερθέντων προβλημάτων μέσω της ενημέρωσης (με χρήση

του πρωτοκόλλου RADIUS) του TCP Accelerator για την ισχύ λήψης κάθε χρήστη. Έτσι, είναι εφικτή η παροχή υποδείξεων στον TCP Accelerator σχετικά με τις χρονικές στιγμές όπου συμβαίνει σημαντική αλλαγή στην ισχύ λήψης του χρήστη. Μέσω των υποδείξεων αυτών, ο TCP Accelerator θα μπορεί να προσαρμόζεται άμεσα στις αλλαγές της ισχύος λήψης του χρήστη, ρυθμίζοντας κατάλληλα το ρυθμό μετάδοσης δεδομένων προς εκείνον. Έπειτα από τη βέλτιστη κατανομή των τηλεπικοινωνιακών πόρων στις κυψέλες και τη βελτιστοποίηση ανά χρήστη με κριτήριο την ισχύ λήψης του, ο TCP Accelerator μπορεί να βελτιστοποιεί την ατομική ροή TCP μέσω των μηχανισμών της §5.1 [64]. Βέβαια, πρέπει να τονιστεί ότι η ισχύς λήψης των χρηστών προστατεύεται νομικά ως προσωπικό δεδομένο, καθώς μπορεί να χρησιμοποιηθεί για την ακριβή εύρεση της τοποθεσίας ενός χρήστη. Επομένως, η αξιοποίηση της ισχύς λήψης από τον TCP Accelerator ενδέχεται να απαγορεύεται από το νόμο και η ανωτέρω τεχνική να μην είναι πρακτικά εφαρμόσιμη. Ωστόσο, προκειμένου να αποφευχθεί ο ανωτέρω περιορισμός, μπορούν εναλλακτικά να τεθούν επίπεδα ισχύος λήψης και να μην χρησιμοποιείται η ακριβής τιμή.

5.3 QUIC Spin Bit Measurements

Οι διαχειριστές ενός δικτύου χρειάζεται να επιτηρούν συνεχώς την κατάστασή του, προκειμένου να ανιχνεύουν και να επιδιορθώνουν έγκαιρα οποιοδήποτε πρόβλημα, διασφαλίζοντας έτσι την ορθή λειτουργία του δικτύου. Οι παράμετροι που κυρίως επηρεάζουν την αντιληπτή ποιότητα υπηρεσίας ενός δικτύου είναι η ρυθμαπόδοση (throughput), η καθυστέρηση (delay) και ο ρυθμός απωλειών (loss rate). Οι τηλεπικοινωνιακοί πάροχοι χρησιμοποιούν σήμερα αρκετές τεχνικές προκειμένου να παρατηρούν τις ανωτέρω παραμέτρους των TCP συνδέσεων στο δίκτυό τους, καθώς οι συνδέσεις TCP αποτελούν την πλειοψηφία της κίνησης στο Internet. Για παράδειγμα, ο χρόνος RTT μίας σύνδεσης TCP μπορεί να υπολογιστεί από τους διαχειριστές του δικτύου, που λειτουργούν ως παθητικοί παρατηρητές, μέσω της παρατήρησης του πεδίου επιλογών των χρονοσφραγίδων (timestamps) της επικεφαλίδας TCP [65].

Όπως γίνεται αντιληπτό, μέθοδοι, όπως αυτή για τη μέτρηση του RTT, βασίζονται στο ότι η επικεφαλίδες του πρωτοκόλλου TCP είναι σε μορφή καθαρού κειμένου και όχι κρυπτογραφημένες. Ωστόσο, το πρωτόκολλο QUIC, που προβλέπεται να αντικαταστήσει το TCP στο κοντινό μέλλον, κρυπτογραφεί όλη την κίνηση που διακινείται σε μία σύνδεση QUIC, συμπεριλαμβανομένων των επικεφαλίδων. Επομένως, είναι απαραίτητο ορισμένα πεδία στην επικεφαλίδα QUIC να μην κρυπτογραφούνται, ώστε να επιτρέπεται στους διαχειριστές ενός δικτύου να πραγματοποιούν μετρήσεις για το χρόνο RTT και το ρυθμό απωλειών των συνδέσεων QUIC.



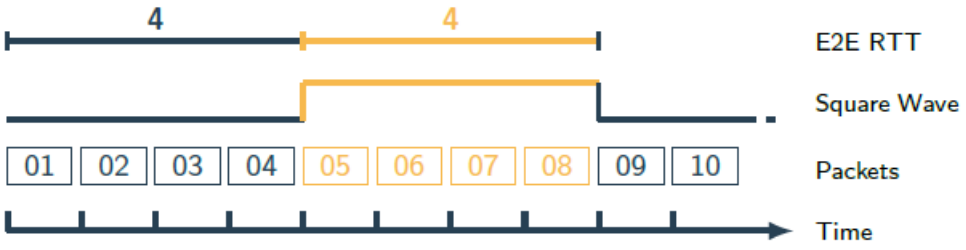
Σχήμα 5.9 *Μέτρηση του χρόνου RTT από έναν ενδιάμεσο παρατηρητή* [66]

Στην §3.3.3 αναλύθηκε η επικεφαλίδα τύπου short header των πακέτων QUIC. Μία short header περιέχει το επονομαζόμενο Spin Bit, το οποίο επιτρέπει τη μέτρηση του χρόνου RTT της σύνδεσης QUIC από κάποιον παθητικό παρατηρητή (π.χ. ο διαχειριστής ενός δικτύου). Η τιμή του Spin Bit (0 ή 1) αντιστρέφεται μία φορά κάθε RTT και στην αρχή κάθε σύνδεσης QUIC έχει την τιμή 0. Στη συνέχεια, η τιμή του Spin Bit καθορίζεται με βάση τους εξής κανόνες:

- Όταν ο πελάτης λαμβάνει ένα πακέτο QUIC με επικεφαλίδα τύπου short header, το οποίο έχει αριθμό πακέτου (packet number) μεγαλύτερο από το μέγιστο αριθμό πακέτου που έχει λάβει έως τότε, αντιστρέφει την τιμή του Spin Bit.

- Όταν ο εξυπηρετητής λαμβάνει ένα πακέτο QUIC με επικεφαλίδα τύπου short header, το οποίο αυξάνει το μέγιστο αριθμό πακέτου που έχει λάβει έως τότε, διατηρεί το Spin Bit στην ίδια τιμή που είχε στο πακέτο που λήφθηκε.

Επομένως, ο εξυπηρετητής δεν μεταβάλλει την τιμή του Spin Bit στα πακέτα που αποστέλλει, παρά μόνο όταν λάβει ένα πακέτο από τον πελάτη, το οποίο περιέχει την αντίστροφη τιμή Spin Bit από αυτή που τώρα χρησιμοποιεί. Αυτός ο απλός μηχανισμός επιτρέπει στα δύο τερματικά σημεία της σύνδεσης QUIC να παράγουν μία τετραγωνική κυματομορφή, όπως αυτή που απεικονίζεται στο Σχήμα 5.10. Ένας ενδιάμεσος παθητικός παρατηρητής μπορεί να υπολογίσει τον χρόνο RTT, ως τη χρονική διαφορά μεταξύ δύο διαδοχικών ακμών της κυματομορφής. Ως ακμές ορίζονται τα σημεία, στα οποία η κυματομορφή μεταβαίνει από χαμηλή (0) σε υψηλή (1) τιμή και αντίστροφα.



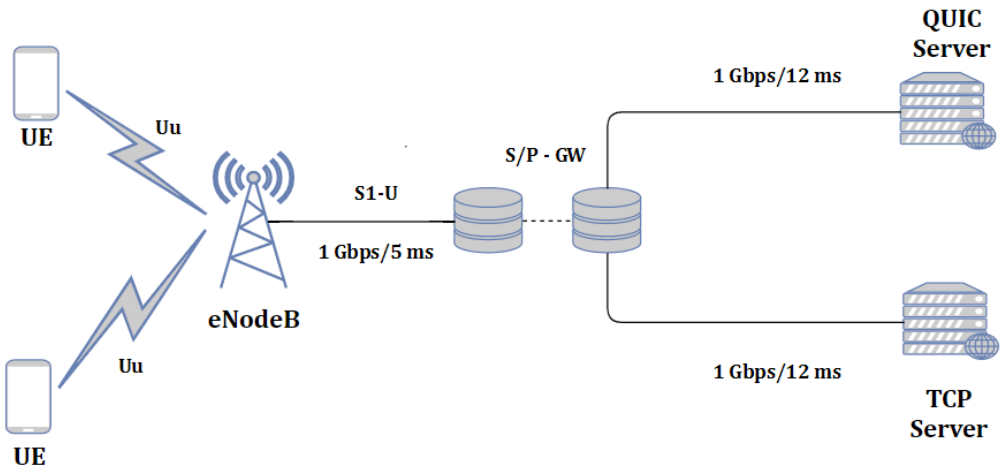
Σχήμα 5.10 *Η τετραγωνική κυματομορφή που παράγει ο μηχανισμός του Spin Bit* [66]

Κεφάλαιο 6. Επίδοση των Πρωτοκόλλων TCP και QUIC σε δίκτυο LTE

Στο κεφάλαιο αυτό συγκρίνεται μέσω προσομοιώσεων η επίδοση των πρωτοκόλλων TCP και QUIC, όταν λειτουργούν πάνω από ένα LTE δίκτυο. Για το σκοπό αυτό, χρησιμοποιήθηκε ο εξομοιωτής Network Simulator - 3.29 [67], ο οποίος επιτρέπει την υλοποίηση διαφόρων σεναρίων για επικοινωνία κόμβων πάνω από διάφορες αρχιτεκτονικές και τεχνολογίες δικτύων. Για το τμήμα του LTE χρησιμοποιήθηκε το LENA module (LTE-EPC Network simulAtor) του ns3 [68], που παρέχει μία ακριβή αναπαράσταση της στοίβας πρωτοκόλλων του LTE και των λειτουργιών του EPC, με το περιορισμό ότι το δίκτυο διαθέτει μοναδικό κόμβο S/P - GW.

6.1 Σενάριο Προσομοίωσης

Στο Σχήμα 6.1 απεικονίζεται το σενάριο που υλοποιήθηκε για την προσομοίωση. Για κάθε ζεύξη αναγράφεται ο μέγιστος ρυθμός μετάδοσης και η αντίστοιχη καθυστέρηση, ενώ οι ουρές είναι τύπου DropTail με χωρητικότητα 100 πακέτα. Η S/P - GW συνδέεται με ζεύξεις σημείου προς σημείου (Point to Point - P2P) με τους δύο απομακρυσμένους εξυπηρετητές TCP και QUIC, στους οποίους δρομολογείται η κίνηση από τα UEs που χρησιμοποιούν το αντίστοιχο πρωτόκολλο.

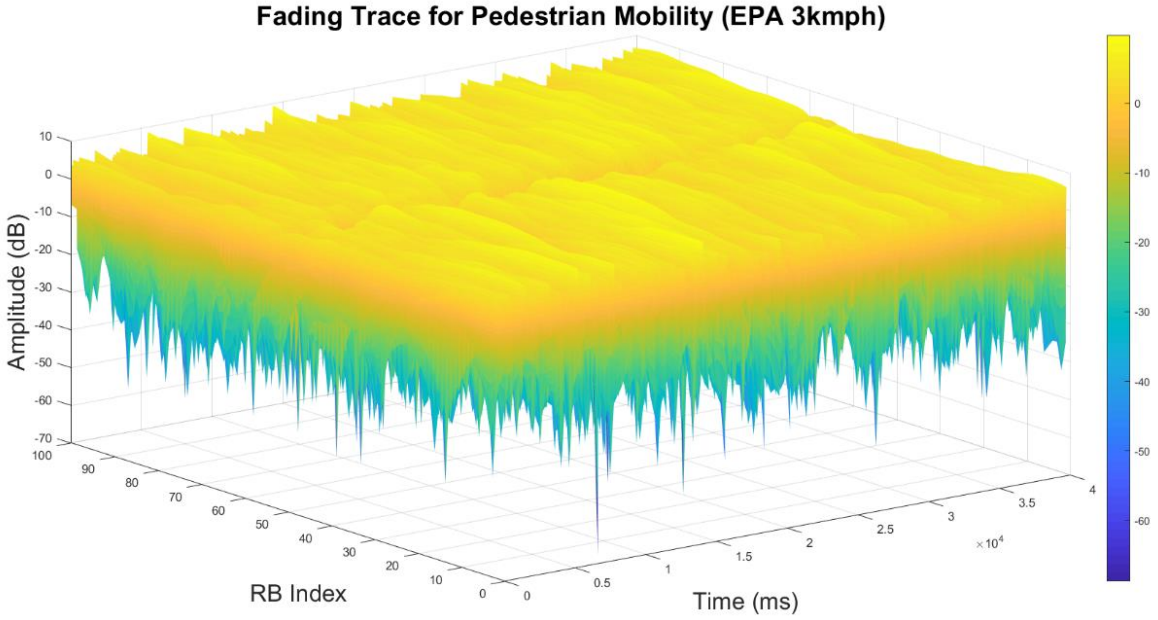


Σχήμα 6.1 *Μοντέλο προσομοίωσης για τη μελέτη των TCP και QUIC σε δίκτυο LTE*

Για τη μοντελοποίηση του ασύρματου δικτύου πρόσβασης (RAN) του LTE, ορίζονται αρχικές τιμές για ορισμένες βασικές παραμέτρους. Ως συχνότητα λειτουργίας επιλέχθηκε η LTE Band 4 (βλ. Πίνακα 6.1), ενώ το εύρος ζώνης ορίστηκε στη μέγιστη δυνατή τιμή των 20MHz. Για τις απώλειες διάδοσης χρησιμοποιήθηκε λογαριθμικό μοντέλο. Οι διαλείψεις στον ασύρματο δίαυλο ακολουθούν κατανομή Rayleigh, για ταχύτητα χρηστών 3km/h και προφίλ καθυστέρησης ισχύος EPA (βλ. Σχήμα 6.2). Το μέγεθος του ενταμιευτή μετάδοσης στον eNB ορίστηκε στα 512kB. Τέλος, οι παράμετροι ισχύος εκπομπής και συντελεστή θορύβου αντιστοιχούν στις συνθήκες ενός macro cell site. Οι αρχικές τιμές αυτών των παραμέτρων της προσομοίωσης δίνονται συγκεντρωτικά στον Πίνακα 6.1.

