

ΤΕΧΝΙΚΕΣ ΧΩΡΙΚΗΣ ΔΙΑΜΟΡΦΩΣΗΣ  
& ΧΡΟΝΟΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ  
ΓΙΑ ΒΕΛΤΙΣΤΗ ΑΠΟΔΟΣΗ ΣΕ ΑΣΥΡΜΑΤΑ ΔΙΚΤΥΑ

Εκπόνηση Διπλωματικής Εργασίας Από:  
Ιωάννα Παπαφίλη

Επιβλέπων Καθηγητής  
Δρ. Ιορδάνης Κουτσόπουλος

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ, ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ & ΔΙΚΤΥΩΝ  
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ  
ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ  
ΣΕΠΤΕΜΒΡΙΟΣ 2006



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ  
ΒΙΒΛΙΟΘΗΚΗ & ΚΕΝΤΡΟ ΠΛΗΡΟΦΟΡΗΣΗΣ  
ΕΙΔΙΚΗ ΣΥΛΛΟΓΗ «ΓΚΡΙΖΑ ΒΙΒΛΙΟΓΡΑΦΙΑ»**

Αριθ. Εισ.: 5029/1  
Ημερ. Εισ.: 21-09-2007  
Δωρεά: Συγγραφέα  
Ταξιθετικός Κωδικός: ΠΤ – ΜΗΥΤΔ  
2006  
ΠΑΠ



## Διπλωματική Εργασία

Τίτλος: **ΤΕΧΝΙΚΕΣ ΧΩΡΙΚΗΣ ΔΙΑΜΟΡΦΩΣΗΣ  
& ΧΡΟΝΟΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ  
ΓΙΑ ΒΕΛΤΙΣΤΗ ΑΠΟΔΟΣΗ ΣΕ ΑΣΥΡΜΑΤΑ  
ΔΙΚΤΥΑ**

Εκπόνηση: **Ιωάννα Παπαφίλη**

Επίβλεψη: **Δρ. Ιορδάνης Κουτσόπουλος**

Τμήμα: **ΜΗΧΑΝΙΚΩΝ Η/Υ, ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ &  
ΔΙΚΤΥΩΝ**







# Περιεχόμενα

Περιεχόμενα	x
<b>1 Εισαγωγή</b>	<b>1</b>
1.1 Γενικά	1
1.2 Κίνητρο	2
1.3 Υπόβαθρο	3
1.4 Μοντέλο σήματος	6
1.5 Μεταβλητές ελέγχου προβλήματος	8
1.6 Διατύπωση προβλήματος	9
1.7 Η συνεισφορά μας	11
1.8 Οργάνωση διπλωματικής εργασίας	12
<b>2 Χωρική Επεξεργασία Σήματος στο Δέκτη</b>	<b>13</b>
2.1 Decorrelating Detector	13
2.1.1 Περίπτωση $K \geq 2$ χρηστών	14
2.2 Successive Interference Canceller	16
2.2.1 Περίπτωση $K \geq 2$ χρηστών	17
2.3 Από κοινού χωρική επεξεργασία σήματος και έλεγχος ισχύος	19
2.3.1 Decorrelating Detector	20
2.3.2 Successive Interference Canceller	23
<b>3 Χρονοπρογραμματισμός</b>	<b>27</b>
3.1 Βέλτιστος χρονοπρογραμματισμός	28
3.1.1 Πλήθος δυνατών υποσυνόλων	28
3.1.2 Πλήθος δυνατών υποσυνόλων και μεταθέσεων	29
3.1.3 Βέλτιστη σειρά αποκωδικοποίησης	29
3.1.4 Βέλτιστη πολιτική χρονοπρογραμματισμού	30
3.2 Χρονοπρογραμματισμός βασιζόμενος στο μήκος ουράς	31
3.3 Χρονοπρογραμματισμός βασιζόμενος στη νόρμα διανύσματος καναλιού	32
3.4 Χρονοπρογραμματισμός βασιζόμενος στο μήκος ουράς και στη νόρμα διανύσματος καναλιού	33
3.5 Τυχαίος χρονοπρογραμματισμός	34



3.6 Χωρίς χρονοπρογραμματισμό . . . . .	35
<b>4 Προσομοίωση</b>	<b>37</b>
4.1 Μοντέλο προσομοίωσης . . . . .	37
4.2 Κριτήριο απόδοσης . . . . .	38
4.3 Αποτελέσματα . . . . .	39
<b>5 Σύνοψη Διπλωματικής Εργασίας</b>	<b>45</b>
5.1 Συμπεράσματα . . . . .	45
5.2 Μελλοντικές κατευθύνσεις . . . . .	46
<b>A'</b>	<b>49</b>
A'.1 Κώδικας Προσομοίωσης . . . . .	49
<b>Βιβλιογραφία</b>	<b>75</b>





# Κεφάλαιο 1

## Εισαγωγή

Σε αυτό το κεφάλαιο, παρουσιάζουμε πληροφορίες εισαγωγικού χαρακτήρα που δίνουν το κίνητρο και το υπόβαθρο αυτής της διπλωματικής εργασίας. Επίσης, περιγράφεται το μοντέλο σήματος και διατυπώνεται το κύριο πρόβλημα με το οποίο ασχολείται αυτή η διπλωματική εργασία. Αναφέρονται συνοπτικά τα νέα στοιχεία που παρουσιάζονται σε αυτή την εργασία και τέλος, παρατίθεται ανασκόπηση της σχετικής με την εργασία βιβλιογραφίας και περιγράφονται συνοπτικά οι βασικές ενότητες της διπλωματικής εργασίας.

### 1.1 Γενικά

Καθώς η χρήση του Internet και υψηλών σε απαιτήσεις εύρους ζώνης (bandwidth) εφαρμογών ολοένα αυξάνεται, μεγαλώνει και η ανάγκη για ασύρματη ευρυζωνική πρόσβαση. Μια από τις κύριες τεχνολογικές καινοτομίες που θα επηρεάσουν το μέλλον της ασύρματης ευρυζωνικής βιομηχανίας είναι η χρήση πολλαπλών κεραιών. Οι πολλαπλές κεραιές στον πομπό, είτε/και στον δέκτη σε συνδυασμό με επεξεργασία σήματος προσφέρουν πρωτοφανές κέρδος σε χωρητικότητα και αξιοπιστία της ασύρματης ζεύξης και σε αύξηση του ρυθμού μετάδοσης.

Σε αυτή τη διπλωματική εργασία, μελετάμε ένα κανάλι πολλαπλής πρόσβασης με απλές κεραιές στους πομπούς και πολλαπλή κεραία στο δέκτη (Single-Input-Multiple-Output Multiple-Access-Channel). Ο στόχος μας είναι η μεγιστοποίηση του throughput, δηλ. του σταθμισμένου αθροίσματος των ρυθμών μετάδοσης των χρηστών, όπου τα βάρη είναι τα μήκη ουρών των

χρηστών.

Τα βασικά σενάρια που μελετούνται είναι δυο:

- ανίχνευση χρηστών (multiuser detection) με τον Decorrelating Detector (DD) και
- ανίχνευση χρηστών με τον Successive Interference Canceller (SIC).

Αρχικά, αναλύουμε την αρχιτεκτονική και τη λειτουργία των δυο δεκτών και υπολογίζουμε τις εκφράσεις Λόγου Σήματος - προς - Παρεμβολή - και - Θόρυβο (Signal - to - Interference - and - Noise Ratio). Επίσης, στη μελέτη μας προστίθεται και Έλεγχος Ισχύος (Power Control). Ορίζουμε το πρόβλημα της εύρεσης της βέλτιστης κατανομής ισχύος και δίνουμε τις κατευθύνσεις για μελλοντική έρευνα.

Στο κύριο μέρος της διπλωματικής εργασίας, μελετούμε την από κοινού εφαρμογή των δυο παραπάνω τεχνικών ανίχνευσης, του Decorrelator και του Successive Interference Canceller, με τεχνικές Χρονοπρογραμματισμού (Scheduling). Ο αντικειμενικός στόχος είναι η εύρεση του βέλτιστου υποσυνόλου χρηστών που μεγιστοποιεί το throughput.

Παράλληλα με τη θεωρητική μελέτη του σεναρίου του SIMO-MAC, πραγματοποιήθηκε προσομοίωση των παραπάνω τεχνικών σε περιβάλλον Matlab, όπου μελετήθηκε η απόδοση τους. Στο τέλος της εργασίας παρουσιάζεται το μοντέλο προσομοίωσης και τα αποτελέσματα που προέκυψαν κατά την προσομοίωση αυτή.

## 1.2 Κίνητρο

Η αλματώδης αύξηση της χρήσης του Internet και εφαρμογών με υψηλές απαιτήσεις bandwidth τα τελευταία χρόνια έχει δημιουργήσει την ανάγκη για ευρυζωνική ασύρματη πρόσβαση. Η μόνη προβλέψιμη τάση είναι ότι οι ρυθμοί μετάδοσης και οι απαιτήσεις Quality-of-Service θα αυξηθούν ραγδαία. Αυτή η ζήτηση για υψηλών ταχυτήτων ασύρματη πρόσβαση/συνδεσιμότητα αποτελεί ώθηση για την ανάπτυξη προηγμένων ασύρματων ευρυζωνικών τεχνολογιών και δικτύων. Μια από τις σημαντικότερες τεχνολογικές καινοτομίες που θα επηρεάσουν το μέλλον της ευρυζωνικής ασύρματης βιομηχανίας είναι η χρήση πολλαπλών κεραιών (multiple antennas) που προσφέρει πρωτοφανή κέρδη στη χωρητικότητα και την αξιοπιστία της ασύρματης ζεύξης και στην αύξηση των ρυθμών μετάδοσης.

Οι πολλαπλές κεραίες αποδεικνύονται πολύ σημαντική τεχνολογία για το σχεδιασμό επόμενης - γενιάς, ασύρματων τηλεπικοινωνιακών συστημάτων, φωνής και/ή δεδομένων καθώς και κατά τον επανασχεδιασμό ήδη υπάρχοντων συστημάτων για την επίτευξη υψηλότερου bandwidth. Η επικοινωνία με χρήση πολλαπλών κεραιών είναι μια τεχνική αποστολής και λήψης ασύρματου σήματος, που επιτρέπει την μετάδοση περισσότερων δεδομένων χωρίς την αύξηση του εύρους ζώνης. Αυτό επιτυγχάνεται επικοινωνώντας μέσω παράλληλων χωρικών καναλιών, την ίδια χρονική στιγμή και στην ίδια συχνότητα.

Οι πολλαπλές κεραίες, στον πομπό και/ή στον δέκτη, σε συνδυασμό με επεξεργασία σήματος αυξάνει την απόδοση του ασύρματου συστήματος από την άποψη της χωρητικότητας (capacity) - μεγαλύτερος αριθμός χρηστών ανά τετραγωνικό μίλι, της κάλυψης (coverage) - μεγαλύτερη διείσδυση (penetration) - και του throughput σε ασύρματα ευρυζωνικά δίκτυα.

### 1.3 Υπόβαθρο

Μέχρι στιγμής έχουν πραγματοποιηθεί πολλές μελέτες πάνω σε συστήματα με πολλαπλές κεραίες. Παρακάτω γίνεται μια σύντομη ανασκόπηση αυτών στις οποίες βασίστηκε η παρούσα εργασία.

Οι H. Boche, E. A. Jorswieck και T. Haustein στην εργασία [8], αναζητούν τη βέλτιστη στρατηγική μετάδοσης για το κανάλι πολλαπλής πρόσβασης σε ένα κυψελωτό σύστημα, όπου ο σταθμός βάσης έχει πολλαπλή κεραία. Το κανάλι και τα μήκη ουρών των χρηστών είναι κάθε στιγμή γνωστά στο σταθμό βάσης. Ο σταθμός βάσης εκτελεί Successive Interference Cancellation. Οι συγγραφείς προτείνουν μια πολιτική ταξινόμησης για το ανοδικό κανάλι και αποδεικνύουν ότι είναι η βέλτιστη. Η πολιτική αυτή παρουσιάζεται αναλυτικότερα στο κεφάλαιο (3). Επίσης, συνδέεται η περιοχή χωρητικότητας (capacity region) του SIMO - MAC στο φυσικό επίπεδο (physical layer) με την περιοχή ευστάθειας (stability region) του αντίστοιχου συστήματος ουρών στο επίπεδο ζεύξης δεδομένων (data link layer). Τέλος, αποδεικνύεται ότι όλα τα διανύσματα ρυθμού αφίξεων που βρίσκονται εντός της περιοχής χωρητικότητας είναι εφικτά διανύσματα.

Στην εργασία [9], οι Holger Boche και Marcin Wiczanowski συνοψίζουν τα αποτελέσματα των ερευνών τους, σχετικά με σχήματα βέλτιστου χρονοπρογραμματισμού και ιδιότητες

ευστάθειας ενός συστήματος bit-queues σε ένα MIMO - MAC (Multiple - Input - Multiple - Output - MAC). Η μελέτη τους αρχίζει από την από κοινού βελτιστοποίηση του φυσικού επιπέδου και του επιπέδου ζεύξης δεδομένων, η οποία αποτελεί πεδίο της επονομαζόμενης cross - layer σχεδίασης. Αποδεικνύουν ότι η περιοχή εργοδικής χωρητικότητας (ergodic capacity region) του MIMO - MAC αντιστοιχεί στην περιοχή ευστάθειας του MIMO - MAC. Επίσης προσδιορίζουν τη βέλτιστη πολιτική χρονοπρογραμματισμού που επιτυγχάνει τη μέγιστη δυνατή περιοχή ευστάθειας, χρησιμοποιώντας χωρική πολυπλεξία των σημάτων των χρηστών και Successive Interference Cancellation στο δέκτη. Η προτεινόμενη πολιτική έχει σαν αποτέλεσμα βέλτιστη επεξεργασία σήματος τόσο στον πομπό, όσο και στον δέκτη και βασίζεται στην επίλυση ενός κυρτού προβλήματος βελτιστοποίησης. Τα αποτελέσματα της μελέτης ισχύουν τόσο για ατομικούς, όσο και για συνολικούς περιορισμούς ισχύος για το MIMO - MAC.

Στην εργασία [11], οι Holger Boche και Martin Schubert μελετούν ένα κυψελωτό, ασύρματο σύστημα όπου πραγματοποιείται από κοινού μορφοποίηση δέσμης και ανάθεση ισχύος (power allocation). Επίσης, οι χρήστες προκαλούν παρεμβολή ο ένας στον άλλο. Οι συγγραφείς θεωρούν το πρόβλημα βελτιστοποίησης του αθροίσματος των ατομικών QoS συναρτήσεων, όπου τα βάρη εξισορροπούν την απόδοση του συστήματος με την αμεροληψία (max - min fairness). Προτείνεται ένας υπολογιστικά αποδοτικός αλγόριθμος που αποδεικνύεται πως συγκλίνει μονότονα για οποιαδήποτε αρχική επιλογή βαρών. Στο τέλος, υποδεικνύεται ο τρόπος που πρέπει να επιλεγούν τα βάρη έτσι ώστε ο αλγόριθμος να συγκλίνει με βάση το κριτήριο της αμεροληψίας.

Οι Iordanis Koutsopoulos, Konstantinos P. Tsoukatos και Konstantinos Aggelis, στην εργασία [12], μελετούν το καθοδικό κανάλι ενός ασύρματου συστήματος, όπου ο σταθμός βάσης μεταδίδει σε ασύρματους χρήστες με μια πολλαπλή κεραία. Οι συγγραφείς εισαγάγουν αλγόριθμους χρονοπρογραμματισμού, οι οποίοι βασίζονται στην τυχαιότητα για να επιτύχουν μεγιστοποίηση του throughput, με χαμηλό υπολογιστικό κόστος. Οι αλγόριθμοι αυτοί, λειτουργούν σε συνδυασμό με μια τεχνική φυσικού επιπέδου, είτε με transmit beamforming, είτε με Costa precoding.

Στη συνέχεια, στην εργασία [16], οι Martin Schubert και Holger Boche θεωρούν το πρόβλημα της από κοινού χωρικής επεξεργασίας σήματος με έλεγχο ισχύος σε ένα "επίπεδο" κανάλι εκπομπής με πολλαπλή κεραία. Σε αυτό το πλαίσιο, μια κοινή στρατηγική βελτιστοποίησης είναι η μεγιστοποίηση του throughput του συστήματος υπό έναν περιορισμό ισχύος. Αυτή η στρατηγική, ωστόσο, δεν εγγυάται δίκαιο μοίρασμα του throughput μεταξύ των χρηστών,

που είναι απαιτούμενο στην περίπτωση που οι δέκτες δεν μπορούν να συνεργαστούν. Έτσι, οι συγγραφείς προτείνουν μια διαφορετική προσέγγιση, η οποία θεωρεί πάντοτε βέλτιστα εξισορροπημένους ρυθμούς μετάδοσης. Αυτή η στρατηγική μπορεί να εφαρμοσθεί μόνο αν ο σταθμός βάσης γνωρίζει τον πίνακα χωρικής συνδιακύμανσης καναλιού. Οι συγγραφείς έχουν αναπτύξει ένα θεωρητικό πλαίσιο που βασίζεται στην δυαδικότητα ανοδικού/καθοδικού καναλιού.

Οι M. Schubert και H. Boche, στην εργασία [10], μελετούν το πρόβλημα ενεργειακά αποδοτικής χωρικής επεξεργασίας σήματος για μετάδοση πολλών χρηστών, τόσο στο ανοδικό, όσο και στο καθοδικό κανάλι. Ο σταθμός βάσης είναι εξοπλισμένος με πολλαπλή κεραία, ενώ οι ασύρματοι χρήστες με απλή. Στο ανοδικό κανάλι η παρεμβολή ακυρώνεται με successive decoding, ενώ στο καθοδικό θεωρείται ιδανικό "dirty-paper" precoding. Ο σχεδιαστικός στόχος είναι είναι η ελαχιστοποίηση της συνολικής ισχύος μετάδοσης ενώ πληρούνται ατομικοί περιορισμοί SINR. Στο ανοδικό κανάλι, το πρόβλημα λύνεται με μια αναδρομική φόρμουλα, χαμηλής πολυπλοκότητας. Στο καθοδικό κανάλι, το πρόβλημα λύνεται αν εκμεταλλευθεί κανείς την δυαδικότητα μεταξύ ανοδικού και καθοδικού καναλιού. Επομένως, η λύση για το ανοδικό μεταφέρεται στο καθοδικό κανάλι. Επίσης, παρουσιάζεται πως η παραπάνω λύση μπορεί να εφαρμοσθεί στο πρόβλημα εξισορρόπησης του ρυθμού μετάδοσης σε γκαουσιανά κανάλια πολλών χρηστών. Οι συγγραφείς προτείνουν μια βέλτιστη ως προς το throughput στρατηγική μετάδοσης τόσο για κανάλιο εκπομπής, όσο και για κανάλια πολλαπλής πρόσβασης υπό έναν περιορισμό αθροίσματος ισχύος.