Μοντέλο Διάδοσης	Απώλειες Διάδοσης	$L = 30.8 + 24.2 \log_{10}(R)$ dB, R σε m
	Διαλείψεις	Rayleigh με ταχύτητα 3km/h και προφίλ καθυστέρησης ισχύος EPA
Παράμετροι LTE	Earfcn DL	2300
	Συχνότητα DL	2145 MHz
	Earfcn UL	20300
	Συχνότητα UL	1745 MHz
	Resource Blocks DL	100
	Resource Blocks UL	100
Παράμετροι eNodeB	Τύπος Κεραίας	Ομοιοκατευθυντική
	Ισχύς Εκπομπής	46 dBm
	Συντελεστής Θορύβου	5 dB
	MAC Scheduler	Round Robin
	HARQ	Enabled
	RLC Mode	RLC AM
	RLC transmission buffer	512 KB
Παράμετροι UE	Τύπος Κεραίας	Ομοιοκατευθυντική
	Ισχύς Εκπομπής	23 dBm
	Συντελεστής Θορύβου	9 dB

Πίνακας 6.1 *Παράμετροι PHY και MAC της προσομοίωσης*



Σχήμα 6.2 *Απώλεια σε dB που θα υποστεί, λόγω διαλείψεων, η μετάδοση των physical frames του LTE κατά τη διάρκεια της προσομοίωσης*

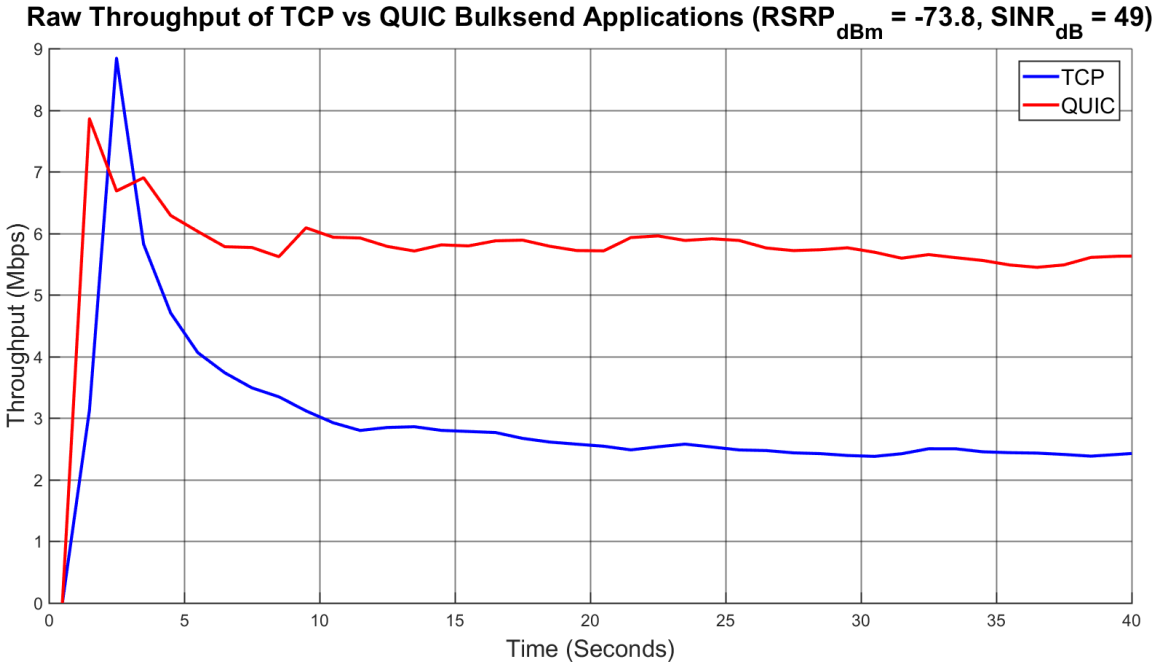
Κύριος στόχος της εργασίας είναι η σύγκριση της επίδοσης των πρωτοκόλλων TCP και QUIC, όταν παρεμβάλλεται η διάδοση πάνω από ασύρματο κανάλι, καθώς και η διερεύνηση των βελτιώσεων που προσφέρουν τα καινούρια χαρακτηριστικά του QUIC. Η υλοποίηση του QUIC στον προσομοιωτή ns3 πραγματοποιήθηκε στο [69].

6.2 Ρυθμαπόδοση Σταθερής Κατάστασης

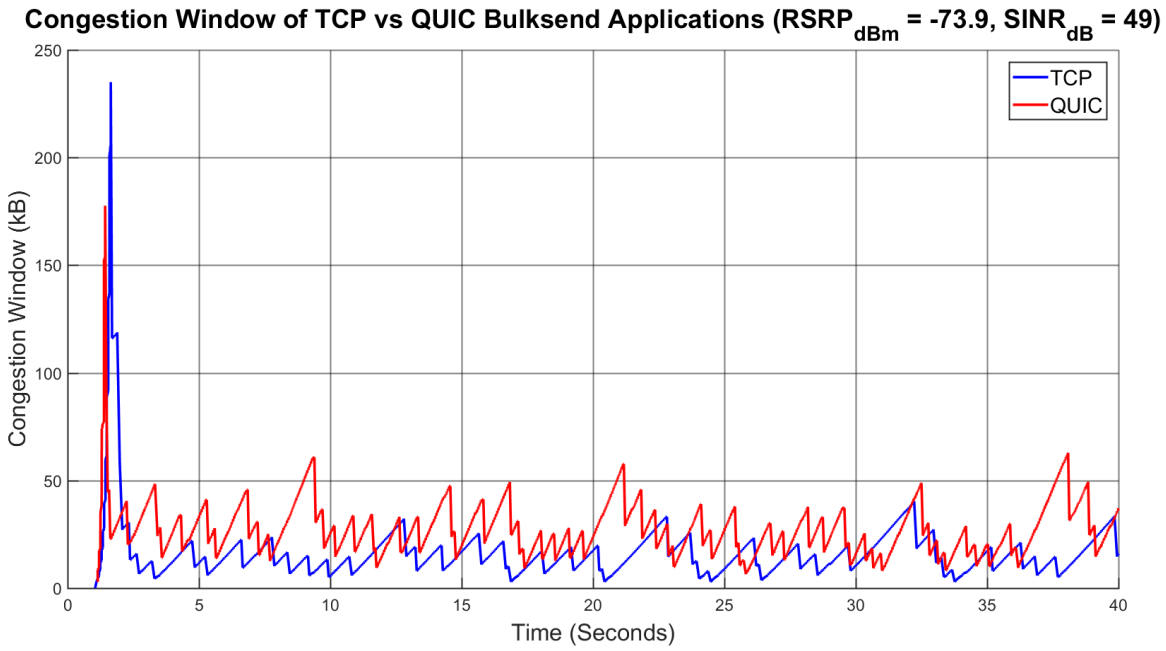
Στο πρώτο σενάριο εξετάζεται η ρυθμαπόδοση (throughput) που επιτυγχάνουν τα δύο πρωτόκολλα σε σταθερή κατάσταση (steady state). Δύο ξεχωριστά UE τοποθετούνται στην ίδια απόσταση από το σταθμό βάσης, εκ των οποίων το πρώτο επικοινωνεί με τον TCP Server, ενώ το δεύτερο με τον QUIC server. Προκειμένου να μελετηθεί η απόδοση των πρωτοκόλλων σε συνεχή ροή δεδομένων στην κατεύθυνση DL, χρησιμοποιείται η εφαρμογή BulkSend του προσομοιωτή ns3, με την οποία κάθε server στέλνει δεδομένα μήκους 512 bytes προς το αντίστοιχο UE. Η εφαρμογή αυτή μεταβιβάζει δεδομένα προς τον ενταμιευτή μετάδοσης του server, κάθε φορά που υπάρχει διαθέσιμος αποθηκευτικός χώρος, φροντίζοντας έτσι να υπάρχουν πάντα δεδομένα έτοιμα για μετάδοση στο δίκτυο.

Για τους αλγορίθμους συμφόρησης που χρησιμοποιούν τα δύο πρωτόκολλα, επιλέχθηκε ο αλγόριθμος TCP New Reno για το TCP και ο αλγόριθμος QUIC Congestion Options για το QUIC. Η επιλογή αυτή έγινε προκειμένου τα δύο πρωτόκολλα να αντιδρούν με παρόμοιο τρόπο στις απώλειες πακέτων, καθώς ο σχεδιασμός του αλγορίθμου QUIC έχει βασιστεί στον αλγόριθμο New Reno, οπότε η σύγκριση θα είναι αξιόπιστη. Επιπλέον, προκειμένου οι συνολικές απώλειες πακέτων να οφείλονται κυρίως στον ασύρματο δίαυλο, στις P2P ζεύξεις μεταξύ των server και της S/P - GW ορίστηκε χαμηλός ρυθμός απώλειας πακέτων (0.5%). Η διάρκεια της προσομοίωσης ορίστηκε στα 40 δευτερόλεπτα. Τέλος, η προσομοίωση επαναλαμβάνεται για διαφορετικές αποστάσεις των UE από τον eNB, προκειμένου να μελετηθεί η επίδραση της ποιότητας του σήματος λήψης. Τα αποτελέσματα παρουσιάζονται στη συνέχεια.

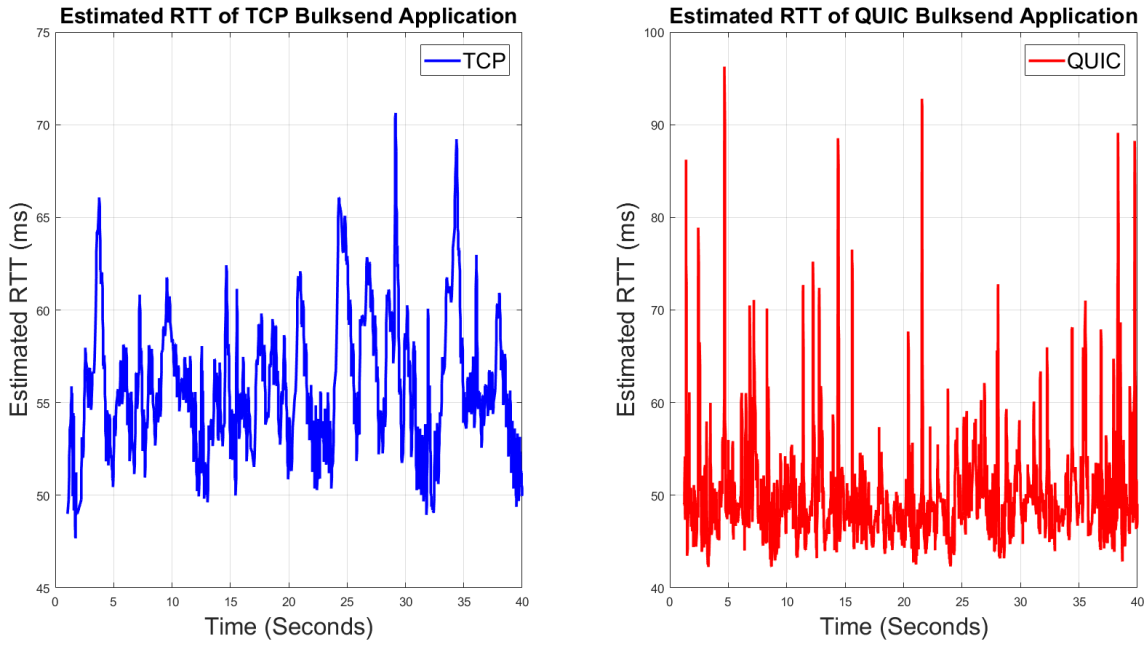
6.2.1 UE Distance = 250m



Σχήμα 6.3 *Ρυθμαπόδοση των πρωτοκόλλων TCP και QUIC (UE distance = 250m)*



Σχήμα 6.4 *Μεταβολή του παραθύρου συμφόρησης (cwnd) για τα πρωτόκολλα TCP και QUIC (UE distance =250m)*

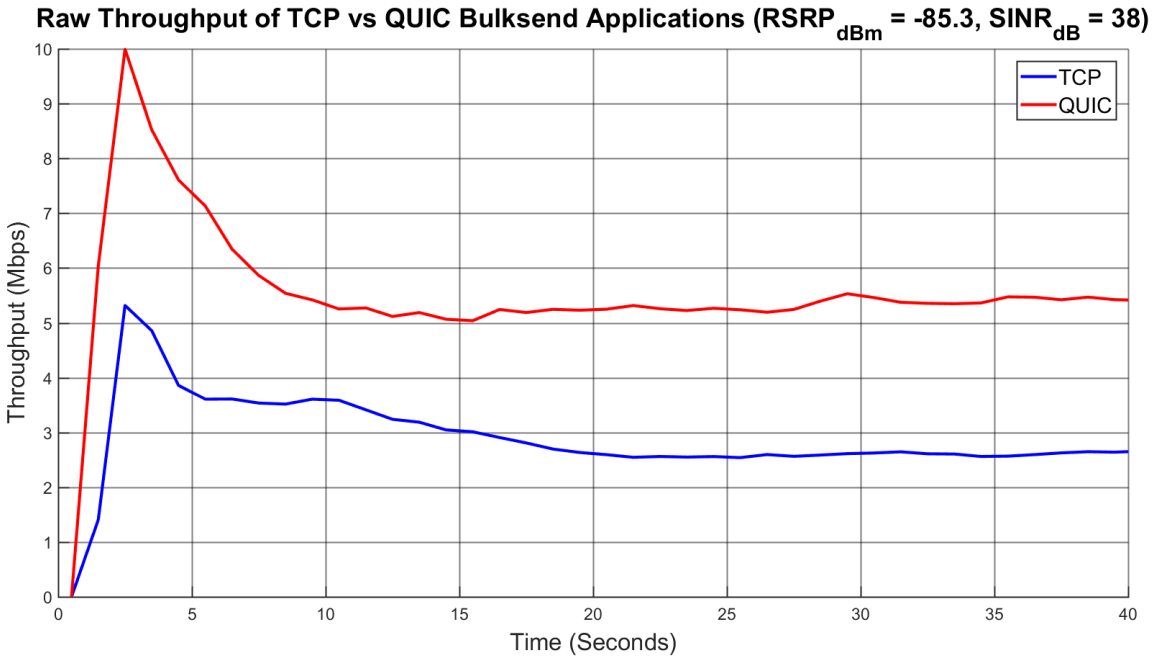


Σχήμα 6.5 *Εκτίμηση χρόνου RTT για τα πρωτόκολλα TCP και QUIC (UE distance = 250m)*