Στην εργασία [17], ο Mung Chiang μελετάει ένα ασύρματο δίκτυο όπου οι μεταδόσεις πραγματοποιούνται με πολλαπλά-βήματα (multihop transmissions) και οι ζευξείς δέχονται παρεμβολή. Το ερώτημα που θέτει είναι εάν μπορεί κανείς να εξισορροπήσει τον έλεγχο ισχύος στο φυσικό επίπεδο και τον έλεγχο συμφόρησης στο επίπεδο μεταφοράς για να αυξήσει την συνολική απόδοση του δικτύου. Ο συγγραφέας παρουσιάζει έναν κατανεμημένο αλγόριθμο ελέγχου ισχύος, ο οποίος, σε συνδυασμό με ήδη υπάρχοντα πρωτόκολλα ελέγχου μετάδοσης (TCPs), αυξάνει το συνολικό throughput και την απόδοση ενέργειας του δικτύου. Επίσης, αποδεικνύει την σύγκλιση του αλγορίθμου στο ολικό βέλτιστο του από κοινού ελέγχου ισχύος και ελέγχου συμφόρησης, για σύγχρονες και ασύγχρονες υλοποιήσεις. Ο ρυθμός σύγκλισης είναι γεωμετρικός. Θεωρητικά αποσπελέσματα και προσομοίωση φανερώνουν και άλλα επιθυμητά χαρακτηριστικά του προτεινόμενου αλγορίθμου, όπως ευρωστία (robustness) και ευελιξία (flexibility).



Στην εργασία [18] των Viswanath και Tse, χαρακτηρίζεται η συνολική χωρητικότητα (sum capacity) του γκαουσιανού καναλιού μετάδοσης δείχνοντας ότι το υπάρχον εσωτερικό όριο του (Marton) και το υπάρχον άνω όριο του (Sato) είναι αυστηρά για αυτό το κανάλι. Επίσης, διερευνάται η τετραμερής συσχέτιση μεταξύ του καναλιού εκπομπής, του αντίστοιχου point - to - point καναλιού (όπου οι δέκτες μπορούν να συνεργαστούν), του καναλιού πολλαπλής πρόσβασης (όπου οι ρόλοι πομπών και δεκτών αντιστρέφονται) και του αντίστοιχου point - to - point καναλιού (όπου οι πομποί μπορούν να συνεργαστούν).

## 1.4 Μοντέλο σήματος

Το αντικείμενο της μελέτης μας είναι ένα κανάλι πολλαπλής πρόσβασης MAC, όπου  $K$  ασύρματοι χρήστες με απλές - μονές κεραίες μεταδίδουν δεδομένα σε ένα σταθμό βάσης (Base Station), ο οποίος είναι όμως εξοπλισμένος με στοιχειοκεραία - πολλαπλή κεραία, με  $M \leq K$  ενεργά στοιχεία. Επισημαίνουμε ξανά πως στην συγκεκριμένη εργασία, μελετούμε δυο διαφορετικούς δέκτες, τον Decorrelating Detector και τον Successive Interference Canceller.

Κατά τη διάρκεια μιας χρονικής στιγμής, δημιουργούνται πακέτα δεδομένων από υψηλότερα επίπεδα και αποθηκεύονται στις ουρές των ασύρματων χρηστών, αναμένοντας για μετάδοση. Οι εφικτοί ρυθμοί μετάδοσης ανήκουν σε ένα διακριτό σύνολο ρυθμών μετάδοσης  $R$ , τάξης  $|R|$ , που περιλαμβάνει και το μηδενικό ρυθμό μετάδοσης.

Ο χρόνος είναι διαιρεμένος σε χρονικές στιγμές. Σε κάθε χρονική στιγμή  $t$ , ο χρονοπρογραμματιστής - σταθμός βάσης (scheduler) αποφασίζει ποιοι χρήστες θα αποκωδικοποιηθούν και με τι ισχύ θα μεταδοθούν.

Στην διπλωματική αυτή, προσεγγίζουμε το πρόβλημα που μελετάμε, τόσο από το επίπεδο ζεύξης δεδομένων (data link layer), όσο και από το φυσικό επίπεδο (physical layer). Αυτού του είδους η προσέγγιση ονομάζεται cross-layer design. Από την άποψη του φυσικού επιπέδου εφαρμόζεται η Θεωρία Πληροφορίας (Information Theory), ενώ στο επίπεδο ζεύξης δεδομένων εφαρμόζουμε τη Θεωρία Ουρών.

Όπως προείπαμε, η μελέτη μας εστιάζεται σε ένα σύστημα  $K$  ασύρματων χρηστών με απλές κεραίες, οι οποίοι μεταδίδουν πακέτα δεδομένων σε ένα σταθμό βάσης εξοπλισμένο με πολλαπλή κεραία,  $M \leq K$  στοιχείων. Πιο συγκεκριμένα, οι ασύρματοι χρήστες μεταδίδουν  $K$

ανεξάρτητα, με μηδενική μέση τιμή σήματα.

Το τυχαίο κανάλι μεταβάλλεται ανεξάρτητα σε κάθε χρονική στιγμή  $t$  ακολουθώντας την κατανομή **Rayleigh**, είναι κάθε στιγμή γνωστό στον σταθμό βάσης και μοντελοποιείται από τον πίνακα  $H$

$$H = [\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_K].$$

Τα μεταδιδόμενα σήματα ανακτώνται από  $K$  beamformers (στοιχεία πολλαπλής κεραίας), των οποίων τα βάρη βρίσκονται συγκεντρωμένα στον πίνακα

$$U = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_K] \in \mathbb{C}^{M \times K}.$$

Το διάνυσμα ενέργειας (ισχύος) στους  $K$  χρήστες είναι

$$\mathbf{p} = [p_1, p_2, \dots, p_K]^T.$$

Επιπλέον, στο σύστημα υπάρχει Προσθετικός Λευκός Γκαουσιανός Θόρυβος (Additive White Gaussian Noise)

$$w_k(t) \sim N(0, N_0), \quad N_0 = E\{|w_k(t)|^2\}.$$

Το προκύπτον SINR, όπως αυτό μετρείται στην έξοδο της  $k$ -οστής κεραίας του σταθμού βάσης, είναι:

$$SINR_k = \frac{p_k |\mathbf{u}_k^H \mathbf{h}_k|^2}{\sum_{i=1, i \neq k}^K p_i |\mathbf{u}_k^H \mathbf{h}_i|^2 + N_0}. \quad (1.4.1)$$

Όπως αναφέρθηκε στην προηγούμενη παράγραφο, οι επιθυμητοί ρυθμοί μετάδοσης

$$\mathbf{R} = (R_1, R_2, \dots, R_K)$$

ανήκουν σε ένα διακριτό σύνολο  $R$  που περιλαμβάνει και το μηδενικό ρυθμό μετάδοσης. Γενικά, η σχέση που συνδέει το ρυθμό μετάδοσης με το SINR είναι:

$$R_k = \log(1 + SINR_k). \quad (1.4.2)$$

Τώρα, όσον αφορά στο επίπεδο ζεύξης δεδομένων, κάθε ασύρματος χρήστης διατηρεί μια ουρά εξυπηρέτησης στην οποία αποθηκεύονται τα πακέτα που δημιουργούνται σε υψηλότερα επίπεδα του συστήματος κάθε χρήστη. Τα στιγμιαία (instantaneous) μήκη των ουρών όλων των χρηστών βρίσκονται συγκεντρωμένα σε ένα διάνυσμα:

$$\mathbf{Q}(t) = [Q_1(t), Q_2(t), \dots, Q_K(t)],$$

όπου το  $Q_k(t)$  αντιστοιχεί στο μήκος ουράς του χρήστη  $k$  και μετριέται σε πακέτα δεδομένων [packets].

Η χρονική εξέλιξη του μήκους ουράς ενός χρήστη  $k$  δίνεται από την αναδρομική σχέση:

$$Q_k(t+1) = Q_k(t) - R_k(t+1) + A_k(t+1), \quad (1.4.3)$$

όπου  $A_k(t)$  είναι το διάνυσμα αφίξεων και  $R_k(t)$  το διάνυσμα ρυθμών μετάδοσης. Το  $\mathbf{A}(t)$  δίνεται από τη σχέση:

$$\mathbf{A}(t) = \alpha \mathbf{L}(t), \quad (1.4.4)$$

όπου  $\alpha$  είναι μια μεταβλητή ελέγχου, μέσω της οποίας ελέγχουμε το ποσό της κίνησης που εισέρχεται στο σύστημα και  $\mathbf{L}(t)$  είναι ένα σταθερό διάνυσμα, του οποίου οι συνιστώσες είναι είτε 0, είτε 1 σύμφωνα με μια Bernoulli κατανομή.

Τέλος, ως  $S(t)$  αναπαριστούμε το υποσύνολο χρηστών που επιλέγει προς μετάδοση ο χρονοπρογραμματιστής τη χρονική στιγμή  $t$

$$\{S(t) \subseteq \{1, 2, \dots, K\}\}.$$

## 1.5 Μεταβλητές ελέγχου προβλήματος

Όπως προαναφέρθηκε σε προγενέστερη παράγραφο, ο χρόνος είναι διαιρεμένος σε χρονικές στιγμές. Σε κάθε χρονική στιγμή  $t$ , ο χρονοπρογραμματιστής αποφασίζει ποιοι χρήστες θα αποκωδικοποιηθούν και με τι ισχύ θα μεταδοθούν.

Για να επιλύσουμε το πρόβλημα της μεγιστοποίησης του throughput στο σύστημα που παρουσιάσαμε, ελέγχουμε τρεις μεταβλητές. Αυτές οι μεταβλητές ελέγχου του συστήματος είναι

$$\left\{ \begin{array}{l} \mathbf{u}_k(t) : k \in S(t) \\ p_k(t) : k \in S(t) \\ S(t) \subseteq \{1, \dots, K\} \end{array} \right\},$$

όπου  $\mathbf{u}_k(t)$  είναι το διάνυσμα χωρικής επεξεργασίας που περιέχει τα βάρη των ενεργών στοιχείων της πολλαπλής κεραίας που αντιστοιχούν στον χρήστη  $k$ , για τη χρονική στιγμή  $t$ ,

$p_k(t)$  είναι η ισχύς μετάδοσης του χρήστη  $k$  τη χρονική στιγμή  $t$  και  $S(t)$  είναι το υποσύνολο χρηστών που επιλέγει ο χρονοπρογραμματιστής τη χρονική στιγμή  $t$ .

Σε αυτή την διπλωματική εργασία, οι δυο περιπτώσεις δεκτών - αποκωδικοποιητών που θα μελετηθούν είναι:

- ο Decorrelating Detector: ένας δέκτης πολυχρηστικής ανίχνευσης που επιχειρεί να μη-δενίσει την παρεμβολή των άλλων χρηστών, ενισχύοντας ωστόσο τον θόρυβο, και
- ο Successive Interference Canceller: ένας δέκτης πολυχρηστικής ανίχνευσης που θεωρεί την παρεμβολή των χρηστών που ήδη έχουν μεταδώσει τα σήματά τους γνωστή και την αφαιρεί από το ληφθέν σήμα.

Επίσης, όπως αναφέρθηκε και σε προηγούμενη παράγραφο, κάθε στιγμή θεωρούμε ότι το κανάλι είναι γνωστό στον σταθμό βάσης. Εφόσον λοιπόν, γνωρίζουμε το κανάλι, μπορούμε πολύ εύκολα να υπολογίσουμε τα  $\mathbf{u}_k(t)$ 's για τον κάθε δέκτη. Θεωρώντας γνωστά τα  $\mathbf{u}_k(t)$ 's σε κάθε χρονική στιγμή και επιλέγοντας το υποσύνολο των χρηστών που θα μεταδοθούν και θα αποκωδικοποιηθούν από τον σταθμό βάσης, δηλ. το  $S(t) \subseteq \{1, \dots, K\}$ , θα προσπαθήσουμε να μεγιστοποιήσουμε το συνολικό throughput του συστήματος. Αντίστοιχα, θα προσπαθήσουμε να μεγιστοποιήσουμε το συνολικό throughput, επιλέγοντας την κατάλληλη ισχύ μετάδοσης για κάθε χρήστη  $p_k(t)$ 's με έλεγχο ισχύος.

Το ζητούμενο είναι σε κάθε περίπτωση, ο σταθμός βάσης να αποκωδικοποιήσει τα σήματα των ασύρματων χρηστών, με τρόπο τέτοιο ώστε το συνολικό throughput του συστήματος να είναι μέγιστο.

## 1.6 Διατύπωση προβλήματος

Ο στόχος αυτής της εργασίας, λοιπόν, είναι η εύρεση της καλύτερης πολιτικής που μεγιστοποιεί το συνολικό throughput του συστήματος:

$$I(t) = \arg \max_{I \in \mathcal{I}} \sum_{k=1}^K Q_k(t) R_k(I, t), \quad (1.6.1)$$

όπου το  $Q_k(t)$  είναι το μήκος ουράς του  $k$ -οστού χρήστη και η εξέλιξή του στο χρόνο δίνεται από τη σχέση (1.4.3), και  $R_k(I, t)$  είναι ο ρυθμός μετάδοσης που αντιστοιχεί στον  $k$ -οστό χρήστη που υπολογίζεται με βάση την σχέση (1.4.2).

**Ορισμός:** Μια πολιτική χρονοπρογραμματισμού  $I(t)$  είναι μια γενική αντιστοίχιση από το Καρτεσιανό γινόμενο του χώρου των  $K$  διανυσμάτων καναλιού  $\mathbf{h}_k \in \mathbf{C}^{M \times 1}$  και του διανύσματος μήκους ουρών  $\mathbf{Q} \in \mathbf{R}_+^K$  στο Καρτεσιανό γινόμενο του χώρου των διανυσμάτων beamforming  $\mathbf{u}_k \in \mathbf{C}^{M \times 1}$  του χώρου των διανυσμάτων κατανομής ισχύος  $\mathbf{p} \in \mathbf{R}_+^K$  και του χώρου όλων των δυνατών υποσυνόλων χρηστών  $S_M \subseteq \{1, \dots, K\}$

$$I : (\times_{k=1}^K \mathbf{C}^{M \times 1}) \times \mathbf{R}_+^K \rightarrow (\times_{k=1}^K \mathbf{C}^{M \times 1}) \times (\times_{k=1}^K \mathbf{R}_+^K) \times S_M. \quad (1.6.2)$$

Ουσιαστικά, με τον όρο πολιτική εννοούμε τις τρεις μεταβλητές ελέγχου που αναλύσαμε στην προηγούμενη παράγραφο, δηλ. το σύνολο όλων των δυνατών πολιτικών  $I$  είναι κάθε δυνατός συνδυασμός των τριών αυτών μεταβλητών ελέγχου:

$$I \equiv \left\{ \begin{array}{l} \mathbf{u}_k(t) : k \in S(t) \\ p_k(t) : k \in S(t) \\ S(t) \subseteq \{1, \dots, K\} \end{array} \right\}. \quad (1.6.3)$$

Στα επόμενα κεφάλαια, θα παρουσιασθούν και θα αναλυθούν οι μηχανισμοί ελέγχου των τριών μεταβλητών ελέγχου. Οι μηχανισμοί αυτοί είναι:

- υπολογισμός διανύσματος χωρικής επεξεργασίας σήματος (spatial processing vector computation),
- χρονοπρογραμματισμός (scheduling) και
- έλεγχος ισχύος (power control).

Ο πρώτος μηχανισμός παρουσιάζεται στο πρώτο μισό του κεφαλαίου (2) και περιλαμβάνει τον υπολογισμό των  $\mathbf{u}_k(t)$ 's. Ο δεύτερος μηχανισμός αποτελεί το κύριο θέμα με το οποίο ασχολείται η εργασία αυτή και αναλύεται εκτενώς στο κεφάλαιο (3). Όσο για τον τρίτο μηχανισμό, θα κάνουμε μια εισαγωγή και θα προτείνουμε οδούς μελλοντικής έρευνας και μελέτης στο δεύτερο μισό του κεφαλαίου (2).

Το πρόβλημα που ορίσαμε παραπάνω διέπεται από τρεις περιορισμούς, έναν περιορισμό ισότητας και δυο περιορισμούς ανισότητας. Οι περιορισμοί αυτοί αφορούν στην ισχύ μετάδοσης και

το μήκος ουράς. Πιο συγκεκριμένα, ο πρώτος περιορισμός για την ισχύ μετάδοσης είναι

$$\sum_{k=1}^K p_k = P_{con}. \quad (1.6.4)$$

Ο δεύτερος περιορισμός για την ισχύ μετάδοσης απορρέει από το φυσικό γεγονός ότι η ισχύς δεν μπορεί να πάρει αρνητικές τιμές:

$$p_k \geq 0, \quad k = 1, 2, \dots, K. \quad (1.6.5)$$

Τέλος, ο τρίτος περιορισμός αφορά στο μήκος ουράς και απορρέει επίσης από το φυσικό γεγονός ότι το μήκος είναι πάντοτε μη αρνητικό:

$$Q_k \geq 0, \quad k = 1, 2, \dots, K. \quad (1.6.6)$$

Εν κατακλείδι, ξαναγράφουμε το πρόβλημα που μελετάμε σε πλήρη μορφή:

$$\begin{aligned} I(t) = \arg \max_{I \in I} \sum_{k=1}^K Q_k(t) R_k(I, t) \\ \text{st. } \sum_{k=1}^K p_k = P_{con} \\ p_k, Q_k \geq 0, \quad k = 1, 2, \dots, K. \end{aligned} \quad (1.6.7)$$

## 1.7 Η συνεισφορά μας

Σε προηγούμενη παράγραφο αναφέραμε ότι το αντικείμενο της μελέτης μας είναι ένα κανάλι πολλαπλής πρόσβασης με  $K$  ασύρματους χρήστες με απλές κεραίες και έναν σταθμό βάσης με πολλαπλή κεραία  $M$  ενεργών στοιχείων. Στο σταθμό βάσης εκτελείται είτε Decorrelator, είτε Successive Interference Canceller. Την απόδοση αυτού του συστήματος, την μοντελοποιούμε με μια συνάρτηση ωφελιμότητας, το throughput. Ο στόχος μας, λοιπόν, είναι η μεγιστοποίηση του συνολικού throughput του συστήματος με τεχνικές χωρικής επεξεργασία σήματος και με τεχνικές χρονοπρογραμματισμού.

Παρακάτω φαίνονται τα νέα δεδομένα που παρουσιάζονται σε αυτή την εργασία.

- παρουσίαση απλών ευριστικών πολιτικών χρονοπρογραμματισμού για κάθε έναν από τους δυο δέκτες - ανιχνευτές,

- εκτέλεση πειραματικής σύγκρισης σε περιβάλλον Matlab της επίδοσης των δυο δεκτών - ανιχνευτών στην cross-layer σχεδίαση,
- εκτέλεση πειραματικής σύγκρισης της επίδοσης των απλών ευριστικών πολιτικών χρονοπρογραμματισμού με τη βέλτιστη πολιτική χρονοπρογραμματισμού για κάθε έναν από τους δυο δέκτες - ανιχνευτές και
- παρουσίαση αλγορίθμου μορφής Waterfilling για την από κοινού χωρική επεξεργασία σήματος με έλεγχο ισχύος για το ανοδικό κανάλι, όπου εκτελείται Decorrelator.

## 1.8 Οργάνωση διπλωματικής εργασίας

Το υπόλοιπο αυτής της διπλωματικής εργασίας χωρίζεται σε 5 κύριες ενότητες που καταλαμβάνουν τα κεφάλαια (2) - (5), αντίστοιχα. Παρακάτω περιγράφεται συνοπτικά το περιχόμενο κάθε κεφαλαίου.

Στο κεφάλαιο (2), περιγράφουμε τη λειτουργία και υπολογίζουμε τα διανύσματα χωρικής επεξεργασίας (spatial processing vectors) και τα SINRs, αρχικά για τον Decorrelating Detector και κατόπιν για τον Successive Interference Canceller. Επιπλέον, μελετούμε το σενάριο του καναλιού του (1.4) όταν εκτελείται από κοινού χωρική επεξεργασία σήματος με έλεγχο ισχύος.

Στο κεφάλαιο (3), εξετάζουμε διάφορες πολιτικές από κοινού μορφοποίησης δέσμης και χρονοπρογραμματισμού για το πρόβλημα μεγιστοποίησης του throughput στο κανάλι που περιγράψαμε στην παράγραφο (1.4).

Το κεφάλαιο (4) είναι αφιερωμένο στο πρακτικό μέρος της εργασίας. Περιγράφουμε λεπτομερώς το μοντέλο προσομοίωσης και το κριτήριο απόδοσης των διαφόρων τεχνικών. Στη συνέχεια παρουσιάζουμε και αναλύουμε τα αποτελέσματα της προσομοίωσης.

Τέλος, στο κεφάλαιο (5), συνοψίζουμε τα βασικότερα σημεία και παραθέτουμε τα τελικά συμπεράσματα της διπλωματικής εργασίας.

Στο Παράρτημα A, παρατίθεται ο κώδικας προσομοίωσης.

## Κεφάλαιο 2

# Χωρική Επεξεργασία Σήματος στο Δέκτη

Όπως είναι γνωστό, οι εφικτοί ρυθμοί μετάδοσης σε ένα κανάλι πολλαπλής πρόσβασης εξαρτώνται από το διάνυσμα χωρικής επεξεργασίας σήματος - διάνυσμα βαρών στοιχείων πολλαπλής κεραίας του δέκτη (spatial processing vector), σε αυτή την περίπτωση του σταθμού βάσης. Όπως αναφέρθηκε και στο εισαγωγικό κεφάλαιο, σε αυτή την εργασία, θα μελετήσουμε δυο διαφορετικά είδη δεκτών - ανιχνευτών:

- τον Decorrelating Detector και
- τον Successive Interference Canceller.

Στο πρώτο μισό του κεφαλαίου, περιγράφουμε τη λειτουργία και τη δομή αυτών των δεκτών και τον τρόπο που αυτοί μοντελοποιούνται στο σύστημά μας. Στο δεύτερο μισό του κεφαλαίου, μελετούμε την από κοινού χωρική επεξεργασία με κάθε έναν από τους δυο δέκτες και έλεγχο ισχύος.

### 2.1 Decorrelating Detector

Ο Decorrelator είναι ένας δέκτης - ανιχνευτής (multiuser detector) που βασίζεται σε μια απλή ιδέα, την αντιστροφή καναλιού, όπως θα δούμε και παρακάτω. Τα πλεονεκτήματα του συγκεκριμένου δέκτη συνοψίζονται στα εξής:



- μπορεί να λειτουργήσει σε κατανεμημένη μορφή, από την άποψη ότι η αποκωδικοποίηση κάθε χρήστη μπορεί να πραγματοποιηθεί εντελώς ανεξάρτητα και
- παρουσιάζει χαμηλή πολυπλοκότητα τόσο στη δομή, όσο και στη λειτουργία του.

Το βασικό μειονέκτημα του συγκεκριμένου δέκτη είναι η ενίσχυση του θορύβου του συστήματος. Επίσης, σε απλό κανάλι (φυσικό επίπεδο) επιτυγχάνει χαμηλότερους ρυθμούς μετάδοσης από τον Successive Interference Canceller που θα παρουσιάσουμε παρακάτω.

### 2.1.1 Περίπτωση $K \geq 2$ χρηστών

Ας ξεκινήσουμε την περιγραφή του Decorrelator γράφοντας την εξίσωση του καναλιού σε μορφή πινάκων:

$$\mathbf{y} = P\mathbf{H}\mathbf{x} + \mathbf{w}, \quad (2.1.1)$$

όπου  $\mathbf{y} := [y_1, y_2, \dots, y_K]^H$  είναι το διάνυσμα που περιέχει τα λαμβανόμενα στον σταθμό βάσης σήματα,  $P = \text{diag}(\sqrt{p_1}, \sqrt{p_2}, \dots, \sqrt{p_K})$  είναι διαγώνιος πίνακας που περιέχει τις ισχύεις μετάδοσης των δυο σημάτων,

$$H = \begin{bmatrix} \mathbf{h}_1^H \\ \mathbf{h}_2^H \\ \vdots \\ \mathbf{h}_K^H \end{bmatrix}$$

είναι ο πίνακας καναλιού όπου ο τελεστής  $(\cdot)^H$  δηλώνει το ανάστροφο-συζυγές διάνυσμα και  $\mathbf{x} := [x_1, x_2, \dots, x_K]^H$  είναι το διάνυσμα που περιέχει τα ανεξάρτητα σήματα  $x_k$  που στέλνουν οι ασύρματοι χρήστες. Τέλος, το  $\mathbf{w}$  είναι λευκός Γκαουσιανός θόρυβος με μηδενική μέση τιμή και πίνακα συνδιακύμανσης  $\sigma^2 I$ .

Πολλαπλασιάζοντας και τα δυο μέλη της εξίσωσης (2.1.1), πρώτα με τον ανάστροφο του πίνακα καναλιού, δηλ. με  $H^H$  και ύστερα με  $(H^H H)^{-1}$ , όπου ο τελεστής  $(\cdot)^H$  δηλώνει τον ανάστροφο - συζυγή πίνακα και ο τελεστής  $(\cdot)^{-1}$  δηλώνει τον αντίστροφο, τότε η εξίσωση (2.1.1) παίρνει τη μορφή

$$(H^H H)^{-1} H^H \mathbf{y} = P\mathbf{x} + (H^H H)^{-1} H^H \mathbf{w}. \quad (2.1.2)$$

Από τη βιβλιογραφία γνωρίζουμε ότι ο πίνακας

$$H^\dagger = (H^H H)^{-1} H^H \quad (2.1.3)$$

ονομάζεται **ψευδοαντίστροφος**.

Στην περίπτωση του Decorrelator, ο λόγος Σήματος - προς - Παρεμβολή - και - Θόρυβο για τον  $k$ -οστό χρήστη είναι

$$SINR_k = \frac{p_k}{\sigma_w^2} = \frac{p_k |u_k^H h_k|^2}{\sigma^2 |u_k|^2} \quad (2.1.4)$$

και το διάνυσμα χωρικής επεξεργασίας είναι

$$u_k = [H^\dagger]_{k,:} \quad (2.1.5)$$

Στην ειδική περίπτωση που ο πίνακας  $H$  είναι τετράγωνος, δηλ.  $M = K$ , και αντιστρέψιμος, τότε ο ψευδοαντίστροφος ισούται με τον αντίστροφο

$$H^\dagger = H^{-1}. \quad (2.1.6)$$

Σε αυτή την περίπτωση, η εξίσωση (2.1.2) γράφεται ως εξής

$$H^{-1} \mathbf{y} = P \mathbf{x} + H^{-1} \mathbf{w}. \quad (2.1.7)$$

Σε κάθε περίπτωση από τις δυο (2.1.2), (2.1.7), παρατηρούμε πως η  $k$ -οστή συνιστώσα της εξίσωσης καναλιού δεν δέχεται παρεμβολή από άλλους χρήστες, δηλ. το  $x_k$  είναι ανεξάρτητο από όλα τα  $\{x_j\}, j \neq k$ . Συνεπώς, η μοναδική πηγή παρεμβολής είναι ο προσθετικός λευκός γκαουσιανός θόρυβος. Χάρη σε αυτή του την ιδιότητα, έχει δοθεί στο συγκεκριμένο δέκτη η ονομασία Decorrelator ή Decorrelating Detector, δηλ. *αποσυσχετιστής*.

### Παράδειγμα $K = 2$ χρηστών

Ας μελετήσουμε τον Decorrelator σε κανάλι με  $K = 2$  χρήστες. Γράφουμε την εξίσωση καναλιού για το  $2 \times 2$  κανάλι

$$H^{-1} \mathbf{y} = P \mathbf{x} + H^{-1} \mathbf{w} \quad \Leftrightarrow$$

$$H^{-1} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = P \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + H^{-1} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}, \quad (2.1.8)$$

όπου

$$H = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix}, \quad P = \text{diag}(\sqrt{p_1}, \sqrt{p_2}).$$

Ο θόρυβος που δέχεται το σύστημα είναι

$$\tilde{\mathbf{w}} = H^{-1}\mathbf{w} \Leftrightarrow$$

$$\begin{bmatrix} \tilde{w}_1 \\ \tilde{w}_2 \end{bmatrix} = \frac{1}{(h_{11}h_{22} - h_{12}h_{21})} \begin{bmatrix} h_{22} & -h_{12} \\ -h_{21} & h_{11} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}$$

Η διακύμανση της πρώτης συνιστώσας του θορύβου  $\tilde{w}_1$  είναι

$$\sigma_{\tilde{w}_1}^2 = E[\tilde{w}_1^2] - E^2[\tilde{w}_1] \Rightarrow \sigma_{\tilde{w}_1}^2 = \frac{h_{21}^2 + h_{22}^2}{(h_{11}h_{22} - h_{12}h_{21})^2} \sigma^2. \quad (2.1.9)$$

Ο λόγος Σήματος - προς - Παρεμβολή - και - Θόρυβο θα είναι

$$SINR_1 = \frac{p_1}{\sigma_{\tilde{w}_1}^2} = \frac{p_1 (u_1^H h_1)^2}{\sigma^2 |u_1|^2}, \quad u_1 = \frac{1}{(h_{11}h_{22} - h_{12}h_{21})} \begin{bmatrix} h_{22} \\ -h_{12} \end{bmatrix} \quad (2.1.10)$$

Όμοια υπολογίζουμε τον λόγο Σήματος - προς - Παρεμβολή - και - Θόρυβο για τον χρήστη 2

$$SINR_2 = \frac{p_2}{\sigma_{\tilde{w}_2}^2} = \frac{p_2 (u_2^H h_2)^2}{\sigma^2 |u_2|^2}, \quad u_2 = \frac{1}{(h_{11}h_{22} - h_{12}h_{21})} \begin{bmatrix} -h_{12} \\ h_{11} \end{bmatrix} \quad (2.1.11)$$

## 2.2 Successive Interference Canceller

Η περιοχή χωρητικότητας  $C$  του ανοδικού γκαουσιανού καναλιού χαρακτηρίζεται από το σύνολο όλων των ρυθμών μετάδοσης  $(R_1, R_2)$  που ικανοποιούν τους ακόλουθους τρεις περιορισμούς

$$\begin{aligned} R_1 &\leq \log \left( 1 + \frac{p_1}{N_0} \right), \\ R_2 &\leq \log \left( 1 + \frac{p_2}{N_0} \right), \\ R_1 + R_2 &\leq \log \left( 1 + \frac{p_1 + p_2}{N_0} \right). \end{aligned} \quad (2.2.1)$$

Και οι τρεις περιορισμοί είναι φυσικοί. Οι δυο πρώτοι δηλώνουν ότι ο ρυθμός μετάδοσης ενός χρήστη δεν μπορεί να υπερβαίνει τη χωρητικότητα της ζεύξης από - σημείο - σε - σημείο (point

- to - point link) με τον άλλο χρήστη απόντα από το σύστημα - αυτοί οι περιορισμοί καλούνται επίσης όρια μονού - χρήστη (single - user bounds). Ο τρίτος περιορισμός λέει ότι το συνολικό throughput δεν μπορεί να υπερβεί τη χωρητικότητα ενός point-to-point AWGN καναλιού με το άθροισμα των λαμβανόμενων ισχύων των δυο χρηστών. Αυτός ο περιορισμός ισχύει, εφόσον τα σήματα των δυο χρηστών είναι ανεξάρτητα και η ισχύς του συνολικού λαμβανόμενου σήματος ισούται με το άθροισμα των ισχύων των επιμέρους λαμβανόμενων σημάτων.

Ωστόσο, είναι εφικτό ενώ ο χρήστης 1 επιτυγχάνει το δικό του όριο μονού - χρήστη, την ίδια στιγμή ο χρήστης 2 να έχει μη μηδενικό ρυθμό μετάδοσης:

$$R_2^* = \log \left( 1 + \frac{p_1 + p_2}{N_0} \right) - \log \left( 1 + \frac{p_1}{N_0} \right) = \log \left( 1 + \frac{p_2}{p_1 + N_0} \right) \quad (2.2.2)$$

Πως συμβαίνει αυτό: Κάθε χρήστης μεταδίδει τα δεδομένα του. Ο δέκτης αποκωδικοποιεί την πληροφορία των δυο χρηστών σε δυο στάδια. Στο πρώτο στάδιο, αποκωδικοποιεί τα δεδομένα του χρήστη 2, θεωρώντας το σήμα του χρήστη 1 ως Γκαουσιανή παρεμβολή. Ο μέγιστος ρυθμός μετάδοσης που μπορεί να επιτύχει ο χρήστης 2 δίνεται από την εξίσωση (2.2.2). Μόλις ο δέκτης αποκωδικοποιήσει τα δεδομένα του χρήστη 2, μπορεί να ανακατασκευάσει το σήμα του χρήστη 2 και να το αφαιρέσει από το συνολικό λαμβανόμενο σήμα. Ο δέκτης μπορεί έπειτα να αποκωδικοποιήσει τα δεδομένα του χρήστη 1. Αφού πλέον υπάρχει μόνο ο Γκαουσιανός θόρυβος στο σύστημα, ο μέγιστος ρυθμός μετάδοσης που μπορεί να επιτύχει ο χρήστης 1 είναι το όριο μονού - χρήστη

$$R_1^* = \log \left( 1 + \frac{p_1}{N_0} \right). \quad (2.2.3)$$

Χάρη σε αυτή την ιδιότητα σταδιακής αφαίρεσης της παρεμβολής από το λαμβανόμενο σήμα, αυτός ο δέκτης ονομάζεται Successive Interference Canceller (SIC) ή απλά Successive Cancellation decoder.

### 2.2.1 Περίπτωση $K \geq 2$ χρηστών

Σε αυτή την παράγραφο, θα μελετήσουμε τον Successive Interference Canceller σε κανάλι με  $K \geq 2$  χρήστες και  $M \leq K$  ενεργά στοιχεία πολλαπλής κεραίας στο σταθμό βάσης. Η παρακάτω ανάλυση βασίζεται στον ορισμό της δομής Βέλτιστου Γραμμικού Δέκτη που δίνεται στην εργασία [18].

Γράφουμε την εξίσωση καναλιού για το  $K \times M$  κανάλι

$$\mathbf{y} = PH\mathbf{x} + \mathbf{w} \Leftrightarrow \begin{bmatrix} y_1 \\ \vdots \\ y_K \end{bmatrix} = PH \begin{bmatrix} x_1 \\ \vdots \\ x_K \end{bmatrix} + \begin{bmatrix} w_1 \\ \vdots \\ w_K \end{bmatrix}, \quad (2.2.4)$$

όπου

$$H = [\mathbf{h}_1, \dots, \mathbf{h}_K] \\ P = \text{diag}(p_1, \dots, p_K).$$

Θεωρούμε σταθερές ισχύεις μετάδοσης  $p_k$ , σταθερά διανύσματα καναλιού  $\mathbf{h}_k$  και σταθερή σειρά αποκωδικοποίησης  $\pi = 1, \dots, K$ . Ορίζουμε τους πίνακες συνδιακύμανσης παρεμβολής - θορύβου

$$Z = HPH^H + \sigma^2 I, \quad (2.2.5)$$

$$Z_k = H_k P_k H_k^H + \sigma^2 I, \quad (2.2.6)$$

όπου

$$H_k = [0, \dots, 0, \mathbf{h}_{k+1}, \dots, \mathbf{h}_K], \quad (2.2.7)$$

$$P_k = \text{diag}(p_1, \dots, p_K). \quad (2.2.8)$$

Παρατηρούμε ότι οι πίνακες  $Z$  και  $Z_k$  είναι συμμετρικοί και θετικά ορισμένοι.