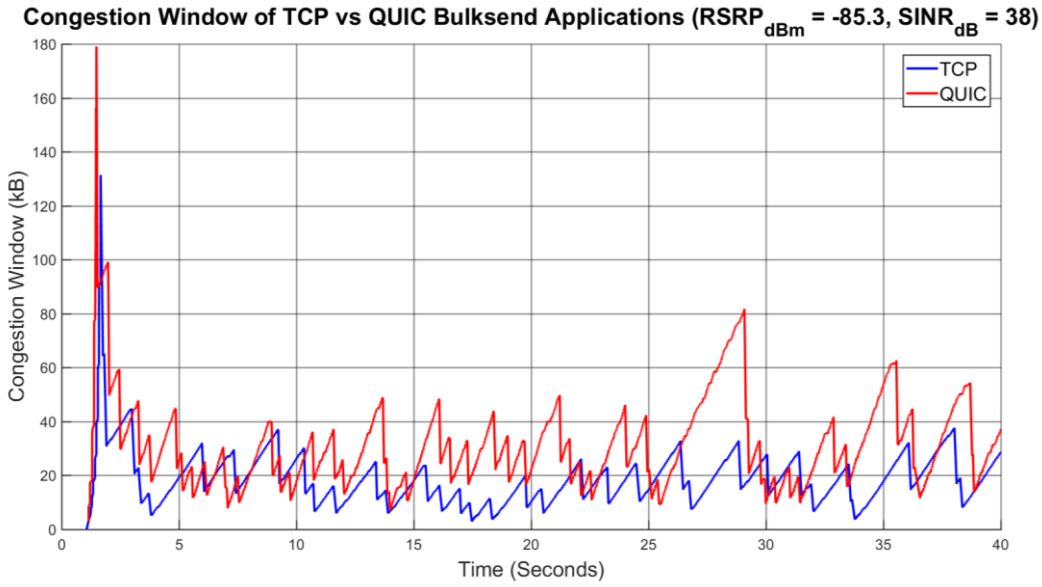
Όπως φαίνεται στο Σχήμα 6.3, το QUIC επιτυγχάνει υψηλότερο throughput στη σταθερή κατάσταση. Ο λόγος που συμβαίνει αυτό μπορεί να εξηγηθεί από το Σχήμα 6.4, που δείχνει την μεταβολή του παραθύρου συμφόρησης για τις δύο ροές. Ο αλγόριθμος συμφόρησης του QUIC, ο οποίος βασίζεται στον αλγόριθμο NewReno (βλ. §3.6), επιτυγχάνει να διατηρεί μεγαλύτερο παράθυρο συμφόρησης καθ' όλη τη διάρκεια της σύνδεσης. Επιπλέον, παρατηρείται ότι το QUIC αντιδρά καλύτερα στις απώλειες

πακέτων, καθώς αυξάνει το παράθυρο συμφόρησης με ταχύτερο ρυθμό ύστερα από ένα τέτοιο συμβάν. Τέλος, το QUIC διατηρεί χαμηλότερο, κατά μέση τιμή, εκτιμώμενο RTT (βλ. Σχήμα 6.5), καθώς από κάθε δείγμα RTT αφαιρείται ο χρόνος επεξεργασίας πακέτου (ACK Delay) σε ένα τερματικό, σύμφωνα με τον μηχανισμό που περιγράφεται στην §3.5.2. Ωστόσο, όπως θα φανεί στη συνέχεια, το γεγονός αυτό μπορεί να οδηγήσει σε μειωμένη απόδοση, όταν οι συνθήκες λήψης στον ασύρματο δίαυλο χειροτερεύουν.

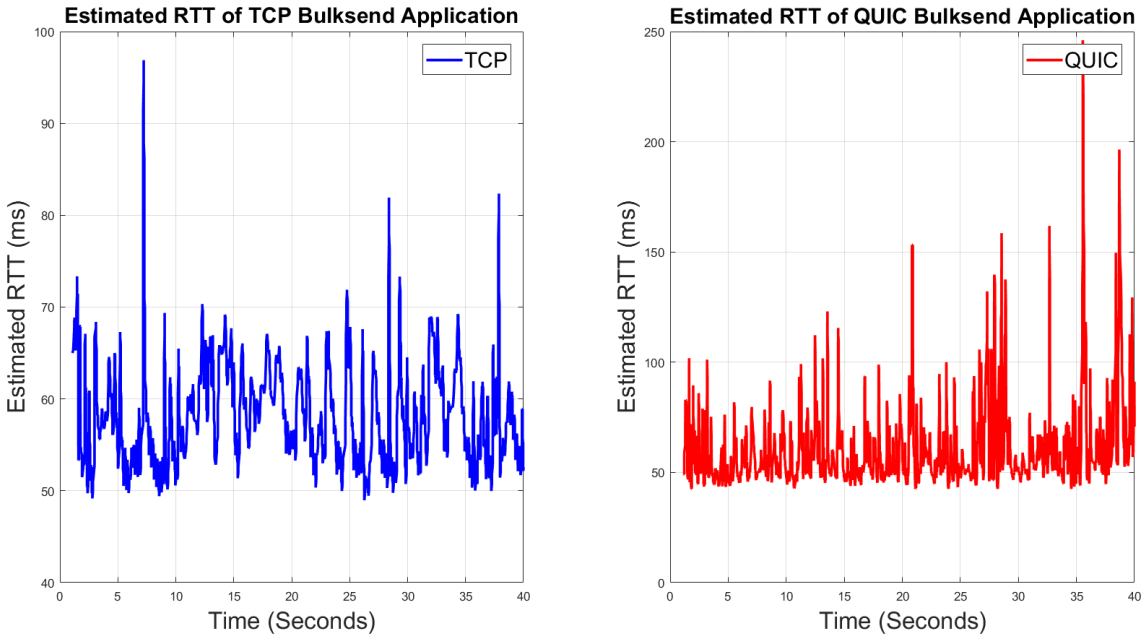
6.2.2 UE Distance = 750m



Σχήμα 6.6 *Ρυθμαπόδοση των πρωτοκόλλων TCP και QUIC (UE distance = 750m)*



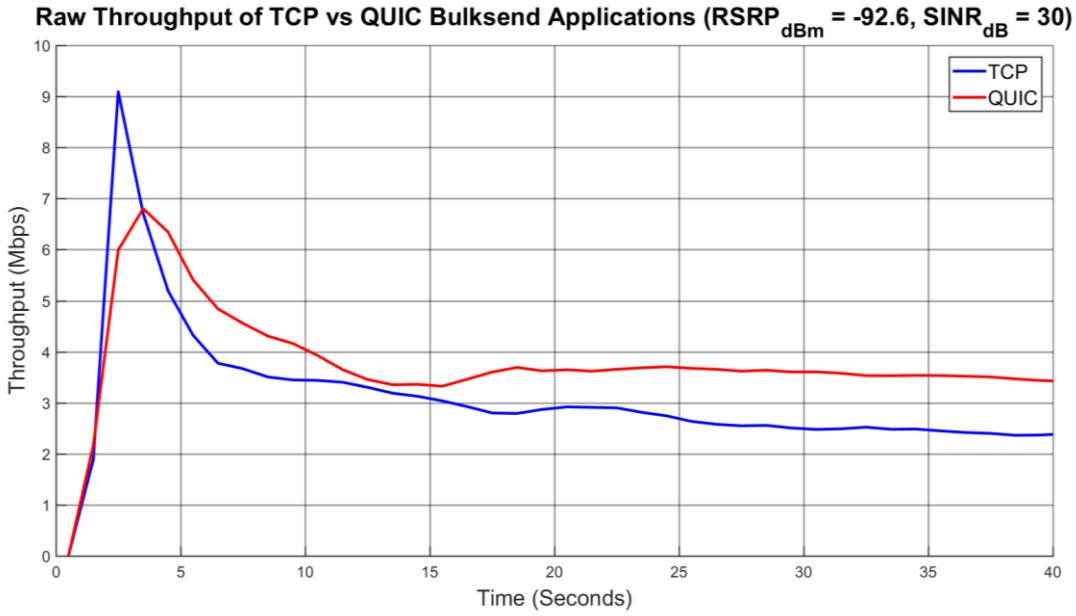
Σχήμα 6.7 *Μεταβολή του παραθύρου συμφόρησης (cwnd) για τα πρωτόκολλα TCP και QUIC (UE distance = 750m)*



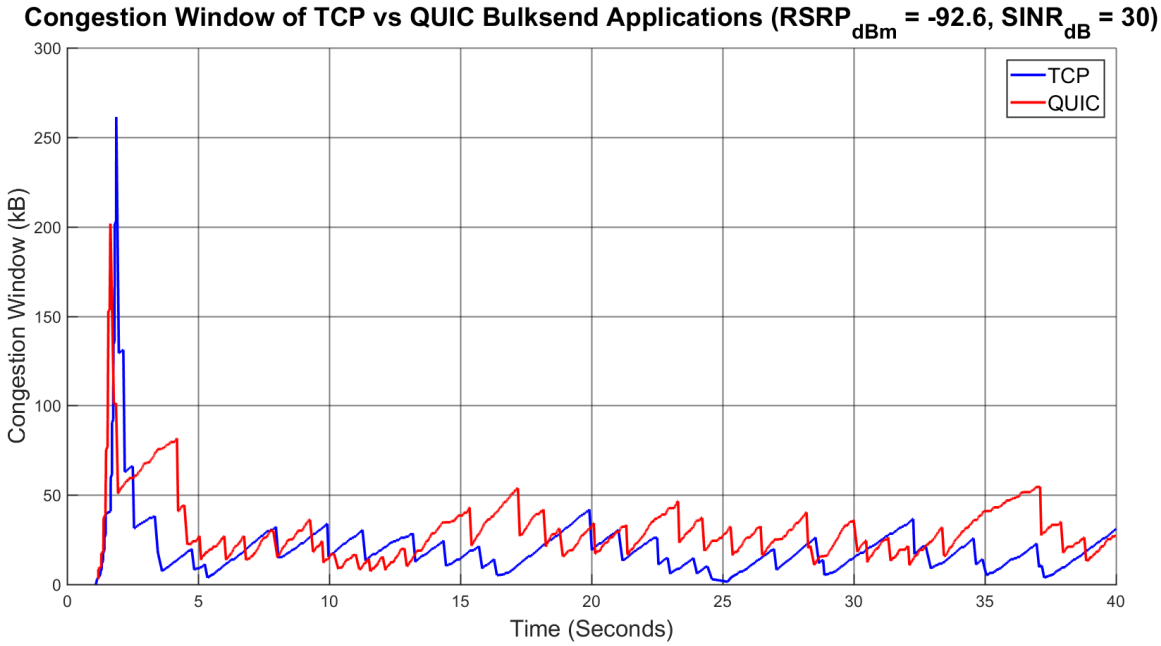
Σχήμα 6.8 *Εκτίμηση χρόνου RTT για τα πρωτόκολλα TCP και QUIC (UE distance = 750m)*

Αυξάνοντας την απόσταση των UE από τον eNB στα 750m, παρατηρείται ότι το QUIC εξακολουθεί να διατηρεί υψηλότερο throughput σταθερής κατάστασης, όπως απεικονίζεται στο Σχήμα 6.6. Επιπλέον, το QUIC συνεχίζει να διατηρεί μεγαλύτερο παράθυρο συμφόρησης συγκριτικά με το TCP (βλ. Σχήμα 6.7). Τέλος, οι εκτιμήσεις του χρόνου RTT και για τα δύο πρωτόκολλα δεν παρουσιάζουν σημαντική μεταβολή, παρά την αύξηση της απόστασης των UE από τον eNB στα 750m (βλ. Σχήμα 6.8).

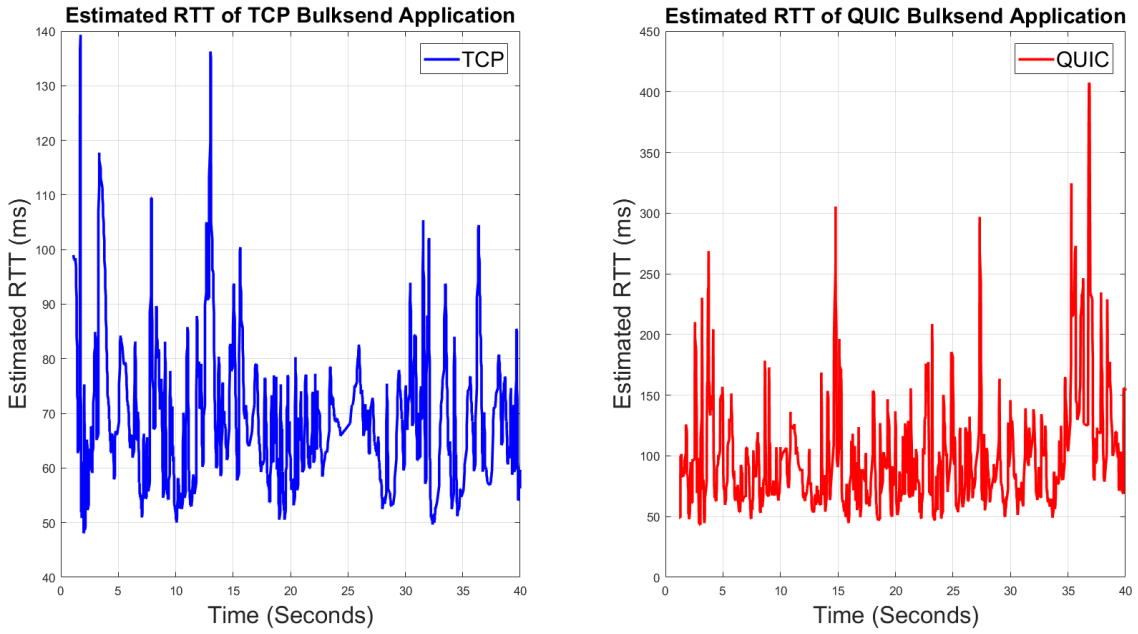
6.2.3 UE Distance = 1500m



Σχήμα 6.9 *Ρυθμαπόδοση των πρωτοκόλλων TCP και QUIC (UE distance = 1500m)*



Σχήμα 6.10 *Μεταβολή του παραθύρου συμφόρησης (cwnd) για τα πρωτόκολλα TCP και QUIC (UE distance = 1500m)*

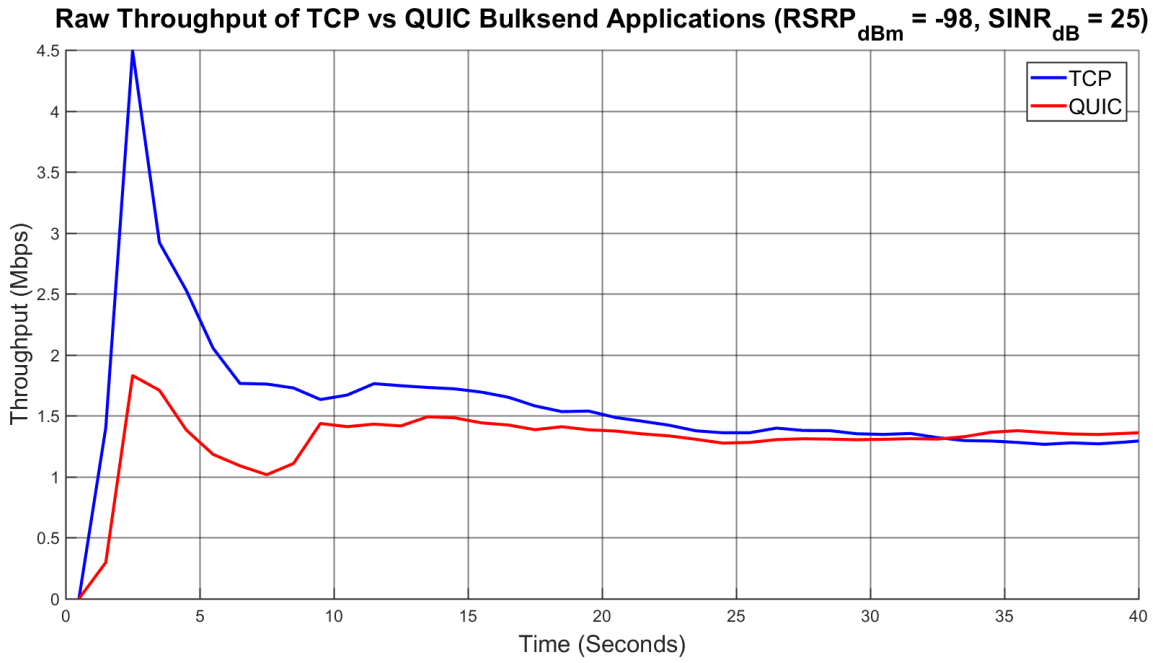


Σχήμα 6.11 *Εκτίμηση χρόνου RTT για τα πρωτόκολλα TCP και QUIC (UE distance = 1500m)*

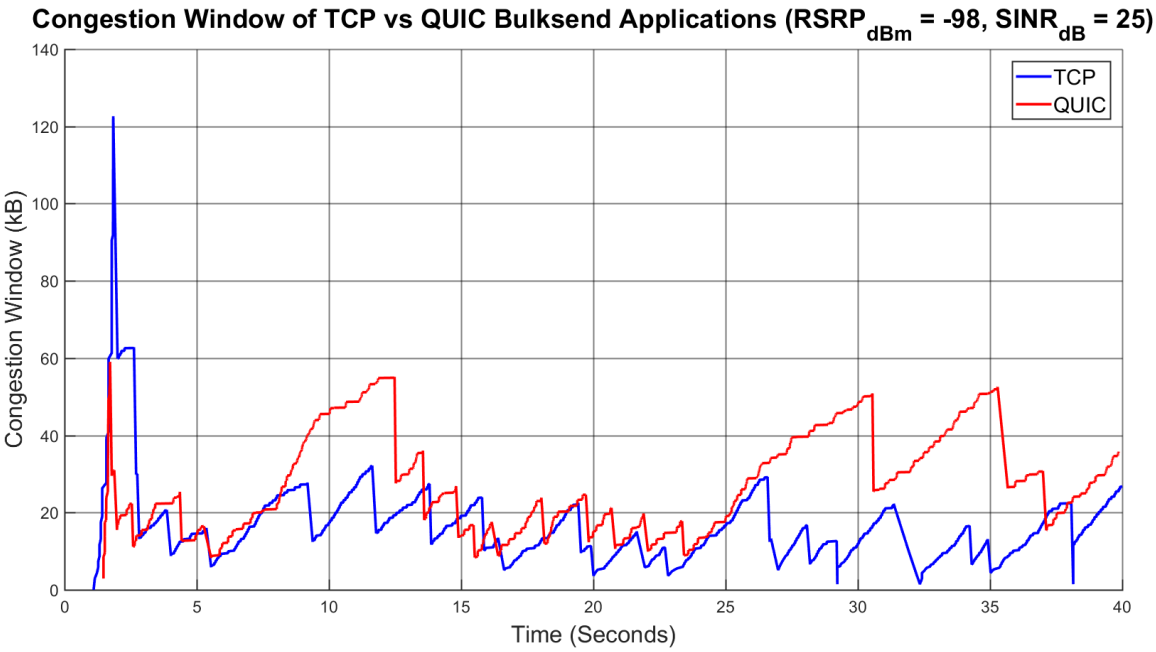
Η περαιτέρω χειροτέρευση του σήματος λήψης στα UE από το σταθμό βάσης οδηγεί τα δύο πρωτόκολλα σε παρόμοια συμπεριφορά των αλγορίθμων συμφόρησης, όπως διαπιστώνεται από το γράφημα του παραθύρου συμφόρησης (βλ. Σχήμα 6.10). Το QUIC εξακολουθεί να διατηρεί υψηλότερο throughput στη σταθερή κατάσταση, αν και η απόδοση του QUIC πλησιάζει αυτή του TCP. Τέλος, ο εκτιμώμενος χρόνος RTT για το QUIC εμφανίζει υψηλότερη διακύμανση, καθώς καταγράφει καλύτερα την αυξημένη

καθυστέρηση που υφίστανται τα πακέτα στον ασύρματο δίαυλο λόγω των αναμεταδόσεων HARQ στο φυσικό στρώμα του LTE.

6.2.4 UE Distance = 2500m

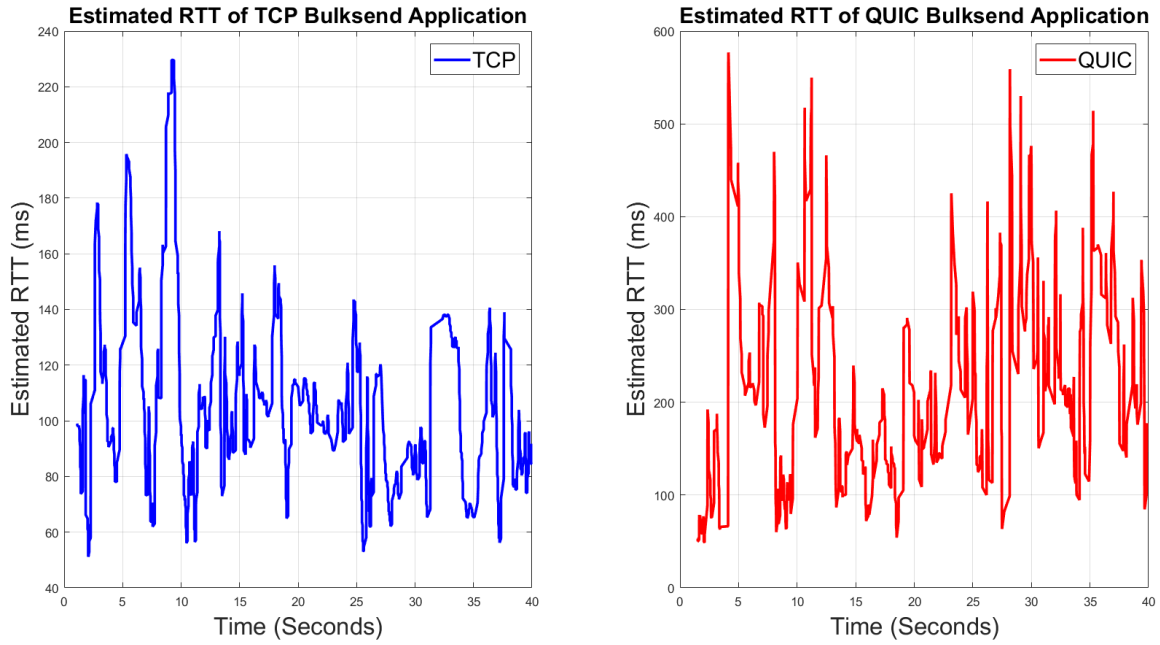


Σχήμα 6.12 *Ρυθμαπόδοση των πρωτοκόλλων TCP και QUIC (UE distance = 2500m)*



Σχήμα 6.13 *Μεταβολή του παραθύρου συμφόρησης (cwnd) για τα πρωτόκολλα TCP και QUIC (UE distance = 2500m)*

Για πολύ χαμηλή ποιότητα του σήματος λήψης, το QUIC εμφανίζει χειρότερη επίδοση στη ρυθμαπόδοση σταθερής κατάστασης (βλ. Σχήμα 6.12). Καίτοι κατά διαστήματα διατηρεί υψηλότερο παράθυρο συμφόρησης, η ροή δεδομένων του QUIC παρουσιάζει παρόμοια χαρακτηριστικά με bursty μεταδόσεις (γρήγορη αύξηση και απότομη μείωση του παραθύρου συμφόρησης, βλ. Σχήμα 6.13), οι οποίες μεταδόσεις συνήθως έχουν χαμηλή αποδοτικότητα (efficiency). Επιπλέον, έχει βρεθεί ότι σε περιπτώσεις αναδιάταξης πακέτων (packet re-ordering) η απόδοση του QUIC μειώνεται, καθώς ενδέχεται να πραγματοποιούνται άσκοπες αναμεταδόσεις πακέτων τα οποία λανθασμένα κρίθηκαν ως χαμένα, ενώ στην πραγματικότητα έχει καθυστερήσει η λήψη της επιβεβαίωσης ACK εξαιτίας των συνθηκών του ασύρματου διαύλου [70]. Η υψηλή διακύμανση του εκτιμώμενου RTT (βλ. Σχήμα 6.14) για το πρωτόκολλο QUIC υποδηλώνει την ύπαρξη της αναδιάταξης πακέτων, φαινόμενο που στα ασύρματα δίκτυα οφείλεται στην επίδραση που έχουν οι ταχείες διαλείψεις στο σήμα λήψης.



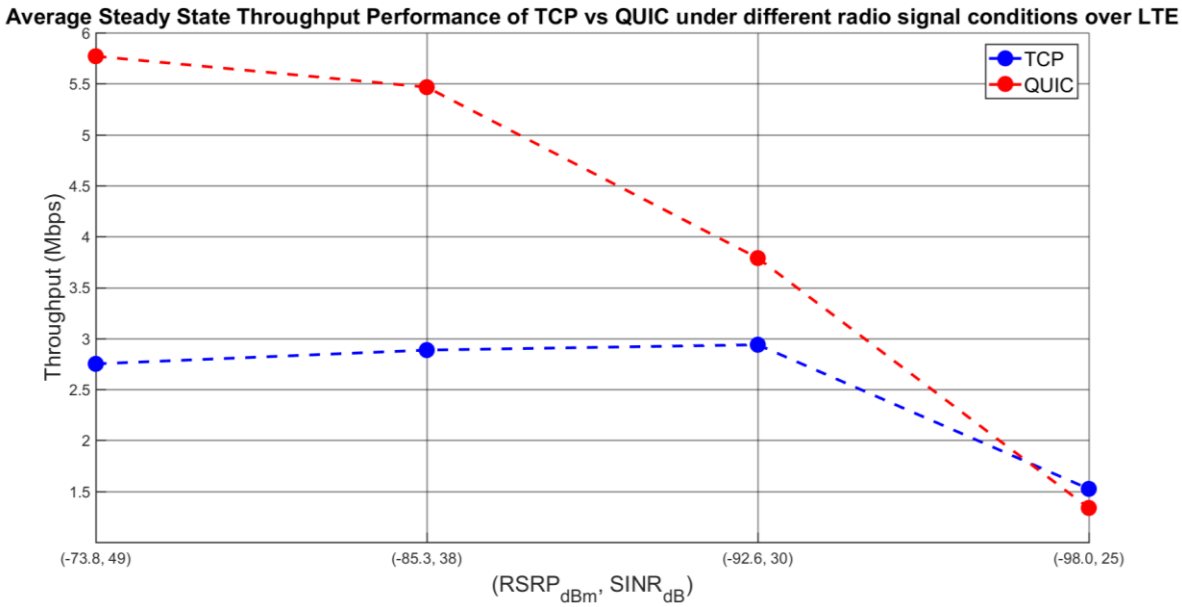
Σχήμα 6.14 *Εκτίμηση χρόνου RTT για τα πρωτόκολλα TCP και QUIC (UE distance = 2500m)*

6.2.5 Συγκριτικά Αποτελέσματα

Στον Πίνακα 6.2 παρουσιάζονται συγκεντρωτικά τα αποτελέσματα για το μέσο throughput στη σταθερή κατάσταση, συναρτήσει της απόστασης του UE από τον eNB. Ως σταθερή κατάσταση θεωρείται το χρονικό διάστημα από 5 έως 40 δευτερόλεπτα, προκειμένου να μην λαμβάνεται υπόψη στις μετρήσεις το στάδιο της αργής εκκίνησης.