Από τη Γραμμική Άλγεβρα γνωρίζουμε ότι ένας πίνακας  $A$  μπορεί να γραφεί σαν γινόμενο τριών πινάκων  $A = LDU$  ( $LDU$  παραγοντοποίηση), όπου οι πίνακες  $L$  και  $U$  είναι κάτω και άνω τριγωνικός αντίστοιχα και έχουν μονάδες στη διαγώνιο και ο πίνακας  $D$  είναι διαγώνιος. Ωστόσο, αν ο πίνακας  $A$  είναι συμμετρικός, τότε ο άνω τριγωνικός  $U$  είναι ο ανάστροφος του κάτω τριγωνικού  $L$ . Άρα ο  $A = LDU$  γίνεται  $A = LDL^T$ . Οπότε οι πίνακες  $Z$  και  $Z_k$  μπορούν να γραφούν αντίστοιχα

$$Z = HPH^H + \sigma^2 I = LDL^H, \quad (2.2.9)$$

$$Z_k = H_k P_k H_k^H + \sigma^2 I = L_k D_k L_k^H. \quad (2.2.10)$$

Τα βέλτιστα διανύσματα χωρικής επεξεργασίας  $\mathbf{u}_k$  είναι εκείνα που μεγιστοποιούν τον λόγο Σήματος - προς - Παρεμβολή - και - Θόρυβο  $SINR$ . Έτσι, λοιπόν για να βρούμε τα βέλτιστα

διανύσματα  $\mathbf{u}_k$ 's γράφουμε

$$\max_{\mathbf{u}_k \neq 0} SINR_k = \max_{\mathbf{u}_k \neq 0} \frac{p_k (\mathbf{u}_k^H \mathbf{h}_k)^2}{\mathbf{u}_k^H \mathbf{Z}_k \mathbf{u}_k} \Rightarrow \quad (2.2.11)$$

$$\max_{\mathbf{u}_k \neq 0} SINR_k = p_k \max_{\alpha_k \neq 0} \frac{\alpha_k^H D_k^{-\frac{1}{2}} L_k^H \mathbf{h}_k \mathbf{h}_k^H L_k D_k^{-\frac{1}{2}} \alpha_k}{\alpha_k^H \alpha_k}, \quad (2.2.12)$$

όπου  $\alpha_k = D_k^{-\frac{1}{2}} L_k^H \mathbf{u}_k$ .

Επομένως, το όρισμα που μεγιστοποιεί την έκφραση για το  $SINR$  είναι

$$\alpha_k = D_k^{-\frac{1}{2}} L_k^H \mathbf{h}_k \quad (2.2.13)$$

και η δομή βέλτιστου δέκτη είναι

$$\mathbf{u}_k = \mathbf{Z}_k^{-1} \mathbf{h}_k. \quad (2.2.14)$$

Τελικά, ο λόγος Σήματος - προς - Παρεμβολή - και - Θόρυβο για τον  $k$ -στό δέκτη είναι

$$SINR_k = p_k \mathbf{h}_k^H \mathbf{Z}_k^{-1} \mathbf{h}_k. \quad (2.2.15)$$

## 2.3 Από κοινού χωρική επεξεργασία σήματος και έλεγχος ισχύος

Στην παρούσα παράγραφο θα μελετήσουμε το πρόβλημα (1.6.7) με από κοινού χωρική επεξεργασία σήματος και έλεγχο ισχύος. Καταρχήν, θα επαναδιατυπώσουμε το πρόβλημα για τους δυο δέκτες, τον Decorrelator και τον Successive Interference Canceller, ύστερα θα μελετήσουμε την κυρτότητα των δυο νέων προβλήματος ως προς την ισχύ μετάδοσης και τέλος, θα προσπαθήσουμε να προσδιορίσουμε τη βέλτιστη κατανομή ισχύος που μεγιστοποιεί το throughput, για κάθε ένα από τα προβλήματα αυτά.

### 2.3.1 Decorrelating Detector

Το πρόβλημα μας είναι, όπως έχουμε ξαναπεριγράψει, η μεγιστοποίηση του συνολικού throughput του συστήματος. Το βέλτιστο throughput  $C_{sum}$  (sum capacity) ορίζεται ως το μέγιστο σταθμισμένο άθροισμα των ρυθμών μετάδοσης που μπορούν να επιτευχθούν από κοινού για απειροελάχιστα μικρό BER (bit-error-rate), όπου τα βάρη είναι τα μήκη ουρών των χρηστών.

Το άθροισμα όλων των ατομικών χωρητικότητων των χρηστών που έχουν επιλεχθεί από τον χρονοπρογραμματιστή μπορεί να γραφεί ως συνάρτηση της κατανομής ισχύος για το ανοδικό κανάλι

$$f(\mathbf{p}) = \sum_{k=1}^M Q_k R_k = \sum_{k=1}^M Q_k \log(1 + SINR_k(\mathbf{p})). \quad (2.3.1)$$

Οπότε, το πρόβλημα μεγιστοποίησης του throughput ως προς  $\mathbf{p}$  γράφεται ως εξής

$$\begin{aligned} C_{sum} &= \max_{\mathbf{p}} f(\mathbf{p}), \\ \text{st. } \sum_{k=1}^M p_k &= P_{con}, \\ Q_k, p_k &\geq 0, \quad 1 \leq k \leq M. \end{aligned} \quad (2.3.2)$$

#### Ανάλυση κυρτότητας

Για να εξετάσουμε αν η συνάρτηση  $f(\mathbf{p})$  έχει βέλτιστο (τοπικό ή ολικό), πρέπει να μελετήσουμε την κυρτότητα της  $f(\mathbf{p})$  ως προς  $p_k$ . Ξαναγράφουμε την εξίσωση (2.1.4) του SINR και το διάνυσμα χωρικής επεξεργασίας που αντιστοιχεί στον  $k$ -οστό χρήστη

$$SINR_k = \frac{p_k}{\sigma_{\tilde{w}_k}^2} = \frac{p_k (\mathbf{u}_k^H \mathbf{h}_k)^2}{\sigma^2 |\mathbf{u}_k|^2}, \quad \mathbf{u}_k = [H^{-1}]_{k,:} \quad (2.3.3)$$

Χάρην ευκολίας θα συνεχίσουμε να χρησιμοποιούμε το συμβολισμό  $\mathbf{u}_k$ , γνωρίζοντας ωστόσο ότι ο υπολογισμός της συγκεκριμένης ποσότητας είναι άμεσος δεδομένου ότι το κανάλι είναι γνωστό στον σταθμό βάσης. Οπότε, η συνάρτηση του throughput  $f(\mathbf{p})$  που θέλουμε να μεγιστοποιήσουμε ξαναγράφεται ως εξής

$$f(\mathbf{p}) = \sum_{k=1}^M Q_k R_k = \sum_{k=1}^M Q_k \log \left( 1 + \frac{p_k |\mathbf{u}_k^H \mathbf{h}_k|^2}{\sigma^2 |\mathbf{u}_k|^2} \right) \quad (2.3.4)$$

Η δεύτερη παράγωγος της  $f(\mathbf{p})$  ως προς  $p_k$  είναι πάντοτε αρνητική

$$\frac{\theta^2 f(\mathbf{p})}{\theta p_k^2} = -\frac{Q_k |\mathbf{u}_k \mathbf{h}_k|^4}{(\sigma^2 |\mathbf{u}_k|^2 + p_k |\mathbf{u}_k \mathbf{h}_k|^2)^2} < 0. \quad (2.3.5)$$

Επομένως, η συνάρτηση  $f(\mathbf{p})$  είναι κοίλη ως προς  $p_k$ . Από τη Θεωρία Βελτιστοποίησης, γνωρίζουμε ότι αυτό σημαίνει πως η συνάρτηση έχει μοναδικό ολικό μέγιστο.

### Αναζήτηση ολικού βέλτιστου - Waterfilling

Όπως γνωρίζουμε, στο ανοδικό κανάλι κανάλι κάθε χρήστης έχει το δικό του ατομικό απόθεμα ενέργειας (ισχύος)  $p_k^{max}$ . Αυτό σημαίνει πως σε κάθε χρονική στιγμή  $t$ , κάθε χρήστης μπορεί να χρησιμοποιήσει το πολύ  $p_k^{max}$  ισχύ για να μεταδώσει.

Στην κλασική περίπτωση χωρικής επεξεργασίας σήματος στο ανοδικό κανάλι, ο έλεγχος ισχύος έχει νόημα αφού ο κάθε χρήστης δεν μπορεί ανεξέλεγκτα να μεταδώσει με τη μέγιστη ισχύ που έχει στη διάθεσή του, γιατί θα αυξηθεί η παρεμβολή που προκαλεί στους άλλους χρήστες και θα μειωθεί το συνολικό throughput. Σε αυτή την περίπτωση, η συνάρτηση του throughput αποδεικνύεται κοίλη ως προς  $p_k$  και η βέλτιστη κατανομή ισχύος υπολογίζεται σε αυτή την περίπτωση με βάση τον αναδρομικό αλγόριθμο JOCP που προτάθηκε στην εργασία [17].

Στην περίπτωση όμως του Decorrelator που μελετούμε εδώ, παρατηρούμε ότι οι συνιστώσες κατανομής ισχύος  $p_k$  βρίσκονται μονάχα στον αριθμητή της έκφρασης του  $SINR$ . Αυτό συμβαίνει χάρη στην ιδιαίτερη δομή και λειτουργία του Decorrelator που ακυρώνει την παρεμβολή από τους άλλους χρήστες και λαμβάνει υπόψη μόνο τον λευκό γκαουσιανό θόρυβο. Συνεπώς, η συνάρτηση του throughput (2.3.4) είναι διαχωρίσιμη στις επιμέρους συναρτήσεις  $f(p_k)$

$$f(\mathbf{p}) = \sum_{k=1}^K f(p_k). \quad (2.3.6)$$

Αυτό σημαίνει πως, μπορούμε να δόσουμε όσο το δυνατόν μεγαλύτερη ισχύ σε έναν χρήστη. Συγκεκριμένα, όσο περισσότερη ισχύ δίνουμε σε έναν χρήστη, τόσο αυξάνεται το συνολικό throughput του συστήματος. Οπότε, σε κάθε χρονική στιγμή, οι χρήστες που επιλέγονται να μεταδώσουν, μεταδίδουν τα δεδομένα τους στον σταθμό βάσης με τη μέγιστη ισχύ που έχουν στη διάθεσή τους  $p_k^{max}$ . Καταλήγουμε, λοιπόν, στο συμπέρασμα πως δεν υφίσταται θέμα ελέγχου ισχύος για την περίπτωση του Decorrelator στο ανοδικό κανάλι.



Ας θεωρήσουμε ένα σενάριο όπου όλοι οι χρήστες του ανοδικού καναλιού που μελετούμε χρησιμοποιούν ισχύ που λαμβάνουν από την ίδια 'δεξαμενή' ισχύος. Σε αυτή την περίπτωση, θα πρέπει να ισχύει

$$\sum_{k=1}^M p_k \leq P_{con}. \quad (2.3.7)$$

Σε αυτό το υποθετικό σενάριο, ένας χρήστης  $k$  δεν μπορεί να χρησιμοποιήσει όλη την ισχύ που υπάρχει στην δεξαμενή, αφού από την ίδια δεξαμενή τροφοδοτούνται και οι υπόλοιποι χρήστες  $j \neq k$ . Συνεπώς, σε αυτή την περίπτωση, ο έλεγχος ισχύος έχει νόημα. Ξαναγράφουμε το πρόβλημα που θέλουμε να λύσουμε (2.3.2)

$$\begin{aligned} C_{sum} &= \max_{\mathbf{p}} f(\mathbf{p}), \\ \text{st. } \sum_{k=1}^M p_k &= P_{con}, \\ Q_k, p_k &\geq 0, \quad 1 \leq k \leq M. \end{aligned} \quad (2.3.8)$$

Η Lagrangian του παραπάνω προβλήματος είναι

$$l(\mathbf{p}, \lambda, \mathbf{m}) = - \sum_{k=1}^M Q_k \log \left( 1 + \frac{p_k |\mathbf{u}_k^H \mathbf{h}_k|^2}{\sigma^2 |\mathbf{u}_k|^2} \right) + \lambda \left( \sum_{k=1}^M p_k - P_{con} \right) + \sum_{k=1}^M \mu_k p_k \quad (2.3.9)$$

Οι συνθήκες Karush-Kuhn-Tucker που προσδιορίζουν τη βέλτιστη λύση του προβλήματος είναι

$$\frac{\partial l}{\partial p_k^*} = - \frac{Q_k}{1 + \frac{p_k^* |\mathbf{u}_k^H \mathbf{h}_k|^2}{\sigma^2 |\mathbf{u}_k|^2}} \frac{|\mathbf{u}_k^H \mathbf{h}_k|^2}{(\sigma^2 |\mathbf{u}_k|^2)} + \lambda^* - \mu_k^* = 0 \quad (2.3.10)$$

$$\mu_k^* p_k^* = 0 \quad (2.3.11)$$

$$\mu_k^* \geq 0 \quad (2.3.12)$$

$$\lambda^* \geq 0 \quad (2.3.13)$$

$$\sum_{k=1}^M p_k^* = P_{con} \quad (2.3.14)$$

$$p_k^* \geq 0 \quad (2.3.15)$$

Επιλύοντας την εξίσωση (2.3.10) ως προς  $\mu_k^*$  και αντικαθιστώντας στην εξίσωση (2.3.11) προκύπτει

$$p_k^* \left( \lambda - \frac{Q_k |\mathbf{u}_k^H \mathbf{h}_k|^2}{\sigma^2 |\mathbf{u}_k|^2 + p_k^* |\mathbf{u}_k^H \mathbf{h}_k|^2} \right) = 0 \quad (2.3.16)$$

Διακρίνουμε περιπτώσεις:

- αν  $p_k^* > 0$ , τότε  $\lambda^* = \frac{Q_k |\mathbf{u}_k^H \mathbf{h}_k|^2}{\sigma^2 |\mathbf{u}_k|^2 + p_k^* |\mathbf{u}_k^H \mathbf{h}_k|^2}$

- αν  $\lambda^* < \frac{Q_k |\mathbf{u}_k^H \mathbf{h}_k|^2}{\sigma^2 |\mathbf{u}_k|^2}$ , τότε  $p_k^* = \frac{Q_k}{\lambda^*} - \frac{\sigma^2 |\mathbf{u}_k|^2}{|\mathbf{u}_k^H \mathbf{h}_k|^2}$
- αν  $\lambda^* = \frac{Q_k |\mathbf{u}_k^H \mathbf{h}_k|^2}{\sigma^2 |\mathbf{u}_k|^2}$ , τότε  $p_k^* = 0$

2. αν  $\lambda^* > \frac{Q_k |\mathbf{u}_k^H \mathbf{h}_k|^2}{\sigma^2 |\mathbf{u}_k|^2 + p_k^* |\mathbf{u}_k^H \mathbf{h}_k|^2}$ , τότε  $p_k^* = 0$

Τελικά, προκύπτει ότι η βέλτιστη ισχύς μετάδοσης που μεγιστοποιεί το throughput είναι

$$p_k^* = \begin{cases} \frac{Q_k}{\lambda^*} - \frac{\sigma^2 |\mathbf{u}_k|^2}{|\mathbf{u}_k^H \mathbf{h}_k|^2}, & \lambda^* < \frac{Q_k |\mathbf{u}_k^H \mathbf{h}_k|^2}{\sigma^2 |\mathbf{u}_k|^2} \\ 0, & \lambda^* \geq \frac{Q_k |\mathbf{u}_k^H \mathbf{h}_k|^2}{\sigma^2 |\mathbf{u}_k|^2} \end{cases} \quad (2.3.17)$$

Η φυσική ερμηνεία του παραπάνω αποτελέσματος είναι ότι δίνουμε τόσο περισσότερο ισχύ σε έναν χρήστη, όσο λιγότερο θόρυβο δέχεται. Η διαδικασία αυτή θα πάψει όταν φτάσουμε στο σημείο όπου εξαντλείται η ισχύς. Το σημείο αυτό είναι

$$\sum_{k=1}^M \max \left\{ \frac{Q_k}{\lambda^*} - \frac{\sigma^2 |\mathbf{u}_k|^2}{|\mathbf{u}_k^H \mathbf{h}_k|^2}, 0 \right\} \quad (2.3.18)$$

Η παραπάνω διαδικασία ονομάζεται *αλγόριθμος Waterfilling*.

Τα παραπάνω αποτελέσματα μπορούν επίσης να εφαρμοστούν με κάποιες παραλλαγές στην περίπτωση του δυαδικού καθοδικού καναλιού, όπου ο σταθμός βάσης μεταδίδει δεδομένα σε  $K$  χρήστες. Το σενάριο από κοινού εκτέλεσης χωρικής επεξεργασίας σήματος με Decorrelator και ελέγχου ισχύος για το καθοδικό κανάλι προτείνεται ως θέμα για μελλοντική μελέτη και έρευνα.

### 2.3.2 Successive Interference Canceller

Θεωρούμε ότι στον σταθμό βάσης εκτελείται ιδανικό Successive Interference Cancellation με σειρά αποκωδικοποίησης  $\pi$ , όπου  $\pi = \pi_1, \dots, \pi_M$  είναι μια συγκεκριμένη μετάθεση του συνόλου των δεικτών των χρηστών  $\{1, \dots, M\}$  που έχουν προγραμματιστεί από τον χρονοπρογραμματιστή προς μετάδοση. Αυτό σημαίνει ότι ο χρήστης με ένδειξη  $\pi_1$  αποκωδικοποιείται πρώτος, ο χρήστης με ένδειξη  $\pi_2$  αποκωδικοποιείται δεύτερος, ... και ο χρήστης με ένδειξη  $\pi_M$  αποκωδικοποιείται τελευταίος.

Το βέλτιστο throughput  $C_{sum}$  ορίζεται, όπως έχουμε προαναφέρει, ως το μέγιστο σταθμισμένο άθροισμα των ρυθμών μετάδοσης επί τα μήκη των ουρών που μπορούν να επιτευχθούν από κοινού για απειροελάχιστα μικρό BER (bit-error-rate). Για σταθερή σειρά αποκωδικοποίησης  $\pi$ , το άθροισμα όλων των ρυθμών μετάδοσης των χρηστών μπορεί να γραφεί ως συνάρτηση της κατανομής ισχύος για το ανοδικό κανάλι:

$$f(\mathbf{p}) = \sum_{k=1}^M Q_k R_k(\mathbf{p}) = \sum_{k=1}^M Q_k \log(1 + SINR_k(\mathbf{p})). \quad (2.3.19)$$

Οπότε, το πρόβλημα μεγιστοποίησης του throughput ως προς  $\mathbf{p}$  ορίζεται ως εξής:

$$\begin{aligned} C_{sum} &= \max_{\mathbf{p}} f(\mathbf{p}), \\ \text{st. } \sum_{k=1}^M p_k &= P_{con}, \\ Q_k, p_k &\geq 0, \quad 1 \leq k \leq M. \end{aligned} \quad (2.3.20)$$

### Ανάλυση κυρτότητας

Θέλουμε να μελετήσουμε την κυρτότητα της αντικειμενικής συνάρτησης ως προς  $p_k$ . Γράφουμε την συνάρτηση του throughput  $f(\mathbf{p})$  που θέλουμε να μεγιστοποιήσουμε σε αναλυτική μορφή

$$\begin{aligned} f(\mathbf{p}) &= \sum_{k=1}^M Q_k R_k \\ &= \sum_{k=1}^M Q_k \log \det \left( I + \frac{1}{\sigma^2} \sum_{j=1}^k p_j \mathbf{h}_j \mathbf{h}_j^H \right) - Q_k \log \det \left( I + \frac{1}{\sigma^2} \sum_{j=1}^{k-1} p_j \mathbf{h}_j \mathbf{h}_j^H \right) \\ &\Rightarrow \sum_{k=1}^M Q_k R_k = \sum_{k=1}^M c_k \log \det \underbrace{\left( I + \frac{1}{\sigma^2} \sum_{j=1}^k p_j \mathbf{h}_j \mathbf{h}_j^H \right)}_{\Phi_k} \end{aligned} \quad (2.3.21)$$

όπου

$$c_k = \begin{cases} Q_k - Q_{k+1}, & 1 \leq k \leq M-1 \\ Q_M, & k = M \end{cases}, c_k \geq 0, \quad 1 \leq k \leq M. \quad (2.3.22)$$

Παρατηρούμε ότι ο όρος  $\Phi_k$  στην εξίσωση (2.3.21) είναι μια κοίλη συνάρτηση ως προς την κατανομή ισχύος  $\pi$ , για κάθε  $1 \leq k \leq M$ . Όμως, γνωρίζουμε ότι το άθροισμα κοίλων συναρτήσεων είναι επίσης κοίλο. Επομένως, η συνάρτηση  $f(\mathbf{p})$  είναι κοίλη ως προς  $p_k$ . Από τη θεωρία βελτιστοποίησης, γνωρίζουμε ότι αυτό σημαίνει πως η συνάρτηση έχει **μοναδικό ολικό βέλτιστο**.