Απόσταση από eNB (m)	Μέσο TCP Throughput (Mbps)	Μέσο QUIC Throughput (Mbps)
250	2.7504	5.7696
750	2.8855	5.4658
1500	2.9941	3.8316
2500	1.5284	1.3247

Πίνακας 6.2 *Μέση ρυθμαπόδοση σταθερής κατάστασης για τα πρωτόκολλα TCP και QUIC*



Σχήμα 6.15 *Μέση ρυθμαπόδοση σταθερής κατάστασης για τα πρωτόκολλα TCP και QUIC συναρτήσει της ποιότητας του σήματος λήψης*

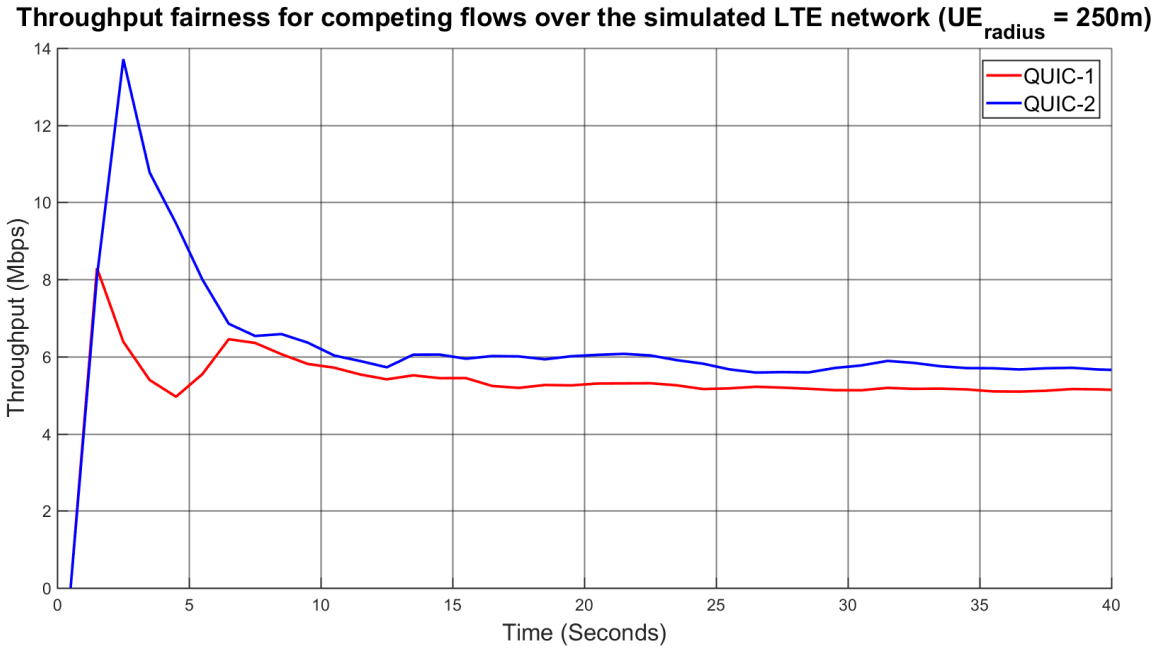
Συνοπτικά, παρατηρείται ότι η βελτίωση της απόδοσης που προσφέρει το QUIC σε ενσύρματα δίκτυα [28] επιτυγχάνεται και στα δίκτυα LTE παρά τις ιδιαιτερότητες του ασύρματου διαύλου, που μπορεί να επηρεάσουν τη λειτουργία των πρωτοκόλλων μεταφοράς. Ωστόσο, η μείωση της ποιότητας του σήματος λήψης από τον eNB τείνει να επηρεάσει περισσότερο το QUIC. Επομένως, υπό δυσμενής συνθήκες, τα δύο πρωτόκολλα επιτυγχάνουν παρόμοια throughput. Αξίζει, βέβαια, να σημειωθεί ότι η ζώνη συχνοτήτων που ορίστηκε στην προσομοίωση (LTE Band 4 - 2145MHz Downlink) χρησιμοποιείται κυρίως για την κάλυψη κινητών χρηστών που βρίσκονται σε κοντινές αποστάσεις από τον eNB. Χρήστες σε μεγαλύτερες αποστάσεις συνδέονται με το σταθμό βάσης σε χαμηλότερες ζώνες συχνοτήτων (600MHz ή 700MHz), προκειμένου να διαθέτουν καλύτερη κάλυψη. Επομένως, όπως υποδηλώνουν τα αποτελέσματα της προσομοίωσης, παρατηρείται ότι πράγματι το QUIC έχει καλύτερη επίδοση από το TCP σε κοντινή ή μεσαία απόσταση από τον eNB.

6.3 Δικαιοσύνη Ροών TCP και QUIC

Μία σημαντική ιδιότητα που πρέπει να εξασφαλίζεται από τα πρωτόκολλα μεταφοράς, είναι η δίκαιη κατανομή του περιορισμένου εύρους ζώνης σε ανταγωνιστικές ροές. Χωρίς αυτή τη δίκαιη κατανομή, μία ροή δεδομένων που χρησιμοποιεί ένα «άδικο» πρωτόκολλο θα έχει καλύτερη επίδοση συγκριτικά με άλλες ανταγωνιστικές ροές. Για παράδειγμα, έχει βρεθεί ότι σε περιβάλλον ενσύρματου δικτύου, η δικαιοσύνη δεν τηρείται ανάμεσα στο TCP και το QUIC, ακόμα και αν χρησιμοποιούν ακριβώς τον ίδιο αλγόριθμο συμφόρησης [70]. Προκειμένου να εξεταστεί η ιδιότητα αυτή και στο περιβάλλον του LTE, το οποίο διαθέτει ιδιαίτερα περιορισμένους πόρους στο ασύρματο τμήμα του δικτύου, χρησιμοποιήθηκαν οι εφαρμογές BulkSend της §6.2 με την ίδια παραμετροποίηση και μεταβλητό αριθμό ροών. Η απόσταση των UE από τον eNB ορίζεται στα 250m. Τα αποτελέσματα για τη ρυθμαπόδοση σταθερής κατάστασης παρουσιάζονται στη συνέχεια.

6.3.1 2 QUIC Flows

Στην περίπτωση δύο ανταγωνιστικών QUIC συνδέσεων, παρατηρείται ότι είναι δίκαιες μεταξύ τους και έχουν παρόμοια επίδοση (βλ. Σχήμα 6.16), επομένως οι πόροι κατανέμονται ισότιμα.



Σχήμα 6.16 *Ρυθμαπόδοση δύο ανταγωνιστικών ροών QUIC*

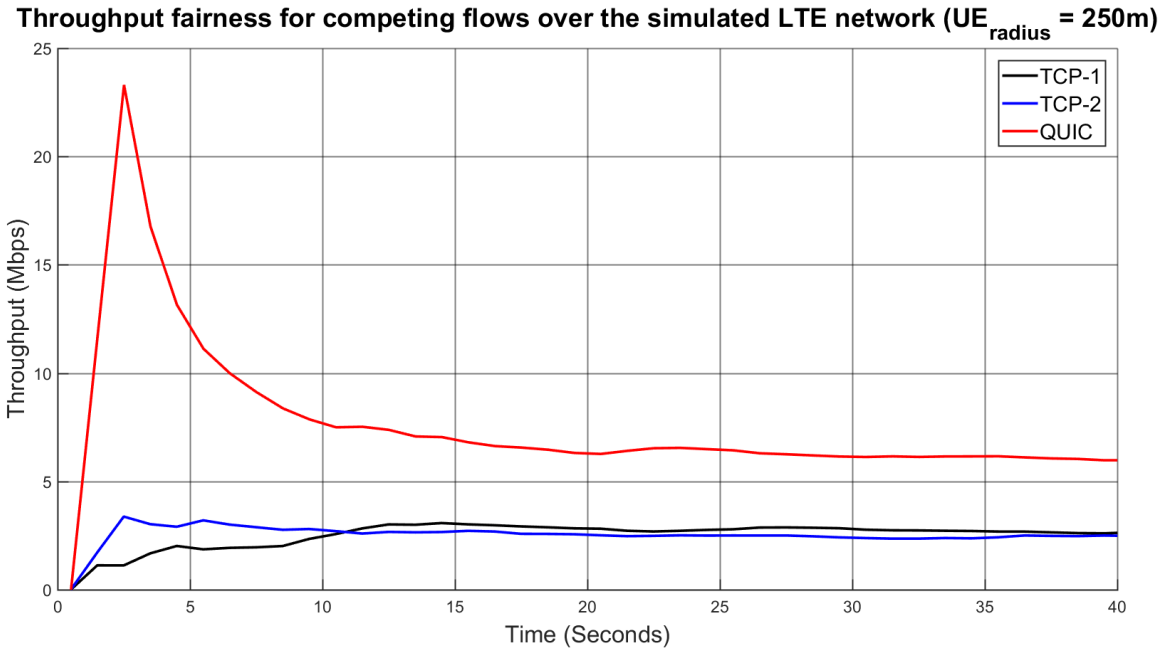
6.3.2 1 QUIC vs Multiple TCP Flows

Έχοντας εξετάσει την απλή περίπτωση για τη σύγκριση του QUIC με το TCP στην §6.2, που για απόσταση 250m το QUIC είχε διπλάσια επίδοση throughput, εξετάζεται και η συνύπαρξη του QUIC με περισσότερες ροές TCP (Σχήμα 6.17 & Σχήμα 6.18).

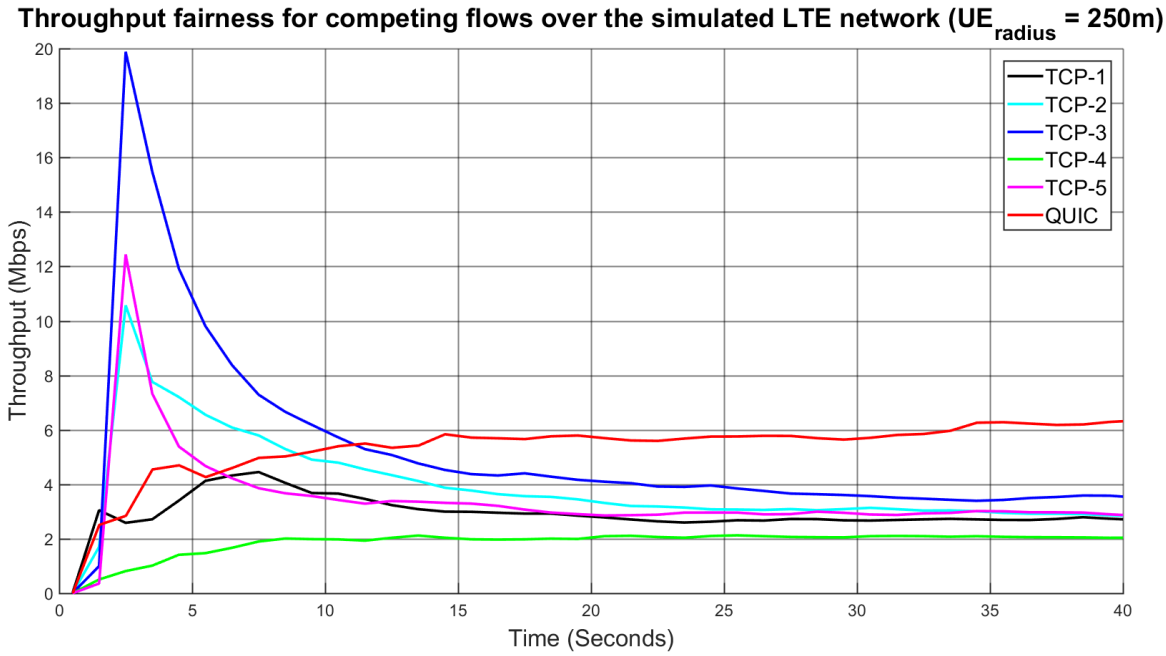
Παρατηρείται ότι οι ροές TCP είναι δίκαιες μεταξύ τους, καθώς στη σταθερή κατάσταση συγκλίνουν περίπου στην ίδια τιμή ρυθμαπόδοσης. Η ροή QUIC εξακολουθεί να διατηρεί υψηλότερο throughput, καίτοι τα δύο πρωτόκολλα χρησιμοποιούν αλγορίθμους συμφόρησης με παρόμοια χαρακτηριστικά (TCP NewReno - QUIC Congestion Control).

6.3.3 2 QUIC vs 2 TCP Flows

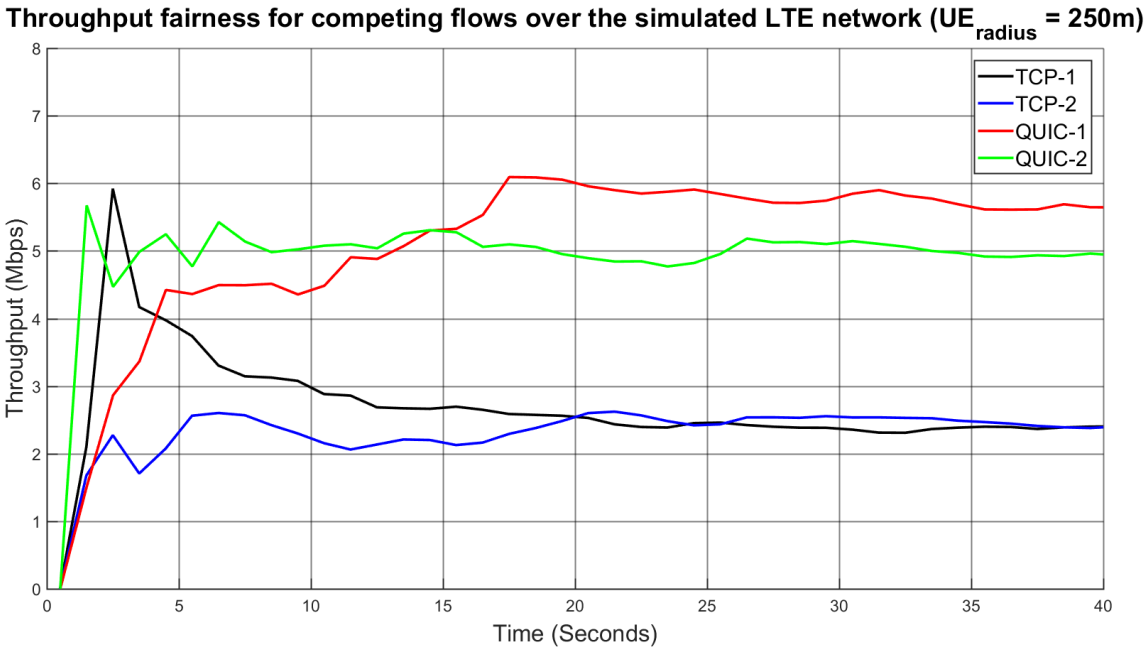
Τέλος, εξετάζεται η περίπτωση πολλαπλών ροών και για τα δύο πρωτόκολλα (βλ. Σχήμα 6.19). Σε συμφωνία με τα προηγούμενα, οι ροές QUIC παραμένουν δίκαιες μεταξύ τους και δεσμεύουν ουσιαστικά σχεδόν το διπλάσιο bandwidth από τις ροές TCP. Το φαινόμενο αυτό μπορεί να εξηγηθεί και από τα διαγράμματα των παραθύρων συμφόρησης της §6.2, όπου το QUIC διατηρούσε υψηλότερο παράθυρο συμφόρησης από το TCP και άρα δέσμευε μεγαλύτερο μέρος του εύρους ζώνης. Συνεπώς, οι ροές QUIC εμφανίζουν έλλειψη δικαιοσύνης ως προς τις ροές TCP και στο περιβάλλον του LTE, καίτοι η μετάδοση στο ραδιοδίαυλο από τον MAC Scheduler είναι μορφής Round Robin, που προσθέτει «δικαιοσύνη» στη χρήση των πόρων του σταθμού βάσης.