### Αναζήτηση ολικού βέλτιστου

Όπως αναφέραμε το πρόβλημα μεγιστοποίησης του throughput ως προς  $\mathbf{p}$  ορίζεται ως εξής:

$$\begin{aligned} C_{sum} &= \max_{\mathbf{p}} f(\mathbf{p}), \\ \text{st. } \sum_{k=1}^M p_k &= P_{con}, \\ Q_k, p_k &\geq 0, \quad 1 \leq k \leq M. \end{aligned} \quad (2.3.23)$$

Η Lagrangian του προβλήματος κατανομής ισχύος (2.3.20) είναι

$$L(\mathbf{p}, \mu, \lambda) = - \sum_{k=1}^M c_k \log \det \left( I + \frac{1}{\sigma^2} \sum_{j=1}^M p_j \mathbf{h}_j \mathbf{h}_j^H \right) + \lambda \left( \sum_{k=1}^M p_k - P_{con} \right) - \sum_{k=1}^M \mu_k p_k. \quad (2.3.24)$$

Οι συνθήκες Karush-Kuhn-Tucker (KKT) για το παραπάνω πρόβλημα βελτιστοποίησης είναι ικανές και αναγκαίες ώστε να είναι η κατανομή ισχύος  $\mathbf{p}^*$  βέλτιστη (βλ. [20]). Από την (2.3.24), προκύπτουν οι ακόλουθες συνθήκες KKT για  $1 \leq k \leq M$

$$\frac{1}{\sigma^2} c_k \mathbf{h}_k \left( I + \frac{1}{\sigma^2} \sum_{j=1}^M p_j^* \mathbf{h}_j \mathbf{h}_j^H \right)^{-1} \mathbf{h}_k = \lambda^* - \mu_k^*, \quad (2.3.25)$$

$$\mu_k^* p_k^* = 0, \quad (2.3.26)$$

$$\mu_k^* \geq 0, \quad (2.3.27)$$

$$\lambda \geq 0 \quad (2.3.28)$$

$$\sum_{k=1}^M p_k^* = P_{con}, \quad (2.3.29)$$

$$p_k^* \geq 0. \quad (2.3.30)$$

Χάρην ευκολίας, ορίζουμε τους συντελεστές  $a_k$

$$a_k(\mathbf{p}) = \frac{1}{\sigma^2} c_k \mathbf{h}_k \left( I + \frac{1}{\sigma^2} \sum_{j=1}^M p_j \mathbf{h}_j \mathbf{h}_j^H \right)^{-1} \mathbf{h}_k. \quad (2.3.31)$$

Επιλύουμε την εξίσωση (2.3.25) ως προς  $\mu_k^*$  και αντικαθιστούμε στην εξίσωση (2.3.26)

$$p_k^* (\lambda - a_k) = 0 \quad (2.3.32)$$

Διακρίνουμε περιπτώσεις:

1. αν  $p_k^* > 0$ , τότε  $a_k = \lambda^*$  ή

2. αν  $p_k^* = 0$ , τότε  $a_k < \lambda^*$

Από τα παραπάνω, προκύπτει ότι για να είναι η κατανομή ισχύος  $\mathbf{p}^*$  βέλτιστη πρέπει να ισχύει

$$\{k_1, k_2 \in S_p(\mathbf{p}) \Rightarrow a_{k_1} = a_{k_2} \quad (2.3.33)$$

$$j \notin S_p(\mathbf{p}) \Leftrightarrow a_j \leq \max_{k \in S_p(\mathbf{p})} a_k \}. \quad (2.3.34)$$

Οι δυο παραπάνω εξισώσεις δηλώνουν ότι όλοι οι χρήστες  $k$  που αποκτούν ισχύ μετάδοσης  $p_k > 0$  έχουν το ίδιο  $a_k = \lambda^*$ . Επιπλέον, όλα τα υπόλοιπα  $a_j$  είναι ίσα ή μικρότερα ή μικρότερα του  $a_k$ .

Η εύρεση αναλυτικής έκφρασης για την βέλτιστη κατανομή ισχύος προτείνεται ως θέμα για μελλοντική μελέτη και έρευνα.

## Κεφάλαιο 3

# Χρονοπρογραμματισμός

Στο παρόν κεφάλαιο θα μελετήσουμε τεχνικές χρονοπρογραμματισμού για το κανάλι που περιγράψαμε στο (1.4). Ξαναγράφουμε το πρόβλημα (1.6.7) που επιθυμούμε να λύσουμε με βέλτιστο χρονοπρογραμματισμό

$$\begin{aligned} I(t) &= \arg \max_{I \in \mathcal{I}} \sum_{k=1}^K Q_k(t) R_k(I, t) \\ \text{st. } \quad &\sum_{k=1}^K p_k = P_{con} \\ &p_k, Q_k \geq 0, \quad k = 1, 2, \dots, K. \end{aligned} \tag{3.0.1}$$

όπου  $I(t)$  είναι μια πολιτική χρονοπρογραμματισμού.

Στην περίπτωση του Decorrelating Detector, μια πολιτική χρονοπρογραμματισμού  $I_d(t)$  είναι μια γενική αντιστοίχιση από το Καρτεσιανό γινόμενο του χώρου των  $K$  διανυσμάτων καναλιού  $\mathbf{h}_k \in \mathbf{C}^{M \times 1}$  και του διανύσματος μήκους ουρών  $\mathbf{Q} \in \mathbf{P}_+^K$  στο Καρτεσιανό γινόμενο του χώρου των διανυσμάτων μορφοποίησης δέσμης  $\mathbf{u}_k \in \mathbf{C}^{M \times 1}$  και του χώρου των συνδυασμών των  $K$  ανά  $M$ ,  $Comb(K, M)$

$$I_d : (\times_{k=1}^K \mathbf{C}^{M \times 1}) \times \mathbf{P}_+^K \rightarrow (\times_{k=1}^K \mathbf{C}^{M \times 1}) \times Comb(K, M). \tag{3.0.2}$$

Στην περίπτωση του Successive Interference Cancellor, μια πολιτική χρονοπρογραμματισμού  $I_s(t)$  είναι μια γενική αντιστοίχιση από το Καρτεσιανό γινόμενο του χώρου των  $K$  διανυσμάτων καναλιού  $\mathbf{h}_k \in \mathbf{C}^{M \times 1}$  και του διανύσματος μήκους ουρών  $\mathbf{X} \in \mathbf{P}_+^K$  στο Καρτεσιανό γινόμενο του χώρου των διανυσμάτων μορφοποίησης δέσμης  $\mathbf{u} \in \mathbf{C}^{M \times 1}$ , του χώρου των συνδιασμών των  $K$  ανά  $M$ ,  $Comb(K, M)$  και του χώρου των  $M!$  μεταθέσεων των στοιχείων κάθε συνδιασμού  $K$  ανά  $M$ ,  $\Pi_M$

$$I_s : (\times_{k=1}^K \mathbf{C}^{M \times 1}) \times \mathbf{R}_+^K \rightarrow (\times_{k=1}^K \mathbf{C}^{M \times 1}) \times \text{Comb}(K, M) \times \Pi_M. \quad (3.0.3)$$

Αρχικά, θα προσδιορίσουμε το μέγεθος του υποσυνόλου των χρηστών που μπορούν να προγραμματιστούν προς μετάδοση σε μια χρονική στιγμή  $t$ , με βάση την ιδιαίτερη δομή του κάθε δέκτη. Ειδικά στην περίπτωση του Successive Interference Canceller πρέπει να προσδιορίσουμε και την σειρά αποκωδικοποίησης των χρηστών. Έπειτα, θα μελετήσουμε κάποιες τεχνικές χρονοπρογραμματισμού, θα παρουσιάσουμε αναλυτικά τη λειτουργία τους και θα αναλύσουμε τα πλεονεκτήματα και τα μειονεκτήματα τους. Πειραματική σύγκριση της απόδοσης όλων των τεχνικών πραγματοποιείται στο Κεφάλαιο 9.

### 3.1 Βέλτιστος χρονοπρογραμματισμός

Η βέλτιστη μέθοδος χρονοπρογραμματισμού είναι αυτή που υπολογίζει το throughput για κάθε δυνατό υποσύνολο (και μετάθεση) χρηστών και επιλέγει το υποσύνολο εκείνο που επιτυγχάνει μέγιστο throughput

$$C_{sum} = \max \sum_k Q_k R_k. \quad (3.1.1)$$

#### 3.1.1 Πλήθος δυνατών υποσυνόλων

Η λειτουργία κάθε δέκτη επηρεάζεται σε σημαντικό βαθμό και από το πλήθος σημάτων προς αποκωδικοποίηση. Σε αυτή την εργασία μελετούμε το σενάριο χρονοπρογραμματισμού ενός υποσυνόλου χρηστών μεγέθους  $M$ , όσο δηλαδή και το πλήθος ενεργών στοιχείων πολλαπλής κεραίας του στήθου βάσης. Όπως αναφέραμε και στο κεφάλαιο (2), το σύνολο των χρηστών που υπάρχουν στο σύστημα είναι  $\{1, 2, \dots, K\}$ . Συνεπώς, το πλήθος όλων των δυνατών υποσυνόλων χρηστών μεγέθους  $M$  ισούται με τον αριθμό όλων των δυνατών συνδιασμών των  $K$  ανά  $M$

$$\text{number of subsets} = \binom{K}{M} = \frac{K!}{M!(K-M)!} \quad (3.1.2)$$

### 3.1.2 Πλήθος δυνατών υποσυνόλων και μεταθέσεων

Ειδικά στην περίπτωση του Successive Interference Canceller σημαντικό ρόλο παίζει και η σειρά αποκωδικοποίησης (decoding order) των χρηστών. Αυτό σημαίνει πως τα δυνατά υποσύνολα χρηστών μεγέθους  $M$  δεν είναι μόνο οι συνδιασμοί των  $K$  ανά  $M$ . Για κάθε έναν από αυτούς τους συνδυασμούς - υποσύνολα πρέπει να ελέγξουμε και κάθε δυνατή μετάθεση των χρηστών που ανήκουν στον συνδυασμό. Αυτό σημαίνει πως για κάθε συνδυασμό - υποσύνολο μεγέθους  $M$  υπάρχουν  $M!$  μεταθέσεις. Οπότε ο αριθμός όλων των δυνατών υποσυνόλων που πρέπει να ελεγχθούν ισούται με τον αριθμό των συνδιασμών των  $K$  ανά  $M$  επί τον αριθμό όλων των δυνατών μεταθέσεων των στοιχείων ενός συνόλου  $M$

$$\text{number of subsets} = \binom{K}{M} M! = \frac{K!}{M!(K-M)!} M! = \frac{K!}{(K-M)!} \quad (3.1.3)$$

### 3.1.3 Βέλτιστη σειρά αποκωδικοποίησης

**maxqueue:** Στην εργασία [8] προτάθηκε μια πολιτική ταξινόμησης χρηστών και αποδείχτηκε πως αυτή η πολιτική επιτυγχάνει βέλτιστο throughput. Σύμφωνα με την πολιτική maxqueue, η βέλτιστη σειρά αποκωδικοποίησης  $\pi$  που μεγιστοποιεί το  $\sum_{k=1}^M Q_k R_k$  είναι

$$Q_{\pi_M} \geq Q_{\pi_{M-1}} \geq \dots \geq Q_{\pi_1} > 0, \quad (3.1.4)$$

όπου  $Q_{\pi_k}$  είναι το μήκος ουράς του  $k$ -οστού χρήστη που αποκωδικοποιείται. Ουσιαστικά, η πολιτική ταξινόμησης maxqueue λέει ότι πρώτος αποκωδικοποιείται ο χρήστης με τη μικρότερη ουρά, δεύτερος ο χρήστης με την αμέσως μεγαλύτερη ουρά, ... και τελευταίος αποκωδικοποιείται ο χρήστης με τη μεγαλύτερη ουρά. Η πολιτική maxqueue που μόλις περιγράψαμε είναι η πολιτική που χρησιμοποιείται στις ευριστικές μεθόδους χρονοπρογραμματισμού που θα αναλύσουμε παρακάτω.



### 3.1.4 Βέλτιστη πολιτική χρονοπρογραμματισμού

**optSched:** Σε κάθε χρονική στιγμή  $t$ , υπολογίζεται το σταθμισμένο άθροισμα ρυθμών μετάδοσης με βάρη τα μήκη ουράς  $Q_k$ , για κάθε δυνατό υποσύνολο χρηστών μεγέθους  $M$ . Η πολιτική αυτή αναζητά το υποσύνολο χρηστών που επιτυγχάνει μέγιστο throughput.

1. Υπολογισμός των διανυσμάτων ρυθμών μετάδοσης  $\mathbf{R}_k(t)$  και του σταθμισμένου αθροίσματος ρυθμών μετάδοσης  $W_k(t) = \sum_{i=1}^M Q_i(t)R_i(t) : k \in \{1, \dots, N\}, N = \text{comb}(K, M) \times \text{perm}(M)^1$  για κάθε δυνατό υποσύνολο χρηστών μεγέθους  $M$ .
2. Επιλογή υποσυνόλου  $S_{M,\pi}^*(t)$  που επιτυγχάνει μέγιστο throughput

$$C_{sum} = \max_k W_k^*. \quad (3.1.5)$$

3. Μετάδοση των χρηστών που ανήκουν στο υποσύνολο  $S_{M,\pi}^*(t)$  με τους αντίστοιχους ρυθμούς μετάδοσης  $\mathbf{R}^*(t)$  - ο ρυθμός μετάδοσης όλων των υπόλοιπων χρηστών που δεν ανήκουν στο  $S_{M,\pi}^*(t)$  τίθεται ίσος με 0.
4. Αποκωδικοποίηση των χρηστών με σειρά  $\pi^*$  <sup>2</sup>.

Η πολιτική optSched επιτυγχάνει τους καλύτερους ρυθμούς μετάδοσης και τις μικρότερες καθυστερήσεις. Ωστόσο, η πολιτική αυτή είναι ακριβή υπολογιστικά σε χώρο και χρόνο.

---

<sup>1</sup>Μόνο για την περίπτωση του SIC.

<sup>2</sup>Μόνο για την περίπτωση του SIC.

### 3.2 Χρονοπρογραμματισμός βασιζόμενος στο μήκος ουράς

Η πολιτική `longestQueue` που παρουσιάζεται παρακάτω αποτελεί μια ευριστική μέθοδο χρονοπρογραμματισμού που βασίζεται στην επιλογή του υποσυνόλου μεγέθους  $M$  με κριτήριο τα μήκη ουράς των χρηστών.

**longestQueue:** Σε κάθε χρονική στιγμή  $t$ , επιλέγονται από τον χρονοπρογραμματιστή οι  $M$  από τους  $K$  χρήστες με τα μεγαλύτερα μήκη ουράς  $Q_k$ . Παρακάτω φαίνεται ο αλγόριθμος που εφαρμόζεται στον σταθμό βάσης.

1. Επιλογή των  $M$  χρηστών με τα μεγαλύτερα μήκη ουρών από τους  $K$ , το υποσύνολο αυτό δηλώνεται ως  $S_M^{Q^{max}}(t)$ .
2. Διάταξη των  $M$  χρηστών σε σειρά αυξανόμενου μήκους ουράς σύμφωνα με την `maxqueue`<sup>3</sup>, δηλ.

$$Q_M \geq Q_{M-1} \geq \dots \geq Q_1 > 0. \quad (3.2.1)$$

3. Υπολογισμός του διανύσματος ρυθμών μετάδοσης  $\mathbf{R}^{Q^{max}}(t)$  για την επιλεγμένη σειρά αποκωδικοποίησης  $\pi^*$  του επιλεγμένου υποσυνόλου χρηστών  $S_M^{Q^{max}}(t)$ .
4. Μετάδοση των χρηστών του επιλεγμένου υποσυνόλου  $S_M^{Q^{max}}(t)$  με ρυθμούς μετάδοσης  $\mathbf{R}^{Q^{max}}(t)$  - ο ρυθμός μετάδοσης όλων των υπόλοιπων χρηστών που δεν ανήκουν στο  $S_M^{Q^{max}}(t)$  τίθεται ίσος με 0.
5. Αποκωδικοποίηση των χρηστών με σειρά  $\pi^*$  <sup>4</sup>.

Η πολιτική `longestQueue` επιτυγχάνει χαμηλότερους ρυθμούς μετάδοσης και υψηλότερες καθυστερήσεις σε σύγκριση με την βέλτιστη πολιτική `optSched`. Ωστόσο, η πολιτική `longestQueue` παρουσιάζει πολύ μικρότερο υπολογιστικό κόστος από την βέλτιστη `optSched`, ανεξαρτήτως του μεγέθους των  $K$  και  $M$ .

<sup>3</sup>Μόνο για την περίπτωση του SIC.

<sup>4</sup>Μόνο για την περίπτωση του SIC.

### 3.3 Χρονοπρογραμματισμός βασιζόμενος στη νόρμα διανύσματος καναλιού

Η πολιτική `bestChannel` που παρουσιάζεται παρακάτω αποτελεί μια ευριστική μέθοδο χρονοπρογραμματισμού που βασίζεται στην επιλογή του υποσυνόλου μεγέθους  $M$  με κριτήριο το διάνυσμα καναλιού κάθε χρήστη  $|\mathbf{h}_k|$ .

**bestChannel:** Σε κάθε χρονική στιγμή  $t$ , επιλέγονται από τον χρονοπρογραμματιστή οι  $M$  από τους  $K$  χρήστες με τη μεγαλύτερη νόρμα καναλιού. Παρακάτω φαίνεται ο αλγόριθμος που εφαρμόζεται στον σταθμό βάσης.

1. Επιλογή των  $M$  χρηστών με τη μεγαλύτερη νόρμα καναλιού από τους  $K$ , το υποσύνολο αυτό δηλώνεται ως  $S_M^{|\mathbf{h}|max}(t)$ .
2. Διάταξη των  $M$  χρηστών σε σειρά αυξανόμενου μήκους ουράς σύμφωνα με την `maxqueue`<sup>5</sup>.
3. Υπολογισμός του διανύσματος ρυθμών μετάδοσης  $\mathbf{R}^{|\mathbf{h}|max}(t)$  για την επιλεγμένη σειρά αποκωδικοποίησης  $\pi^*$  του επιλεγμένου υποσυνόλου χρηστών  $S_M^{|\mathbf{h}|max}(t)$ .
4. Μετάδοση των χρηστών του επιλεγμένου υποσυνόλου  $S_M^{|\mathbf{h}|max}(t)$  με ρυθμούς μετάδοσης  $\mathbf{R}^{|\mathbf{h}|max}(t)$  - ο ρυθμός μετάδοσης όλων των υπόλοιπων χρηστών που δεν ανήκουν στο  $S_M^{|\mathbf{h}|max}(t)$  τίθεται ίσος με 0.
5. Αποκωδικοποίηση των χρηστών με σειρά  $\pi^*$  <sup>6</sup>.

Η πολιτική `bestChannel` επιτυγχάνει χαμηλότερους ρυθμούς μετάδοσης και υψηλότερες καθυστερήσεις σε σύγκριση με την βέλτιστη πολιτική `optSched`. Ωστόσο, η πολιτική `bestChannel` παρουσιάζει πολύ μικρότερο υπολογιστικό κόστος από την βέλτιστη `optSched`, ανεξαρτήτως του μεγέθους των  $K$  και  $M$ .

---

<sup>5</sup>Μόνο για την περίπτωση του SIC.

<sup>6</sup>Μόνο για την περίπτωση του SIC.

### 3.4 Χρονοπρογραμματισμός βασιζόμενος στο μήκος ουράς και στη νόρμα διανύσματος καναλιού

Η πολιτική `bestQueueChannel` που παρουσιάζεται παρακάτω αποτελεί μια ευριστική μέθοδο χρονοπρογραμματισμού που βασίζεται στην επιλογή του υποσυνόλου μεγέθους  $M$  με κριτήριο το γινόμενο του μήκους ουράς  $Q_k$  επί το διάνυσμα καναλιού κάθε χρήστη  $|\mathbf{h}_k|$ .

**bestQueueChannel:** Σε κάθε χρονική στιγμή  $t$ , επιλέγονται από τον χρονοπρογραμματιστή οι  $M$  από τους  $K$  χρήστες με το μεγαλύτερο γινόμενο μήκους ουράς  $Q_k$  επί τη νόρμα διανύσματος καναλιού  $|\mathbf{h}_k|$ . Παρακάτω φαίνεται ο αλγόριθμος που εφαρμόζεται στον σταθμό βάσης.

1. Επιλογή των  $M$  χρηστών με το μεγαλύτερο γινόμενο μήκους ουράς  $Q_k(t)$  επί τη νόρμα διανύσματος καναλιού  $|\mathbf{h}_k|(t)$  από τους  $K$ , το υποσύνολο αυτό δηλώνεται ως  $S_M^{\{Q_k|\mathbf{h}|\}_{max}}(t)$ .
2. Διάταξη των  $M$  χρηστών σε σειρά αυξανόμενου μήκους ουράς σύμφωνα με την `maxqueue`<sup>7</sup>.
3. Υπολογισμός του διανύσματος ρυθμών μετάδοσης  $\mathbf{R}^{\{Q_k|\mathbf{h}|\}_{max}}(t)$  για το επιλεγμένο υποσύνολο χρηστών  $S_M^{\{Q_k|\mathbf{h}|\}_{max}}(t)$ .
4. Μετάδοση των χρηστών του επιλεγμένου υποσυνόλου  $S_M^{\{Q_k|\mathbf{h}|\}_{max}}(t)$  με ρυθμούς μετάδοσης  $\mathbf{R}^{\{Q_k|\mathbf{h}|\}_{max}}(t)$  - ο ρυθμός μετάδοσης όλων των υπόλοιπων χρηστών που δεν ανήκουν στο  $S_M^{\{Q_k|\mathbf{h}|\}_{max}}(t)$  τίθεται ίσος με 0.
5. Αποκωδικοποίηση των χρηστών με σειρά  $\pi^*$ <sup>8</sup>.

Η πολιτική `bestQueueChannel` επιτυγχάνει χαμηλότερους ρυθμούς μετάδοσης και υψηλότερες καθυστερήσεις σε σύγκριση με την βέλτιστη πολιτική `optSched`. Ωστόσο, η πολιτική `bestQueueChannel` παρουσιάζει πολύ μικρότερο υπολογιστικό κόστος από την βέλτιστη `optSched`, ανεξαρτήτως του μεγέθους των  $K$  και  $M$ .

<sup>7</sup>Μόνο για την περίπτωση του SIC.

<sup>8</sup>Μόνο για την περίπτωση του SIC.

### 3.5 Τυχαίος χρονοπρογραμματισμός

Η πολιτική random που παρουσιάζεται παρακάτω αποτελεί μια ευριστική μέθοδο χρονοπρογραμματισμού που βασίζεται στην τυχαία επιλογή του υποσυνόλου μεγέθους  $M$ .

**random:** Σε κάθε χρονική στιγμή  $t$ , ο χρονοπρογραμματιστής επιλέγει ένα υποσύνολο χρηστών μεγέθους  $M$ , από το σύνολο όλων των δυνατών υποσυνόλων του  $\{1, \dots, K\}$  μεγέθους  $M$ , με τυχαίο τρόπο. Παρακάτω περιγράφεται αναλυτικά ο αλγόριθμος που εφαρμόζεται στον σταθμό βάσης.

1. Αντιστοίχιση ενός τυχαίου αριθμού - δείκτη  $r_k : k \in \{1, \dots, N\}, N = comb(K, M) \times perm(M)$  σε κάθε δυνατό υποσύνολο μεγέθους  $M$ . Οι τυχαίοι αριθμοί ακολουθούν την κανονική κατανομή  $N(0, 1)$ .
2. Επιλογή του υποσυνόλου  $S_M^*(t)$  στο οποίο αντιστοιχεί ο μέγιστος δείκτης  $r^* = \max\{r_1, \dots, r_N\}$ .
3. Διάταξη των  $M$  χρηστών σε σειρά αυξανόμενου μήκους ουράς σύμφωνα με την *maxqueue*<sup>9</sup>.
4. Υπολογισμός του διανύσματος ρυθμών μετάδοσης  $\mathbf{R}^*(t)$  για το υποσύνολο  $S_M^*(t)$  που επιλέχθηκε.
5. Μετάδοση των χρηστών που ανήκουν στο υποσύνολο  $S_M^*(t)$  με τους αντίστοιχους ρυθμούς μετάδοσης  $\mathbf{R}^*(t)$  - ο ρυθμός μετάδοσης όλων των υπόλοιπων χρηστών που δεν ανήκουν στο  $S_M^*(t)$  τίθεται ίσος με 0.
6. Αποκωδικοποίηση των χρηστών με σειρά  $\pi^*$ <sup>10</sup>.

Η πολιτική random επιτυγχάνει χαμηλότερους ρυθμούς μετάδοσης και υψηλότερες καθυστερήσεις σε σύγκριση με την βέλτιστη πολιτική optSched. Ωστόσο, η πολιτική random παρουσιάζει πολύ μικρότερο υπολογιστικό κόστος από την βέλτιστη optSched, ανεξαρτήτως του μεγέθους των  $K$  και  $M$ .

<sup>9</sup>Μόνο για την περίπτωση του SIC.

<sup>10</sup>Μόνο για την περίπτωση του SIC.

### 3.6 Χωρίς χρονοπρογραμματισμό

Η πολιτική noSched που παρουσιάζεται παρακάτω δεν αποτελεί μέθοδο χρονοπρογραμματισμού. Ουσιαστικά, πρόκειται για απουσία χρονοπρογραμματισμού και μετάδοση όλων των  $K > M$  χρηστών.

**noSched:** Σε κάθε χρονική στιγμή  $t$ , μεταδίδονται κανονικά όλοι οι  $K > M$  χρήστες. Η σειρά αποκωδικοποίησης των χρηστών ορίζεται σύμφωνα την πολιτική maxqueue.

1. Διάταξη των  $K$  χρηστών σε σειρά αυξανόμενου μήκους ουράς σύμφωνα με την maxqueue<sup>11</sup>.
2. Υπολογισμός του διανύσματος ρυθμών μετάδοσης  $\mathbf{R}(t)$  για το σύνολο χρηστών  $\{1, \dots, K\}$ .
3. Μετάδοση όλων των χρηστών με τους αντίστοιχους ρυθμούς μετάδοσης  $\mathbf{R}(t)$ .
4. Αποκωδικοποίηση των χρηστών με σειρά  $\pi^*$ <sup>12</sup>.

Η πολιτική noSched επιτυγχάνει τους χειρότερους ρυθμούς μετάδοσης και τις χειρότερες καθυστερήσεις σε σύγκριση με όλες τις πολιτικές που προαναφέρθηκαν. Φυσικά, η πολιτική noSched παρουσιάζει το μικρότερο υπολογιστικό κόστος.

---

<sup>11</sup>Μόνο για την περίπτωση του SIC.

<sup>12</sup>Μόνο για την περίπτωση του SIC.



## Κεφάλαιο 4

# Προσομοίωση

Στο παρόν κεφάλαιο παρουσιάζουμε την διαδικασία που ακολουθήθηκε στην προσομοίωση και τα αποτελέσματα που παρήχθησαν. Η προσομοίωση πραγματοποιήθηκε σε περιβάλλον Matlab. Παρακάτω περιγράφεται το μοντέλο προσομοίωσης και το κριτήριο απόδοσης που χρησιμοποιήθηκε και παρουσιάζονται αναλυτικά τα αποτελέσματα.

### 4.1 Μοντέλο προσομοίωσης

Θεωρήσαμε ένα ανοδικό κανάλι με  $K$  ασύρματους χρήστες με απλές κεραιές και έναν σταθμό βάσης εξοπλισμένο με πολλαπλή κεραία  $M$  ενεργών στοιχείων. Όπως αναφέραμε και στο (2), το κανάλι  $H$  ακολουθεί την κατανομή Rayleigh και μεταβάλλεται ανεξάρτητα σε κάθε χρονική στιγμή. Η διάρκεια προσομοίωσης ήταν  $d = 20000$  χρονικές στιγμές.

Όπως αναφέραμε και στο κεφάλαιο (1), η αναδρομική σχέση που αναπαριστά την εξέλιξη του μήκους ουράς ενός χρήστη  $k$  στο χρόνο, είναι:

$$Q_k(t+1) = Q_k(t) - R_k(t+1) + A_k(t+1), \quad (4.1.1)$$

όπου  $A_k(t)$  είναι το διάνυσμα αφίξεων και  $R_k(t)$  το διάνυσμα ρυθμών μετάδοσης. Το  $\mathbf{A}(t)$  δίνεται από τη σχέση:

$$\mathbf{A}(t) = \alpha \mathbf{L}(t), \quad (4.1.2)$$

όπου  $\alpha$  είναι μια μεταβλητή ελέγχου, μέσω της οποίας ελέγχουμε το ποσό της κίνησης που εισέρχεται στο σύστημα και  $\mathbf{L}(t)$  είναι ένα σταθερό διάνυσμα, του οποίου οι συνιστώσες είναι είτε 0, είτε 1 σύμφωνα με μια Bernoulli κατανομή.



## 4.2 Κριτήριο απόδοσης

Το κριτήριο απόδοσης που χρησιμοποιείται στην προσομοίωση είναι η **μέση καθυστέρηση πακέτου** (average packet delay) προς το **συνολικό throughput** του συστήματος.

Όπως είπαμε ο συνολικός αριθμός πακέτων που περιμένουν στις ουρές των χρηστών προς εξυπηρέτηση, τη χρονική στιγμή  $t$ , δίνεται από τον τύπο

$$\mathbf{Q}(t+1) = [\mathbf{Q}(t) - \mathbf{R}(t+1)]^+ + \mathbf{A}(t+1), \quad (4.2.1)$$

όπου  $\mathbf{A}(t+1)$  είναι το διάνυσμα αφίξεων (πακέτα που φθάνουν προς εξυπηρέτηση) τη χρονική στιγμή  $t+1$  και  $\mathbf{R}(t+1)$  είναι το διάνυσμα ρυθμών μετάδοσης (πακέτα που μεταδίδονται επιτυχώς) τη χρονική στιγμή  $t+1$ .

Η μέση τιμή του  $\mathbf{Q}(t)$  είναι

$$\mathbf{Q}_\infty = \lim_{t \rightarrow \infty} \mathbf{Q}(t), \quad (4.2.2)$$

που κατά προσέγγιση ισούται με

$$\mathbf{Q}_\infty = \frac{1}{t} \sum_{i=1}^t \mathbf{Q}(i). \quad (4.2.3)$$

Από την τελευταία εξίσωση προκύπτει

$$\mathbf{Q}_\infty = \frac{\sum_{i=1}^t \mathbf{A}(i) \sum_{i=1}^t \mathbf{Q}(i)}{\sum_{i=1}^t \mathbf{A}(i)}. \quad (4.2.4)$$

Έστω  $E[T]$  η μέση καθυστέρηση πακέτου και  $\lambda = \frac{1}{n} \sum_{i=1}^t \mathbf{A}(i)$  ο συνολικός ρυθμός αφίξεων (packets/slot) για όλους τους χρήστες. Ισχύει

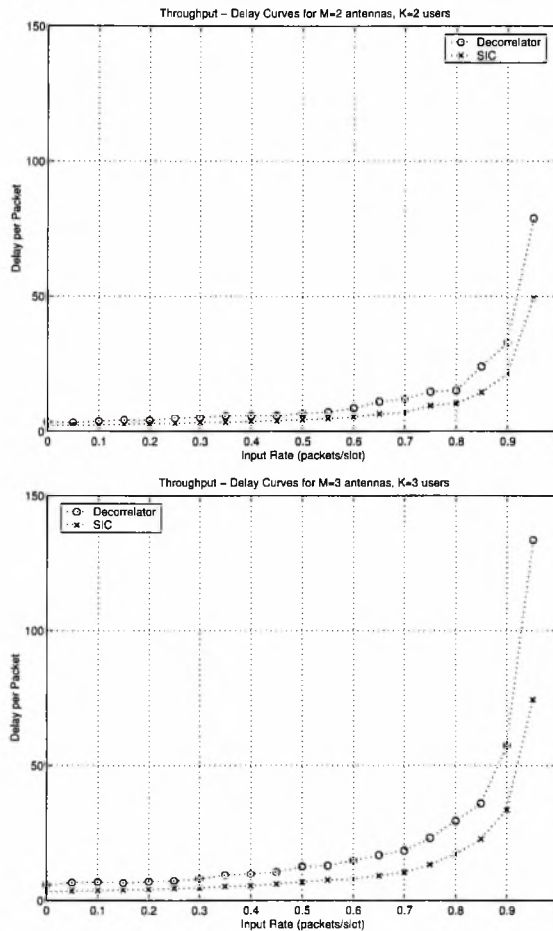
$$\mathbf{Q}_\infty = \lambda E[T]. \quad (4.2.5)$$

Επομένως, από τις εξισώσεις (4.2.4) και (4.2.5) προκύπτει ότι η μέση καθυστέρηση πακέτου είναι

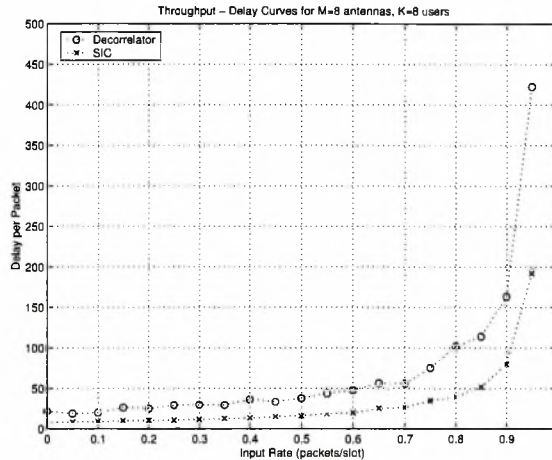
$$E[T] = \frac{\sum_{i=1}^t \mathbf{Q}(i)}{\sum_{i=1}^t \mathbf{A}(i)} \quad (4.2.6)$$

### 4.3 Αποτελέσματα

Αρχικά, στα σχήματα (4.1) και (4.2) συγκρίνουμε τον Decorrelator με τον Successive Interference Canceller, για  $2 \times 2$ , για  $3 \times 3$  και  $8 \times 8$  κανάλι αντίστοιχα. Υπενθυμίζουμε ότι  $M$  είναι ο αριθμός ενεργών στοιχείων στην πολλαπλή κεραία του σταθμού βάσης και  $K$  είναι ο αριθμός ασύρματων χρηστών. Όπως είναι αναμενόμενο ο Successive Interference Canceller παρουσιάζει εμφανώς καλύτερη συμπεριφορά από τον Decorrelator.