Σχήμα 6.17 *Ρυθμαπόδοση ανταγωνιστικών ροών: 1 QUIC vs 2 TCP*



Σχήμα 6.18 *Ρυθμαπόδοση ανταγωνιστικών ροών: 1 QUIC vs 5 TCP*



Σχήμα 6.19 *Ρυθμαπόδοση ανταγωνιστικών ροών: 2 QUIC vs 2 TCP*

Πλήθος Ροών	Πρωτόκολλο	Μέσο Throughput (Mbps)	Τυπική Απόκλιση
2 QUIC	QUIC	5.3852	0.3679
1 QUIC vs 2 TCP	QUIC	5.9790	-
	TCP	2.5713	0.1178
1 QUIC vs 5 TCP	QUIC	6.3422	-
	TCP	2.7768	0.5245
2 QUIC vs 2 TCP	QUIC	5.2883	0.5016
	TCP	2.4048	0.0011

Πίνακας 6.3 *Ρυθμαπόδοση ανταγωνιστικών ροών TCP και QUIC*

6.4 Επίδοση σε File Download

Έχοντας εξετάσει την επίδοση των δύο πρωτοκόλλων σε συνεχείς ροές δεδομένων μεγάλης διάρκειας, θα μελετηθεί η επίδοσή τους κατά τη λήψη αρχείων συγκεκριμένου μεγέθους και η πιθανή βελτίωση που θα επιφέρει το QUIC. Τα UE τοποθετήθηκαν σε απόσταση 250m από τον eNB, προκειμένου να έχουν ισχυρό σήμα λήψης. Τα μεγέθη αρχείων που επιλέχθηκαν είναι αντιπροσωπευτικά σημερινών web pages στο Διαδίκτυο, ώστε να προσομοιωθεί ουσιαστικά ο χρόνος λήψης μίας ιστοσελίδας. Ως μέτρα επίδοσης θα χρησιμοποιηθούν ο χρόνος λήψης και το goodput, το οποίο ορίζεται ως εξής:

$$Goodput = \frac{Μέγεθος\ Αρχείου}{Χρόνος\ Λήψης}\ (σε\ Mbps) \qquad (6.1)$$

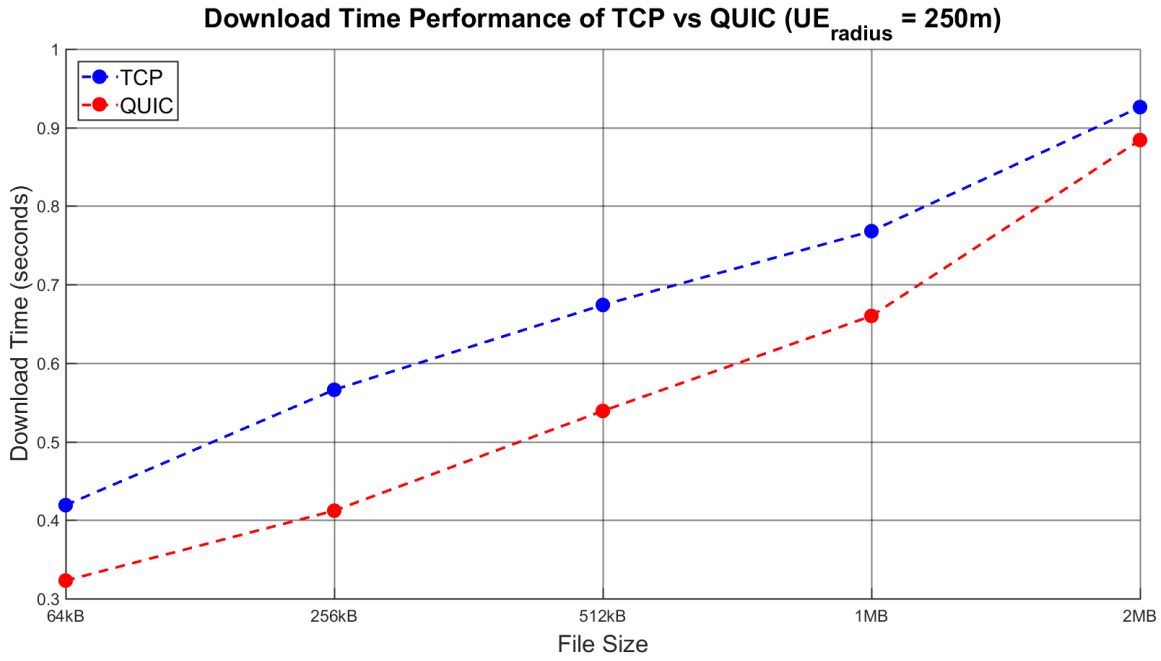
Τα σχετικά αποτελέσματα παρουσιάζονται στον Πίνακα 6.4 και Πίνακα 6.5 αντίστοιχα.

Μέγεθος Αρχείου	Χρόνος Λήψης TCP (s)	Χρόνος Λήψης QUIC (s)	Μείωση (%)
64 kB	0.419	0.323	-9.6
256 kB	0.566	0.412	-15.4
512 kB	0.674	0.539	-13.5
1 MB	0.768	0.66	-10.8
2 MB	0.926	0.884	-4.2

Πίνακας 6.4 *Χρόνοι λήψης αρχείων για τα πρωτόκολλα TCP και QUIC*

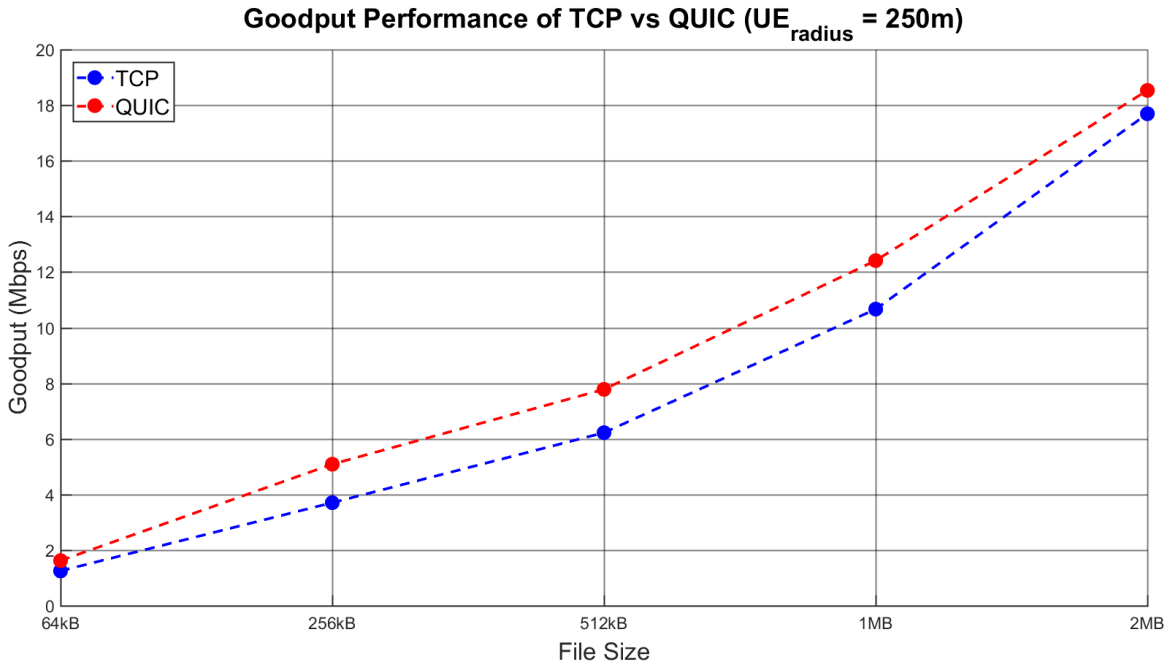
Μέγεθος Αρχείου	Goodput TCP (Mbps)	Goodput QUIC (Mbps)	Βελτίωση (%)
64 kB	1.25128	1.62318	29.7
256 kB	3.70522	5.09017	37.4
512 kB	6.223	7.78164	25.1
1 MB	10.667	12.4121	16.4
2 MB	17.6933	18.5339	4.8

Πίνακας 6.5 *Goodput λήψης αρχείων για τα πρωτόκολλα TCP και QUIC*



Σχήμα 6.20 *Χρόνοι λήψης αρχείων για τα πρωτόκολλα TCP και QUIC συναρτήσει του μεγέθους αρχείου*

Αν και από πρώτη θεώρηση η μείωση στους χρόνους λήψης αρχείων δεν παρουσιάζουν αισθητές διαφορές, η ποσοστιαία μείωση φανερώνει τη βελτίωση που προσφέρει το QUIC. Όπως είναι φυσικό για ένα περιβάλλον προσομοίωσης όπου ο χρήστης δίνει τιμές για την καθυστέρηση και το ρυθμό μετάδοσης των ζεύξεων, μία μείωση 10% σε ένα συνολικό χρόνο λήψης 0.5s δεν θα είχε ιδιαίτερη σημασία. Ωστόσο, πρέπει να τονιστεί ότι στα πραγματικά δίκτυα η από άκρο σε άκρο καθυστέρηση μπορεί να είναι έως και χιλιάδες milliseconds. Άρα στην περίπτωση αυτή, μία αντίστοιχη μείωση του χρόνου λήψης ενός αρχείου θα έχει ουσιαστική και θετική επίδραση στην αντιληπτή εμπειρία χρήστη (user experience).



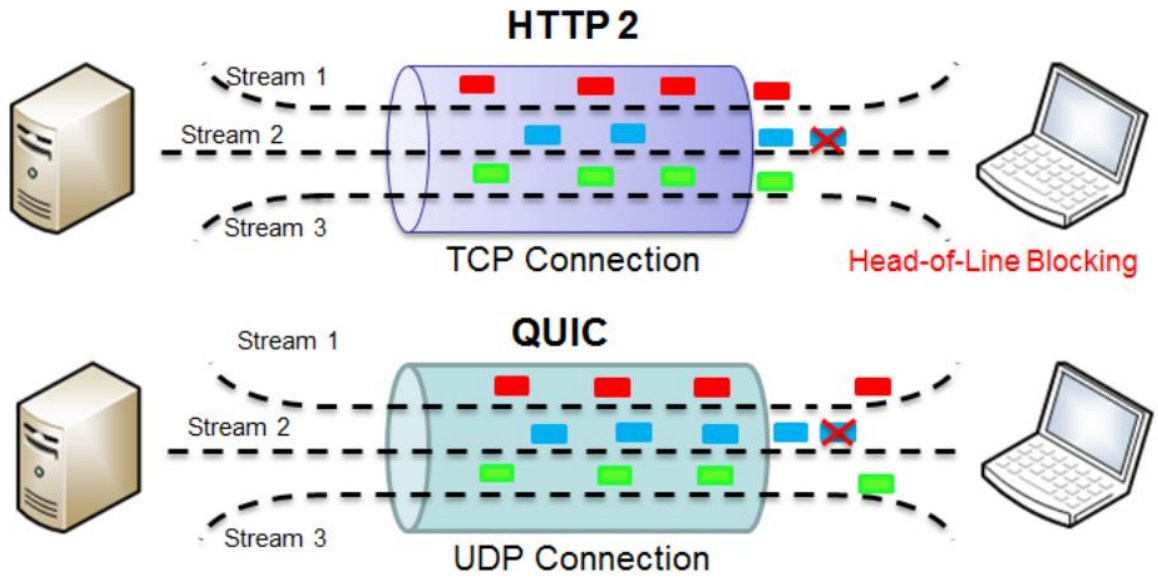
Σχήμα 6.21 *Goodput λήψης αρχείων για τα πρωτόκολλα TCP και QUIC συναρτήσει του μεγέθους αρχείου*

Επιπλέον, πρέπει να ληφθεί υπόψη ότι το QUIC επιτρέπει σε δύο τερματικά που έχουν επικοινωνήσει στο παρελθόν, να εγκαθιστούν σύνδεση με χρήση της χειραψίας 0-RTT (βλ. §3.7.1). Ο μηχανισμός αυτός επιταχύνει την εκκίνηση της αποστολής δεδομένων εφαρμογής κατά 1 RTT, συγκριτικά με το TCP. Η εφαρμογή της χειραψίας 0-RTT στο σενάριο προσομοίωσης που εξετάζεται, έδειξε ότι ο χρόνος εγκατάστασης σύνδεσης μειώνεται από τα 72ms σε 22ms. Δεδομένου ότι σχεδόν το 40% των συνδέσεων HTTPS είναι από επανεκκίνηση προηγούμενων συνδέσεων, γίνεται κατανοητό ότι το QUIC θα επιφέρει αισθητή βελτίωση στην εμπειρία του Web Browsing για τους χρήστες.