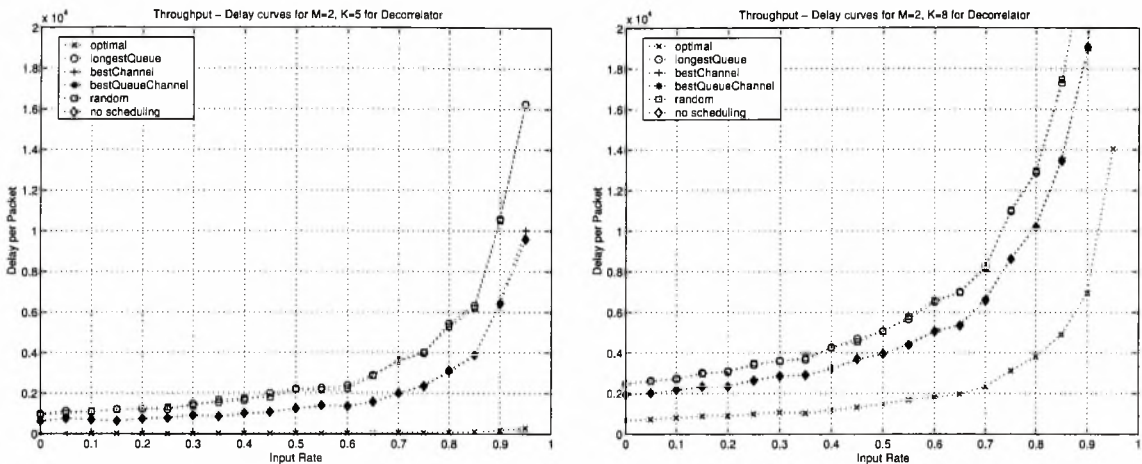
Σχήμα 4.1: Συγκριτικές καμπύλες throughput - delay για Decorrelator και Successive Interference Canceller σε  $2 \times 2$  και  $3 \times 3$  κανάλι αντίστοιχα



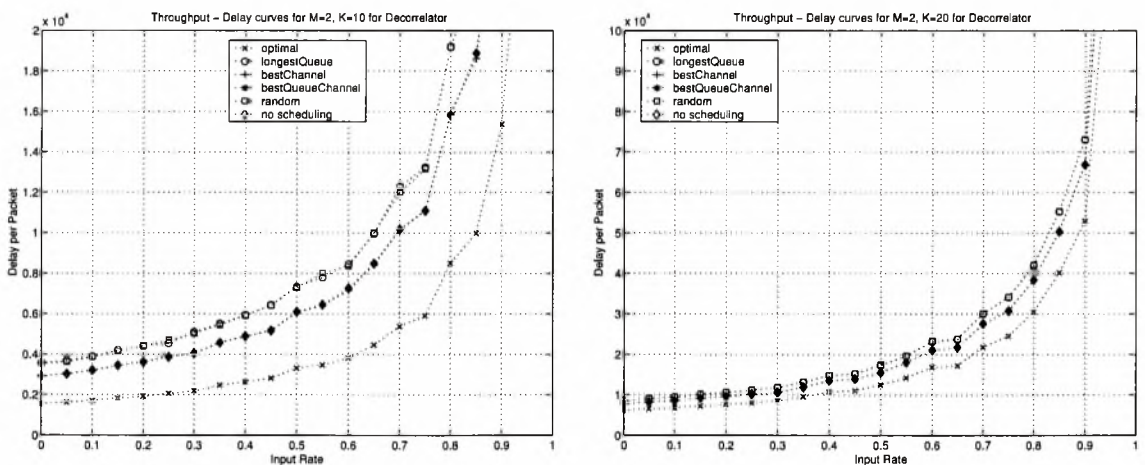
Σχήμα 4.2: Συγκριτικές καμπύλες throughput - delay για Decorrelator και Successive Interference Cancellor σε  $8 \times 8$  κανάλι

Στη συνέχεια, στα σχήματα (4.3) και (4.4) συγκρίνουμε την απόδοση των έξι πολιτικών χρονοπρογραμματισμού που παρουσιάστηκαν στο (3) για  $5 \times 2$ , για  $8 \times 2$ , για  $10 \times 2$  και τέλος,  $20 \times 2$  κανάλι αντίστοιχα, στην περίπτωση που στον σταθμό βάσης εκτελείται Decorrelator.

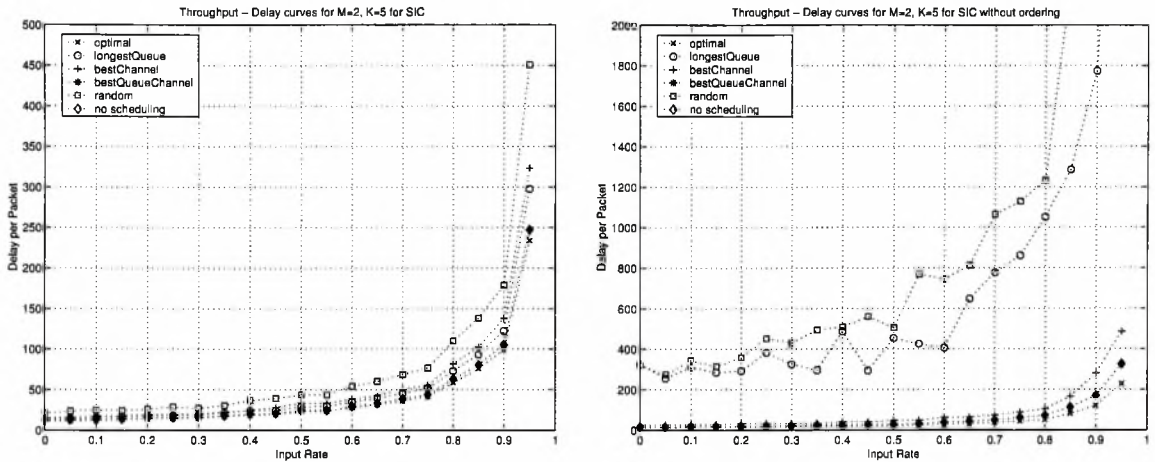
Έπειτα, στα σχήματα (4.5), (4.6), (4.7) και (4.8) συγκρίνουμε την απόδοση των έξι πολιτικών χρονοπρογραμματισμού που παρουσιάστηκαν στο (3) για  $5 \times 2$ , για  $8 \times 2$ , για  $10 \times 2$  και τέλος,  $20 \times 2$  κανάλι αντίστοιχα, στην περίπτωση που στον σταθμό βάσης εκτελείται Successive Interference Cancellor. Αριστερά φαίνονται τα γραφήματα για την περίπτωση που οι πολιτικές longestQueue, bestChannel, bestQueueChannel, random και noSched χρησιμοποιούν την πολιτική ταξινόμησης maxqueue, ενώ δεξιά φαίνονται τα γραφήματα για την περίπτωση που δεν τη χρησιμοποιούν.



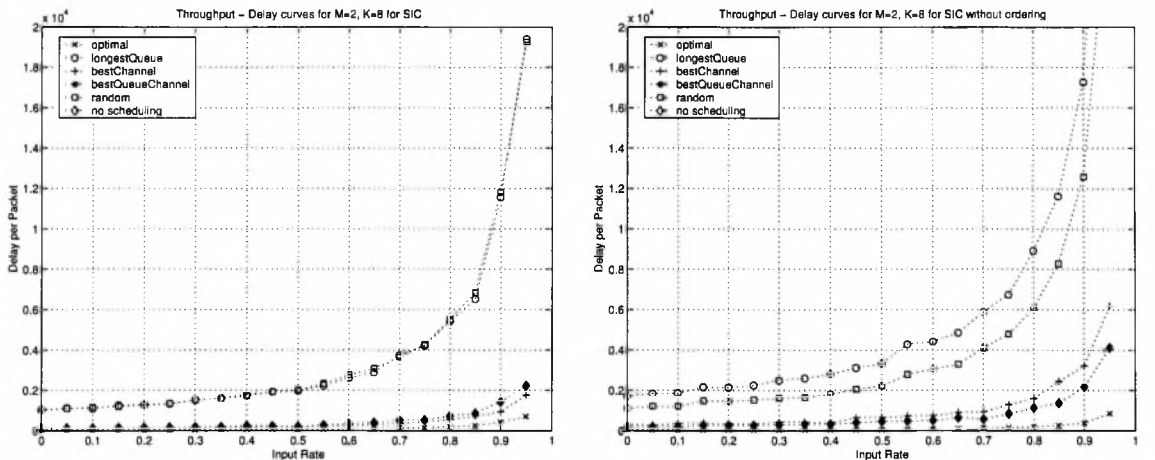
Σχήμα 4.3: Συγκριτικές καμπύλες throughput - delay για τις πολιτικές χρονοπρογραμματισμού optSched, longestQueue, bestChannel, bestQueueChannel, random και noSched σε  $5 \times 2$  και  $8 \times 2$  κανάλι, όπου εκτελείται Decorrelator.



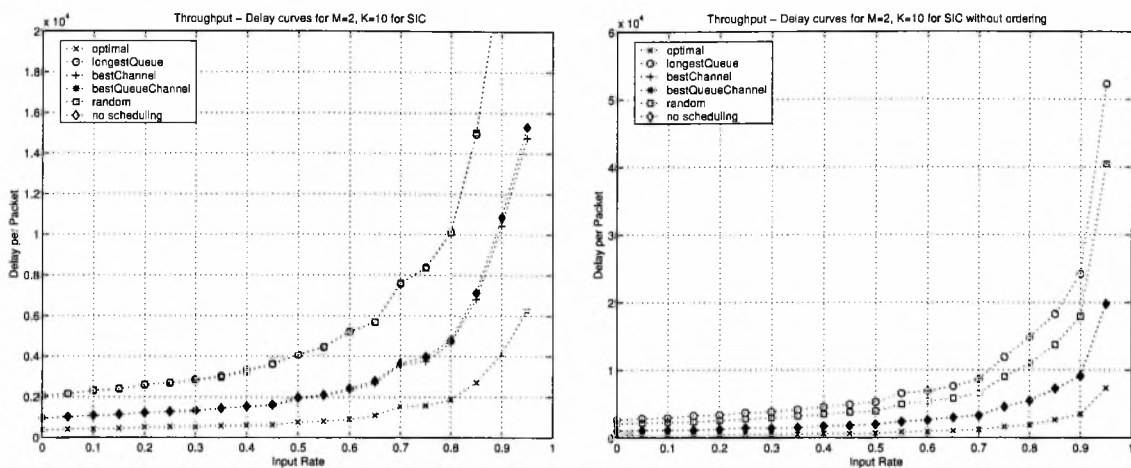
Σχήμα 4.4: Συγκριτικές καμπύλες throughput - delay για τις πολιτικές χρονοπρογραμματισμού optSched, longestQueue, bestChannel, bestQueueChannel, random και noSched σε  $10 \times 2$  και  $20 \times 2$  κανάλι, όπου εκτελείται Decorrelator.



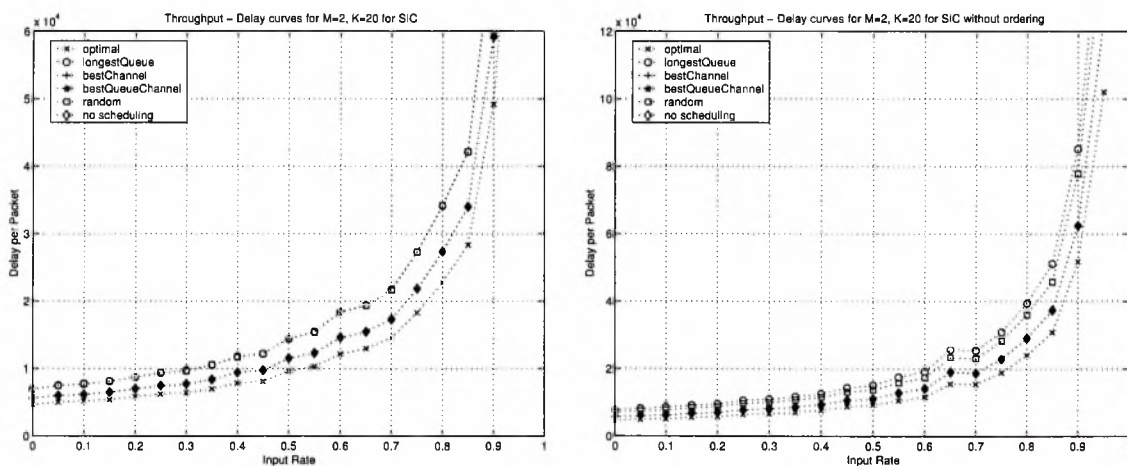
Σχήμα 4.5: Συγκριτικές καμπύλες throughput - delay για τις πολιτικές χρονοπρογραμματισμού optSched, longestQueue, bestChannel, bestQueueChannel, random και noSched με χρήση της maxqueue (αριστερά) ή χωρίς (δεξιά), σε  $5 \times 2$  κανάλι, όπου εκτελείται Successive Interference Canceller.



Σχήμα 4.6: Συγκριτικές καμπύλες throughput - delay για τις πολιτικές χρονοπρογραμματισμού optSched, longestQueue, bestChannel, bestQueueChannel, random και noSched με χρήση της maxqueue (αριστερά) ή χωρίς (δεξιά), σε  $5 \times 2$  κανάλι, όπου εκτελείται Successive Interference Canceller.



Σχήμα 4.7: Συγκριτικές καμπύλες throughput - delay για τις πολιτικές χρονοπρογραμματισμού optSched, longestQueue, bestChannel, bestQueueChannel, random και noSched με χρήση της maxqueue (αριστερά) ή χωρίς (δεξιά), σε  $5 \times 2$  κανάλι, όπου εκτελείται Successive Interference Canceller.



Σχήμα 4.8: Συγκριτικές καμπύλες throughput - delay για τις πολιτικές χρονοπρογραμματισμού optSched, longestQueue, bestChannel, bestQueueChannel, random και noSched με χρήση της maxqueue (αριστερά) ή χωρίς (δεξιά), σε  $5 \times 2$  κανάλι, όπου εκτελείται Successive Interference Canceller.





## Κεφάλαιο 5

# Σύνοψη Διπλωματικής Εργασίας

### 5.1 Συμπεράσματα

Το θέμα που μας απασχόλησε σε αυτή την εργασία ήταν το πρόβλημα της μεγιστοποίησης του throughput σε ένα ανοδικό κανάλι με  $K$  ασύρματους χρήστες με απλές κεραίες και έναν σταθμό βάσης με πολλαπλή κεραία  $M$  ενεργών στοιχείων. Για το παραπάνω κανάλι μελετήσαμε δυο διαφορετικά είδη δεκτών - ανιχνευτών, τον Decorrelating Detector και τον Successive Interference Cancellor. Σε περιβάλλον Matlab προσομοιώσαμε τη λειτουργία των δυο δεκτών - ανιχνευτών και εξετάσαμε τη συμπεριφορά και την απόδοσή τους (μέση καθυστέρηση ανά πακέτο) για διαφορετικά σενάρια.

Κατά την σύγκριση των δυο δεκτών (χωρίς χρονοπρογραμματισμό) για  $2 \times 2$ ,  $3 \times 3$  και  $8 \times 8$  κανάλι, παρατηρήσαμε πως ο Successive Interference Cancellor παρουσιάζει πάντοτε καλύτερη συμπεριφορά από τον Decorrelator. Ενδεικτικά αναφέρουμε πως για  $inputrate = 0.9$  η μέση καθυστέρηση πακέτου για τον Successive Interference Cancellor είναι μόλις το 58% της μέσης καθυστέρησης πακέτου για τον Decorrelator. Αυτό σημαίνει πως ο Successive Interference Cancellor είναι έως και 2 φορές καλύτερος του Decorrelator.

Ύστερα συγκρίναμε για κάθε δέκτη, Decorrelator και Successive Interference Cancellor, τις έξι πολιτικές χρονοπρογραμματισμού που παρουσιάσαμε στο κεφάλαιο (3), σε  $5 \times 2$ ,  $8 \times 2$ ,  $10 \times 2$  και  $20 \times 2$  κανάλι. Παρατηρήσαμε, ότι η πολιτική `optSched` εμφανίζει πάντοτε σημαντικά καλύτερη συμπεριφορά, όπως ήταν αναμενόμενο, από όλες τις υπόλοιπες πολιτικές. Η διαφορά της σε σύγκριση με τις άλλες πολιτικές είναι ακόμα μεγαλύτερη στην περίπτωση του Decorrelator. Αυτό συμβαίνει γιατί στην περίπτωση του Successive Interference Cancellor υιοθετήσαμε



την πολιτική `maxqueue` για την ταξινόμηση των χρηστών, πράγμα που βελτιώνει κατά μεγάλο βαθμό την απόδοση όλων των πολιτικών (εξαιρείται η `optSched` που δεν χρησιμοποιεί την εν λόγω πολιτική).

Επίσης, τόσο στην περίπτωση του `Decorrelator`, όσο και στην περίπτωση του `Successive Interference Cancellor` παρατηρήσαμε ότι οι πολιτικές `bestChannel` και `bestQueueChannel` παρουσίασαν μικρότερες καθυστερήσεις από τις άλλες υποβέλτιστες πολιτικές ή την πολιτική `noSched` (απουσία χρονοπρογραμματισμού). Αντιθέτως, οι πολιτικές `longestQueue` και `random` παρουσίασαν την χειρότερη συμπεριφορά.

## 5.2 Μελλοντικές κατευθύνσεις

Η μελέτη που έγινε στην παρούσα εργασία μπορεί να επεκταθεί προς αρκετές κατευθύνσεις. Καταρχήν, παρουσιάσαμε την βέλτιστη πολιτική χρονοπρογραμματισμού `optSched` και τη συγκρίναμε με απλές ευριστικές πολιτικές χρονοπρογραμματισμού. Πράγματι, αποδείχτηκε σε πειραματικό επίπεδο ότι η πολιτική `optSched` παρουσιάζει σημαντικά καλύτερη συμπεριφορά από όλες τις υπόλοιπες πολιτικές που παρουσιάστηκαν, με κόστος βέβαια τις μεγάλες απαιτήσεις σε χρόνο και χώρο. Παρά την πειραματική επιβεβαίωση απομένει να αποδειχτεί και θεωρητικά το γεγονός ότι η `optSched` είναι η βέλτιστη πολιτική χρονοπρογραμματισμού για το κανάλι που μελετούμε.

Στη συνέχεια, ασχοληθήκαμε με το πρόβλημα της από κοινού χωρικής επεξεργασίας σήματος με έλεγχο ισχύος. Στην περίπτωση του `Decorrelator` για το ανοδικό κανάλι, διαπιστώσαμε ότι κάθε χρήστης μπορεί να χρησιμοποιήσει όλη την ισχύ που έχει στη διάθεση του για να μεταδώσει, χάρη στην ιδιαίτερη δομή του συγκεκριμένου δέκτη. Αν υποθέσουμε ένα ειδικό σενάριο, όπου όλοι οι χρήστες χρησιμοποιούν ισχύ από μια 'δεξαμενή' ισχύος, τότε τίθεται πρόβλημα ελέγχου ισχύος και η προτεινόμενη πολιτική βασίζεται στον αλγόριθμο `Waterfilling` που παρουσιάσαμε στο (2.3). Αυτή η πολιτική μπορεί, σε επόμενη μελέτη, να προσαρμοστεί κατάλληλα για το αντίστοιχο καθοδικό κανάλι, όπου ένας σταθμός βάσης πρέπει να μεταδώσει δεδομένα σε  $K$  ασύρματους χρήστες με την βέλτιστη κατανομή ισχύος για να επιτύχει μέγιστο `throughput`.