6.5 Επίδραση των QUIC Streams στη Ρυθμαπόδοση

Όπως έχει αναλυθεί στο Κεφάλαιο 3, η σημαντικότερη ίσως καινοτομία που θα φέρει το πρωτόκολλο QUIC είναι η απεικόνιση των HTTP streams στο επίπεδο μεταφοράς, με χρήση των QUIC streams. Οι σημερινές ιστοσελίδες περιέχουν ένα μεγάλο πλήθος από αντικείμενα (objects) που χρειάζεται να μεταφερθούν από τον server στον client, προκειμένου οι ιστοσελίδες να εμφανιστούν στο χρήστη. Το πρωτόκολλο HTTP/2 μπορεί να πολυπλέκει και να ξεχωρίζει τα δεδομένα από αυτά τα objects στο επίπεδο εφαρμογής, χρησιμοποιώντας HTTP streams. Ωστόσο, στο επίπεδο μεταφοράς η ιδιότητα αυτή χάνεται, καθώς το πρωτόκολλο TCP δεν διαθέτει μηχανισμό ο οποίος να αναγνωρίζει σε ποιο HTTP stream ανήκουν τα δεδομένα που μεταφέρονται στο κάθε τμήμα TCP. Το γεγονός αυτό προκαλεί το επονομαζόμενο Head of Line Blocking, ένα πρόβλημα που περιορίζει την επίδοση του TCP. Όταν συμβεί κάποια απώλεια πακέτου, το TCP στην πλευρά του παραλήπτη «παγώνει» τη διαβίβαση δεδομένων προς το επίπεδο εφαρμογής, μέχρις ότου ο αλγόριθμος συμφόρησης να αναμεταδώσει το χαμένο πακέτο. Συνεπώς, οποιοδήποτε επόμενο πακέτο, που φθάνει ακέραιο στον παραλήπτη και δυνητικά θα μπορούσε να παραδοθεί σε κάποιο HTTP stream που δεν είχε απώλεια δεδομένων κατά τη μετάδοση, αντί αυτού αποθηκεύεται σε κάποιον ενταμιευτή μέχρι να

παραληφθεί σωστά το χαμένο πακέτο (βλ. Σχήμα 6.22). Όπως γίνεται αντιληπτό, το γεγονός αυτό καθυστερεί σημαντικά το συνολικό χρόνο λήψης όλων των αντικειμένων και κατ' επέκταση την τελική εμφάνιση της ιστοσελίδας στο χρήστη. Το QUIC επιλύει το πρόβλημα αυτό με την εισαγωγή των QUIC streams στο επίπεδο μεταφοράς. Με το μηχανισμό αυτό, κάθε HTTP stream αντιστοιχίζεται μονοσήμαντα σε ένα QUIC stream, με αποτέλεσμα στις περιπτώσεις απώλειας πακέτου να μπορεί το QUIC να αναγνωρίζει σε ποιο stream ανήκαν τα δεδομένα. Έτσι, ο παραλήπτης «παγώνει» τη μεταβίβαση δεδομένων προς το επίπεδο εφαρμογής μόνο για τα HTTP streams που έχουν υποστεί απώλειες πακέτων. Τα υπόλοιπα HTTP streams μπορούν να συνεχίσουν να παραλαμβάνουν δεδομένα, όπως απεικονίζεται στο Σχήμα 6.22, αυξάνοντας έτσι το συνολικό throughput.

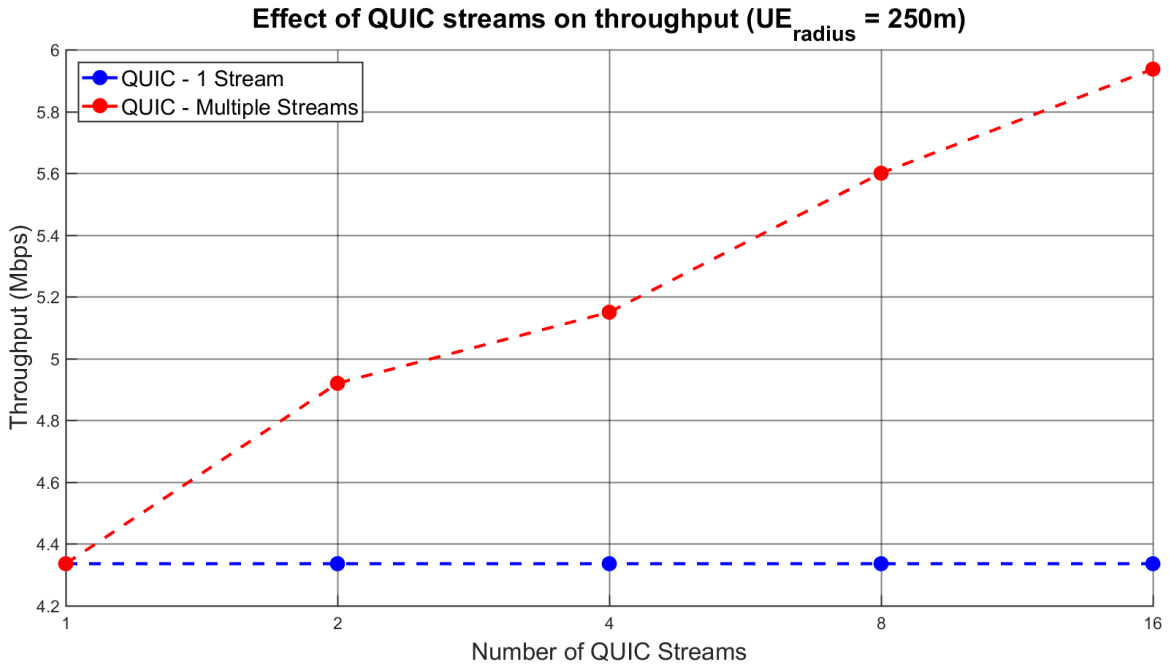


Σχήμα 6.22 *Η χρήση των QUIC streams για την επίλυση του Head of Line Blocking* [38]

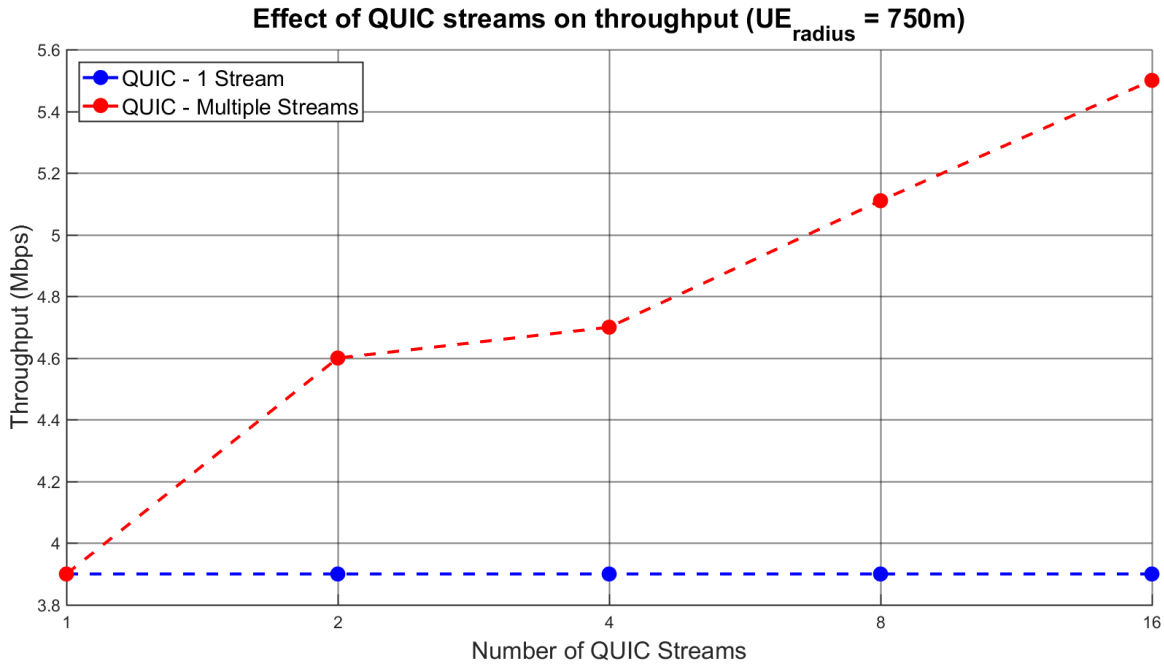
Στο τελευταίο αυτό τμήμα της παρούσας εργασίας, μελετάται η επίδραση των QUIC streams στη ρυθμαπόδοση για μετάδοση πάνω από δίκτυο LTE. Για το σκοπό αυτό, χρησιμοποιείται μία QUIC εφαρμογή που στέλνει δεδομένα μήκους 512 bytes κάθε 300μs από τον QUIC server προς ένα UE (βλ. Σχήμα 6.1). Ο αριθμός των QUIC streams είναι μεταβλητός, ενώ οι υπόλοιπες παράμετροι είναι σταθερές. Ο ρυθμός απωλειών στις ενσύρματες ζεύξεις παραμένει χαμηλός (0.5 %), προκειμένου να γίνει επικέντρωση στις απώλειες λόγω διάδοσης στον ασύρματο δίαυλο. Τα αποτελέσματα για διαφορετικές αποστάσεις του UE από τον eNB παρουσιάζονται αναλυτικά στον Πίνακα 6.6.

Απόσταση UE από eNB	**Πλήθος QUIC Streams**				
	1	**2**	**4**	**8**	**16**
	Throughput (Mbps)				
250m	4.336	4.92	5.15	5.6	5.937
750m	3.9	4.6	4.7	5.11	5.5
1500m	2.65	3.14	3.302	3.37	3.78
2500m	1.40	1.49	1.95	2.13	2.175

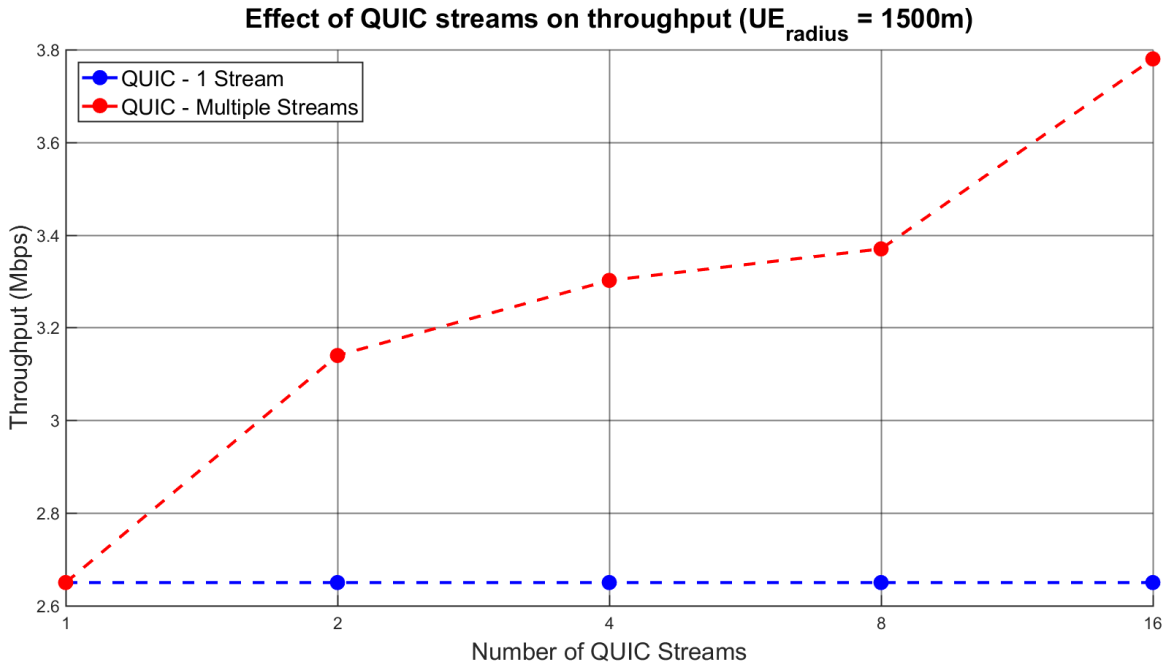
Πίνακας 6.6 *Επίδραση των QUIC streams στη ρυθμαπόδοση*



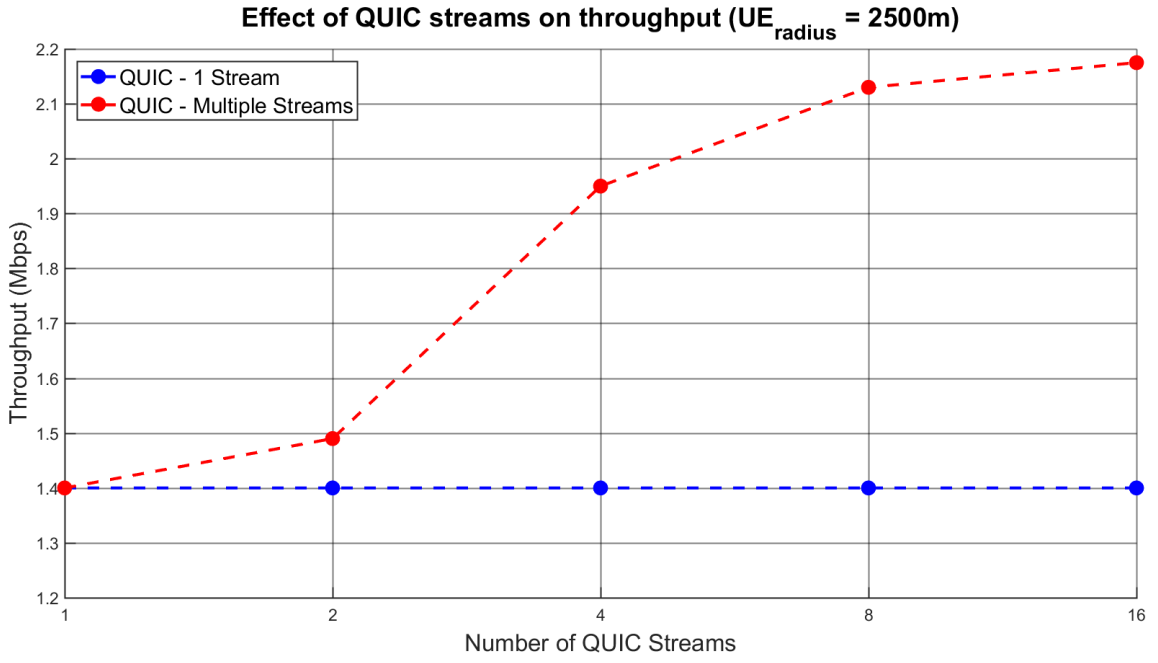
Σχήμα 6.23 *Ρυθμαπόδοση του πρωτοκόλλου QUIC συναρτήσει του πλήθους των QUIC streams (UE distance = 250m)*



Σχήμα 6.24 *Ρυθμαπόδοση του πρωτοκόλλου QUIC συναρτήσει του πλήθους των QUIC streams (UE distance = 750m)*



Σχήμα 6.25 *Ρυθμαπόδοση του πρωτοκόλλου QUIC συναρτήσει του πλήθους των QUIC streams (UE distance = 1500m)*



Σχήμα 6.26 *Ρυθμαπόδοση του πρωτοκόλλου QUIC συναρτήσει του πλήθους των QUIC streams (UE distance = 2500m)*

Συνολικά, συμπεραίνεται ότι η χρήση των QUIC streams βελτιώνει τη ρυθμαπόδοση του πρωτοκόλλου QUIC. Όπως διαισθητικά αναμενόταν, με την αύξηση του αριθμού των streams αυξάνεται και το throughput. Ωστόσο, όσο χειροτερεύει το σήμα λήψης, το κέρδος σε throughput είναι μικρότερο. Συνεπώς, η χρήση των QUIC streams αντιμετωπίζει αποτελεσματικά το πρόβλημα του Head of Line Blocking και σε ασύρματα LTE δίκτυα που μπορεί να εμφανίζουν διακυμάνσεις στο διαθέσιμο εύρος ζώνης, εξαιτίας διαλείψεων στον ασύρματο δίαυλο.

6.6 Συμπεράσματα και Μελλοντικές Επεκτάσεις

Τα αποτελέσματα των προσομοιώσεων υποδηλώνουν ότι για καλή ή μέτρια ποιότητα του σήματος λήψης στο UE, το QUIC επιτυγχάνει καλύτερη επίδοση από το TCP στη ρυθμαπόδοση σταθερής κατάστασης. Ωστόσο, για χαμηλή ποιότητα του σήματος λήψης, τα πρωτόκολλα TCP και QUIC παρουσιάζουν παρόμοια επίδοση. Επιπλέον, οι ροές QUIC καταφέρνουν και καταλαμβάνουν μεγαλύτερο ποσοστό του διαθέσιμου εύρους ζώνης συγκριτικά με τις ροές TCP, υποδεικνύοντας έτσι ότι τα πρωτόκολλα QUIC και TCP δεν είναι δίκαια μεταξύ τους, ακόμα και αν οι ραδιοπόροι του σταθμού βάσης eNB κατανέμονται ισόποσα σε όλα τα UE. Όσον αφορά το κατέβασμα αρχείων, το QUIC επιτυγχάνει μικρότερους χρόνους κατεβάσματος (download times) για όλα τα μεγέθη αρχείων που εξετάστηκαν. Τέλος, η χρήση του μηχανισμού των QUIC streams επιφέρει ουσιαστική βελτίωση στη ρυθμαπόδοση του πρωτοκόλλου QUIC, ακόμα και στη περίπτωση όπου το σήμα λήψης έχει χαμηλή ποιότητα.

Σε μελλοντική έρευνα, αξίζει να μελετηθούν οι πιθανοί τρόποι βελτιστοποίησης του QUIC σε ένα ασύρματο δίκτυο. Επειδή το QUIC κρυπτογραφεί τις πληροφορίες που περιέχονται στις επικεφαλίδες των πακέτων QUIC, οι TCP Accelerators θα πρέπει να τροποποιηθούν κατάλληλα ώστε να μπορούν να εξυπηρετούν και την κίνηση QUIC. Επιπλέον, σημαντική θα είναι και η έρευνα για τη περαιτέρω χρήση του Spin Bit στην παρακολούθηση (monitoring) της κατάστασης των συνδέσεων QUIC, με πολλούς τηλεπικοινωνιακούς παρόχους να έχουν ήδη οργανώσει κατάλληλες ομάδες έρευνας και ανάπτυξης (Research and Development - RND). Τέλος, σημαντικό πεδίο έρευνας θα αποτελέσει και η ανάπτυξη νέων αλγορίθμων συμφόρησης για το πρωτόκολλο QUIC, οι οποίοι θα αξιοποιούν βέλτιστα τα καινοτόμα χαρακτηριστικά του πρωτοκόλλου QUIC, όπως αυτά παρουσιάστηκαν στην παρούσα εργασία.

Βιβλιογραφία

[1] Cisco, “Cisco Annual Internet Report (2018–2023),” *Cisco*, pp. 1–41, 2020.

[2] J. Kurose and K. Ross, *Computer Networking: A Top-Down Approach*, 6th ed. 2013.

[3] S. Deering and R. Hinden, “Internet Protocol, Version 6 (IPv6) Specification - RFC 8200,” 2017.

[4] “Usage Statistics of HTTP/2 for Websites, August 2020.” [Online]. Available: https://w3techs.com/technologies/details/ce-http2. [Accessed: 03-Aug-2020].

[5] “IP routing - Wikipedia.” [Online]. Available: https://en.wikipedia.org/wiki/IP_routing. [Accessed: 03-Aug-2020].

[6] V. Paxson and M. Allman, “RFC 2988 - Computing TCP’s Retransmission Timer,” 2000.

[7] “TCP window scale option - Wikipedia.” [Online]. Available: https://en.wikipedia.org/wiki/TCP_window_scale_option. [Accessed: 04-Aug-2020].

[8] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan, “Rate Control for Communication Networks: Shadow Prices, Proportional Fairness,” 1998.

[9] R. Srikant, *The Mathematics of Internet Congestion Control*. Birkhauser, 2004.

[10] M. Allman and E. Blanton, “RFC 5681 - TCP Congestion Control,” 2009.

[11] D. M. Chiu and R. Jain, “Analysis of the increase and decrease algorithms for congestion avoidance in computer networks,” *Computer Networks and ISDN Systems*, vol. 17, no. 1. pp. 1–14, 1989, doi: 10.1016/0169-7552(89)90019-6.

[12] J. Padhye, V. Firoiu, D. F. Towsley, and J. F. Kurose, “Modeling TCP Reno Performance: A Simple Model and Its Empirical Validation,” 2000.

[13] A. Gurtov, “RFC 6582 - The NewReno Modification to TCP’s Fast Recovery Algorithm,” 2012.

[14] V. Vasanthi, N. Ajith Singh, M. Romen Kumar, and M. Hemalatha, “Evaluation of protocols and algorithms for improving the performance of TCP over wireless/wired network,” in *Communications in Computer and Information Science*, 2011, vol. 250 CCIS, pp. 693–697, doi: 10.1007/978-3-642-25734-6_120.

[15] L. S. Brakmo and L. L. Peterson, “TCP Vegas: End to End Congestion Avoidance on a Global Internet,” *IEEE J. Sel. Areas Commun.*, vol. 13, no. 8, pp. 1465–1480, 1995, doi: 10.1109/49.464716.

[16] R. J. La, J. Walrand, and V. Anantharam, “Issues in TCP Vegas.”

[17] S. Ha, I. Rhee, and L. Xu, “CUBIC: A New TCP-Friendly High-Speed TCP Variant.”

[18] A. Afanasyev, N. Tilley, P. Reiher, and L. Kleinrock, “Host-to-host congestion control for TCP,” *IEEE Commun. Surv. Tutorials*, vol. 12, no. 3, pp. 304–342, Sep. 2010, doi: 10.1109/SURV.2010.042710.00114.

[19] J. Katz and Y. Lindell, *Introduction to Modern Cryptography*. CRC Press, 2007.

[20] M. Belshe, R. Peon, and M. Thomson, “RFC 7540 - Hypertext Transfer Protocol Version 2 - HTTP/2,” *Internet Eng. Task Force*, 2015.

[21] “Symmetric-key algorithm - Wikipedia.” [Online]. Available: https://en.wikipedia.org/wiki/Symmetric-key_algorithm. [Accessed: 22-Jun-2020].

[22] C. Paar, J. Pelzl, and B. Preneel, *Understanding Cryptography: A Textbook for Students and Practitioners*. Springer, 2010.

[23] “Public-key cryptography - Wikipedia.” [Online]. Available: https://en.wikipedia.org/wiki/Public-key_cryptography. [Accessed: 22-Jun-2020].

[24] W. Diffie and M. E. Hellman, “New Directions in Cryptography Invited Paper.”

[25] R. L. Rivest, A. Shamir, and L. Adleman, “A Method for Obtaining Digital Signatures and Public-Key Cryptosystems.”

[26] E. Rescorla, “RFC 8446 - The Transport Layer Security TLS Protocol Version 1.3,” 2018.

[27] J. Rosenberg, “The New Waist of the Hourglass.” [Online]. Available: https://tools.ietf.org/id/draft-tschofenig-hourglass-00.html. [Accessed: 25-Jun-2020].

[28] A. Langley *et al.*, “The QUIC Transport Protocol - Design and Internet Scale Deployment,” pp. 183–196, 2017, doi: 10.1145/3098822.3098842.

[29] Google, “Next generation multiplexed transport over UDP,” 2018.

[30] M. Honda, F. Huici, C. Raiciu, J. Araujo, and L. Rizzo, “Rekindling Network Protocol Innovation with User-Level Stacks.”

[31] M. Bishop, “draft-ietf-quic-http-29 - Hypertext Transfer Protocol Version 3 (HTTP/3),” 2020.

[32] J. Mogul and S. Deering, “RFC 1191 - Path MTU discovery,” 1990.

[33] J. Iyengar, “The Maturing of QUIC.” [Online]. Available: https://www.fastly.com/blog/maturing-of-quic. [Accessed: 30-Jun-2020].

[34] Jana Iyengar (Fastly) and Martin Thomson (Mozilla), “QUIC: A UDP-Based Multiplexed and Secure Transport (draft-ietf-quic-transport-29),” 2020.

[35] J. Iyengar and I. Swett, “QUIC Loss Detection and Congestion Control (draft-ietf-

quic-recovery-27),” 2020.

[36] L. Eggert, G. Fairhurst, and G. Shepherd, “RFC 8085 - UDP Usage Guidelines,” 2017.

[37] M. Kuehlewind and B. Trammel, “ Manageability of the QUIC Transport Protocol (draft-ietf-quic-manageability-06),” 2020.

[38] Y. Cui, T. Li, C. Liu, X. Wang, and M. Kuhlewind, “Innovating transport with QUIC: Design approaches and research challenges,” *IEEE Internet Comput.*, vol. 21, no. 2, pp. 72–76, 2017, doi: 10.1109/MIC.2017.44.

[39] M. Thomson and S. Turner, “Using TLS to Secure QUIC (draft-ietf-quic-tls-29),” 2020.

[40] E. Gagliardi and O. Levillain, “Analysis of QUIC Session Establishment and Its Implementations,” *Int. Conf. Inf. Secur. Theory Pract.*, 2019.

[41] 3GPP, “TSG-RAN TS 36.300. Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN),” vol. 0, p. Version 8.9.0, 2009.

[42] P. Boyland, “The State of Mobile Network Experience ,” 2019.

[43] R. Stewart, “RFC 4960 - Stream Control Transmission Protocol,” 2007.

[44] N. LaSorte, W. J. Barnes, and H. H. Refai, “The history of orthogonal frequency division multiplexing,” *GLOBECOM - IEEE Glob. Telecommun. Conf.*, no. January, pp. 3592–3596, 2008, doi: 10.1109/GLOCOM.2008.ECP.690.

[45] I. Poole, “LTE-FDD, TDD, TD-LTE Duplex Schemes,” *Radio-electronics.com*.

[46] Telesystem Innovations, “LTE in a Nutshell : The Pysical Layer,” pp. 2–5, 2010.

[47] A. Fattah and F. Hasan Abdul-Baqi, “Polarization Division Multiplexing Coherent Optical OFDM Transmission Systems,” 2015.

[48] 3GPP, “TR 25.814, Technical Specification Group Radio Access Network; Physical Layer Aspects of Evolved UTRA.”

[49] M. Ali, N. Sukar, and M. Pal, “SC-FDMA & OFDMA in LTE physical layer,” *Int. J. Eng. Trends Technol.*, vol. 12, no. 2, 2014.

[50] J. Zyren and W. Mccoy, “Overview of the 3GPP Long Term Evolution Physical Layer,” 2007.

[51] A. Ganguly and A. Banerjee, “VLSI architecture for analog radix-4 DFT front-end in QAM–OFDM receiver,” *Analog Integr. Circuits Signal Process.*, 2019.

[52] M. R. Sriharsha, S. Dama, and K. Kuchi, “A complete cell search and synchronization in LTE,” *Eurasip J. Wirel. Commun. Netw.*, vol. 2017, no. 1, pp. 1–14, Dec. 2017, doi: 10.1186/s13638-017-0886-3.

[53] M. Matthe, “Python OFDM Example - DSPIllustrations.com.” [Online]. Available: https://dspillustrations.com/pages/posts/misc/python-ofdm-example.html. [Accessed: 16-Jul-2020].

[54] K. Liu and J. Y. Lee, “Mobile Accelerator: A New Approach to Improve TCP Performance in Mobile Data Networks,” 2011.

[55] K. Liu and J. Lee, “Impact of TCP protocol efficiency on mobile network capacity loss,” 2013.

[56] F. Ren and C. Lin, “Modeling and Improving TCP Performance over Cellular Link with Variable Bandwidth,” *IEEE TMC*, vol. 10(8), 2011.

[57] Sandvine, “TCP Optimization : Opportunities , KPIs , and Considerations - An Industry Whitepaper,” pp. 1–18.

[58] Π. Πανόπουλος, “Βελτιστοποίηση της λειτουργίας του TCP accelerator με χρήση του Mobile Edge Computing,” National Technical University of Athens, 2019.

[59] S. Ladiwala, R. Ramaswamy, and T. Wolf, “Transparent TCP acceleration,” *Comput. Commun.*, vol. 32, no. 4, pp. 691–702, 2009, doi: 10.1016/j.comcom.2008.11.036.

[60] Sandvine, “TCP Accelerator Overview: Take your network to the next level by taking control of TCP.”

[61] I. Johansson and Z. Sarker, “RFC 8298 - Self-Clocked Rate Adaptation for Multimedia,” 2017.

[62] “RADIUS - Wikipedia.” [Online]. Available: https://en.wikipedia.org/wiki/RADIUS. [Accessed: 08-Aug-2020].

[63] Sandvine, “Network Congestion Management,” vol. 6, no. 1, pp. 1–16, 2006.

[64] Sandvine, “Network Congestion Management: Considerations and Techniques,” p. 16, 2015.

[65] D. Borman, “RFC 7323 - TCP Extensions for High Performance,” 2014.

[66] R. Sisto, M. Guido, and F. Bulgarella, “QUIC Performance Measurement Algorithms to evaluate connection delay and loss rate,” Politecnico Di Torino, 2019.

[67] “ns-3 | a discrete-event network simulator for internet systems.” [Online]. Available: https://www.nsnam.org/. [Accessed: 25-Jul-2020].

[68] “LTE Module — Model Library.” [Online]. Available: https://www.nsnam.org/docs/release/3.29/models/html/lte.html. [Accessed: 25-Jul-2020].

[69] A. De Biasio, F. Chiariotti, M. Polese, A. Zanella, and M. Zorzi, “A QUIC implementation for ns-3,” *ACM Int. Conf. Proceeding Ser.*, pp. 1–8, 2019, doi: 10.1145/3321349.3321351.

[70] A. M. Kakhki, S. Jero, D. Choffnes, C. Nita-Rotaru, and A. Mislove, “Taking a long look at QUIC,” *Commun. ACM*, vol. 62, no. 7, pp. 86–94, 2019, doi: 10.1145/3330336.