Τέλος, ένα άλλο θέμα που έχει μείνει ανοιχτό είναι το πρόβλημα της από κοινού χωρικής

επεξεργασίας σήματος με έλεγχο ισχύος για το ανοδικό κανάλι, όταν στον σταθμό βάσης εκτελείται Successive Interference Cancellation. Μελετήσαμε το συγκεκριμένο πρόβλημα στην παράγραφο (2.3.2) και καταλήξαμε ότι υπάρχει μοναδικό ολικό βέλτιστο. Επίσης, διατυπώσαμε τις σχέσεις (2.3.33) και (2.3.34) για τη βέλτιστη κατανομή ισχύος. Ωστόσο, απομένει να βρεθεί η αναλυτική έκφραση για την βέλτιστη κατανομή ισχύος που λύνει το πρόβλημα μεγιστοποίησης του throughput. Φυσικά, εφόσον λυθεί το πρόβλημα για το ανοδικό κανάλι, η λύση μπορεί λόγω δυαδικότητας να μεταφερθεί στο αντίστοιχο καθοδικό. Συγκεκριμένα, στο καθοδικό κανάλι θα αποτελείται από έναν σταθμό βάσης με  $M$  ενεργά στοιχεία πολλαπλής κεραίας, όπου εκτελείται "Costa precoding", και  $K$  ασύρματους χρήστες που δέχονται δεδομένα από τον σταθμό βάσης. Σε αυτό το σενάριο, ο σταθμός βάσης θα πρέπει να αναθέσει κατάλληλα την διαθέσιμη ισχύ του για να μεταδώσει αποδοτικά και στους  $K$  χρήστες.



# Παράρτημα Α'

## Α'.1 Κώδικας Προσομοίωσης

---

```
% SINGLE - INPUT - MULTIPLE - OUTPUT
% MULTIPLE ACCESS CHANNEL
% K = M users with single antenna - 1 BS with M antennas
M=3;          % ELEMENTS OF MULTIPLE ANTENNA AT THE BS - NUMBER OF USERS
T=1000;       % SIMULATION DURATION - time slots
index=0; p=zeros(1,20); A=zeros(1,20); delay_sic=zeros(1,20);
delay_dec=zeros(1,20); for j=0:0.05:0.95
    index=index+1;
    p(index)=j;
    arrival=zeros(1,M);          % ARXIKOPOIHSH
    rate_sic=zeros(1,M);
    rate_dec=zeros(1,M);
    old_queue_sic=zeros(1,M);
    queue_sic=zeros(1,M);
```

```

old_queue_dec=zeros(1,M);
queue_dec=zeros(1,M);
Q_sic(index)=0;
Q_dec(index)=0;
a=0;
A(index)=0;
for t=1:T
    channel=random('rayl',1,M,M);
                                % TYXAI0 KANALI - METABALETAI SE KATHE
                                % SLOT ME BASH THN KATANOMH RAYLEIGH

    L=rand(1);
    if L >= p(index)            % BERNOULLI KATANOMH
        a=1;
        A(index)=A(index)+1;
    end                          % YPOLOGISMOS RATES

    rate_sic = max(sic_no_sched(M, M, channel, queue_sic),0);
    rate_dec = max(decorrelator(M, channel),0);
    if t==1
        queue_sic = max(a - rate_sic,0);
        queue_dec = max(a - rate_dec,0);
    else
        queue_sic = max(queue_sic - rate_sic,0) + a; % ENHMERWSH QUEUE
        queue_dec = max(queue_dec - rate_dec,0) + a;
    end

    Q_sic(index) = Q_sic(index) + sum(queue_sic);
    Q_dec(index) = Q_dec(index) + sum(queue_dec);

```

```

end
delay_sic(index) = Q_sic(index) / A(index);           % YPOLOGISMOS DELAY
delay_dec(index) = Q_dec(index) / A(index);
end
plot(p, delay_dec, 'o:', p, delay_sic, 'x:')          % EKTYPWSH GRAFHMATOS
grid axis([0 1 0 150]) xlabel('Input Rate (packets/slot)')
ylabel('Delay per Packet') title('Throughput - Delay Curves for M=8
antennas, K=8 users') legend('Decorrelator', 'SIC', 0)

```

---

```

% METHOD decorrelator
% decorrelator calculates the rate vector for the scheduled subset of users
% when a decorrelating detector is implemented at the BS
function rate_dec = decorrelator(M, channel);
rate_dec=zeros(1,M);
g=pinv(channel);                                     % YPOLOGISMOS BEAMFORMER
for i=1:M                                           % YPOLOGISMOS SINR
    dsinr(i)=((g(i,:)*channel(:,i))^2)/(0.1*g(i,:)*g(i,:));
    rate_dec(i)=round(log(1+dsinr(i)));              % YPOLOGISMOS RATE
end

```

---

```

% METHOD sic
% sic calculates the rate vector for the scheduled subset of users
% when a successive interference canceller is implemented at the BS
function rate_sic = sic(M, channel); beams=zeros(M,M);

```

```

rate_sic=zeros(1,M);
for i=1:M
    s=zeros(M,M);
    for j=(i+1):M
        s=s+(channel(:,j)*channel(:,j)');
    end
    f=inv(eye(M)+s)*channel(:,i);
    if i==1
        beams=f;
    else
        beams=[beams f];    % YPOLOGISMOS BEAMFORMER
    end
end
for i=1:M
    interf=0;
    for j=(i+1):M          % YPOLOGISMOS INTERFERENCE
        interf=interf+(beams(:,i)'*channel(:,j));
    end                    % YPOLOGISMOS SINR
    ssinr(i)=(beams(:,i)'*channel(:,i))^2/(interf^2 + 0.1);
    rate_sic(i)=round(log(1+ssinr(i)));    % YPOLOGISMOS RATE
end

```

---

```

% SINGLE - INPUT - MULTIPLE - OUTPUT

```

```

% MULTIPLE ACCESS CHANNEL

```

```

% K users with single antenna - 1 BS with M antennas

```

```
K=6;                % WIRELESS USERS
M=4;                % ELEMENTS OF MULTIPLE ANTENNA AT THE BS
T=1000;            % SIMULATION DURATION - time slots
index=0; p=zeros(1,20); Q=zeros(1,20); A=zeros(1,20);
delay_opt=zeros(1,20); delay_queue=zeros(1,20);
delay_channel=zeros(1,20); delay_queue_channel=zeros(1,20);
delay_rand=zeros(1,20); delay_no_sched=zeros(1,20);

for j=0:0.05:0.95
    index=index+1;
    p(index)=j;
    arrival=zeros(1,K);
    rate_opt=zeros(1,K);
    rate_queue=zeros(1,K);
    rate_channel=zeros(1,K);
    rate_queue_channel=zeros(1,K);
    rate_rand=zeros(1,K);
    rate_no_sched=zeros(1,K);
    old_queue_opt=zeros(1,K);
    queue_opt=zeros(1,K);
    old_queue_queue=zeros(1,K);
    queue_queue=zeros(1,K);
    old_queue_channel=zeros(1,K);
    queue_channel=zeros(1,K);
    old_queue_queue_channel=zeros(1,K);
    queue_queue_channel=zeros(1,K);
```



```

old_queue_rand=zeros(1,K);
queue_rand=zeros(1,K);
old_queue_no_sched=zeros(1,K);
queue_no_sched=zeros(1,K);
Q_opt(index)=0;
Q_queue(index)=0;
Q_channel(index)=0;
Q_queue_channel(index)=0;
Q_rand(index)=0;
Q_no_sched(index)=0;
a=0;
A(index)=0;
for t=1:T
    channel=random('rayl',1,M,K);
    %a=0;
    L=rand(1);
    if L >= p(index)                % BERNOULLI KATANOMH
        a=1;
        A(index)=A(index)+1;
    end
    rate_opt=max(dec_opt_sched(K, M, channel, queue_opt),0);
    rate_queue=max(dec_queue_sched(K, M, channel, queue_queue),0);
    rate_channel=max(dec_channel_sched(K, M, channel),0);
    rate_queue_channel=max(dec_queue_channel_sched(K, M, channel,
        queue_queue_channel),0);
    rate_rand=max(dec_rand_sched(K, M, channel),0);

```

```

rate_no_sched=max(decorrelator(K, M, channel),0);
if t==1
    queue_opt=max(a - rate_opt,0);
    queue_queue=max(a - rate_queue,0);
    queue_channel=max(a - rate_channel,0);
    queue_queue_channel=max(a - rate_queue_channel,0);
    queue_rand=max(a - rate_rand, 0);
    queue_no_sched=max(a - rate_no_sched, 0);
else
    queue_opt=max(queue_opt - rate_opt, 0) + a;
    queue_queue=max(queue_queue - rate_queue, 0) + a;
    queue_channel=max(queue_channel - rate_channel, 0) + a;
    queue_queue_channel=max(queue_queue_channel - rate_queue_channel,
        0) + a;
    queue_rand=max(queue_rand - rate_rand, 0) + a;
    queue_no_sched=max(queue_no_sched - rate_no_sched, 0) + a;
end
Q_opt(index)=Q_opt(index) + sum(queue_opt);
Q_queue(index)=Q_queue(index) + sum(queue_queue);
Q_channel(index)=Q_channel(index) + sum(queue_channel);
Q_queue_channel(index)=Q_queue_channel(index) +
    sum(queue_queue_channel);
Q_rand(index)=Q_rand(index) + sum(queue_rand);
Q_no_sched(index)=Q_no_sched(index) + sum(queue_no_sched);
end
delay_opt(index)=Q_opt(index)/A(index);

```

```

delay_queue(index)=Q_queue(index)/A(index);
delay_channel(index)=Q_channel(index)/A(index);
delay_queue_channel(index)=Q_queue_channel(index)/A(index);
delay_rand(index)=Q_rand(index)/A(index);
delay_no_sched(index)=Q_no_sched(index)/A(index);
end plot(p, delay_opt, 'x:', p, delay_queue, 'o:', p, delay_channel,
'+:', p, delay_queue_channel, '*:', p, delay_rand, 's:', p,
delay_queue_channel, 'd:') grid axis([0 1 0 20000])
xlabel('InputRate') ylabel('Delay per Packet') title('Throughput -
Delay curves for M=2, K=6 for Decorrelator/SIC') legend('optimal',
'longestQueue', 'bestChannel', 'bestQueueChannel', 'random', 'no
scheduling', 0)

```

---

```

% METHOD dec_opt_sched
% dec_opt_sched chooses the subset of users that achieves maximum sum rate
function rate_dec_opt = dec_opt_sched(K, M, channel, queue);
c=combntns(1:K,M); [A,B]=size(c); nqueue=zeros(A,B);
temp_rate_opt=zeros(A,M); throughput=zeros(1,A); maxthrou=0;
indice=0; rate_dec_opt=zeros(1,K);
for i=1:A
    h=zeros(B,B);
    for j=1:B
        for k=1:K
            if c(i,j)==k
                if h==0

```

```

        h=channel(:,k);
    else
        h=[h channel(:,k)];
    end
    nqueue(i,j)=queue(k);
end
end
end
temp_rate_opt(i,:)=decorrelator(M, M, h); % YPOLOGISMOS RATE GIA KATHE
end % DYNATO YPOSYNOLO
for i=1:A
    for j=1:B
        throughput(i)=throughput(i) + nqueue(i,j)*temp_rate_opt(i,j);
    end
end
for i=1:A
    if maxthrou <= throughput(i) % EYRESH YPOSYNOLOY ME TO
        maxthrou=throughput(i); % MEGISTO THROUGHPUT
        indice=i;
    end
end
for j=1:B
    for k=1:K % ANTISTOIXHSH RATE SE KATHE XRHSTH
        if c(indice,j)==k
            rate_dec_opt(k)=temp_rate_opt(indice,j);
        end
    end
end

```

```

    end
end
-----

% METHOD dec_queue_sched
% dec_queue_sched chooses a subset of users with the longest queue lengths
function rate_dec_queue = dec_queue_sched(K, M, channel, queue);
c=combntrns(1:K,M); [A,B]=size(c); temp_rate_queue=zeros(A,M);
sum_queues=zeros(1,A); maxqueue=0; indice=0;
rate_dec_queue=zeros(1,K);
for i=1:A
    for j=1:B
        for k=1:K % EYRESH YPOSYNOLOY XRHSTWN ME TA MEGALYTERA MHKH OYRAS
            if c(i,j)==k
                sum_queues(i)=sum_queues(i)+queue(k);
            end
        end
    end
    if maxqueue <= sum_queues(i)
        maxqueue=sum_queues(i);
        indice=i;
    end
end
end
h=zeros(B,B);
for j=1:B
    for k=1:K

```

```

        if c(indice,j)==k
            if h==0
                h=channel(:,k);
            else
                h=[h channel(:,k)];
            end
        end
    end
end

end % YPOLOGISMOS RATE

temp_rate_queue(indice,:)=decorrelator(M, M,h);

for j=1:B
    for k=1:K
        if c(indice,j)==k % ANTISTOIXHSH RATE
            rate_dec_queue(k)=temp_rate_queue(indice,j);
        end
    end
end

end

-----

% METHOD dec_channel_sched
% dec_channel_sched chooses a subset of users with the largest channel norms
function rate_dec_channel = dec_channel_sched(K, M, channel);
c=combnats(1:K,M); [A,B]=size(c); temp_rate_channel=zeros(A,M);
sum_channel=zeros(1,A); max_channel=0; indice=0;
rate_dec_channel=zeros(1,K);
for i=1:A

```

```

for j=1:B % EYRESH YPOSYNOLOY XRHSTWN ME TH MEGALYTERH NORMA KANALIOY
    for k=1:K
        if c(i,j)==k
            sum_channel(i)=sum_channel(i)+(channel(1,k)*channel(1,k)+
                channel(2,k)*channel(2,k));
        end
    end
end
if max_channel <= sum_channel(i)
    max_channel = sum_channel(i);
    indice = i;
end
end
h=zeros(B,B);
for j=1:B
    for k=1:K
        if c(indice,j)==k
            if h==0
                h=channel(:,k);
            else
                h=[h channel(:,k)];
            end
        end
    end
end
end temp_rate_channel(indice,:)=decorrelator(M, M, h); % YPOLOGISMOS RATE
for j=1:B

```

```

for k=1:K
    if c(indice,j)==k                                % ANTISTOIXHSH RATE
        rate_dec_channel(k)=temp_rate_channel(indice,j);
    end
end
end
end

-----

% METHOD dec_queue_channel_sched
% dec_queue_channel_sched chooses a subset of users with the largest product
    queue length times channel norm
function rate_dec_queue_channel = dec_queue_channel_sched(K, M,
channel, queue);
c=combnats(1:K,M);
[A,B]=size(c);
temp_rate_queue_channel=zeros(A,M); sum_queue_channel=zeros(1,A);
max_queue_channel=0; indice=0; rate_dec_queue_channel=zeros(1,K);
for i=1:A
    for j=1:B          % EYRESH YPOSYNOLOY XRHSTWN ME TO MEGALYTERO GINOMENO
        for k=1:K      % MHKOYS OYRAS EPI NORMA KANALIOY
            if c(i,j)==k
                sum_queue_channel(i)=sum_queue_channel(i)+queue(k)*
                    (channel(1,k)*channel(1,k)+channel(2,k)*channel(2,k));
            end
        end
    end
end
end
end

```



```

if max_queue_channel <= sum_queue_channel(i)
    max_queue_channel = sum_queue_channel(i);
    indice = i;
end
end
h=zeros(B,B);
for j=1:B
    for k=1:K
        if c(indice,j)==k
            if h==0
                h=channel(:,k);
            else
                h=[h channel(:,k)];
            end
        end
    end
end
end

temp_rate_queue_channel(indice,:)=decorrelator(M, M, h); % YPOLOGISMOS RATE
for j=1:B
    for k=1:K
        if c(indice,j)==k % ANTISTOIXHSH RATE
            rate_dec_queue_channel(k)=temp_rate_queue_channel(indice,j);
        end
    end
end
end

```

---

```
% METHOD dec_rand_sched
% dec_rand_sched chooses a random subset of users
function rate_dec_rand = dec_rand_sched(K, M, channel);
c=combntns(1:K,M); [A,B]=size(c); temp_rate_rand=zeros(A,M);
maxvalue=0; indice=0; rate_dec_rand=zeros(1,K);
rand_value=rand(1,A);
for i=1:A
    if maxvalue <= rand_value(i)                % EPILOGH TYXAI OY YPOSYNOLOY
        maxvalue=rand_value(i);
        indice=i;
    end
end
h=zeros(B,B);
for j=1:B
    for k=1:K
        if c(indice,j)==k
            if h==0
                h=channel(:,k);
            else
                h=[h channel(:,k)];
            end
        end
    end
end
end
end
```

```

temp_rate_rand(indice,:)=decorrelator(M, M, h);           % YPOLOGISMOS RATE
for j=1:B
    for k=1:K
        if c(indice,j)==k
            rate_dec_rand(k)=temp_rate_rand(indice,j); % ANTISTOIXHSH RATE
        end
    end
end
end

```

---

```

% METHOD sic_opt_sched
% sic_opt_sched chooses the subset of users that achieves maximum throughput
function rate_sic_opt = sic_opt_sched(K, M, channel, queue);
c = combntns(1:K,M);           % SYNDYASMOI TWN K ANA M
[A,B] = size(c);
for i = 1:A
    d(:,:,i) = perms(c(i,:)); % M! METATHESEIS KATHE YPOSYNOLOY
end
[D1,D2,D3] = size(d);
temp_rate_opt=zeros(D1,D2,D3);
for i=1:D3
    for j=1:D1
        h=zeros(D2,D2);
        for k=1:D2
            for m=1:K
                if d(j,k,i)==m

```

```

        if h==0
            h=channel(:,m); % PINAKAS KANALIOY GIA TO YPOSYNOLO
        else
            h=[h channel(:,m)];
        end
        % DIANYSMA OYRWN GIA TO YPOSYNOLO
        squeue(j,k,i)=queue(m);
    end
end
end
end
% RYTHMOI METADOSHS GIA TO YPOSYNOLO
temp_rate_opt(j,:,i)=sic(M, M, h);
end
end
maxthrou=0;
throughput=zeros(D3,D1);
for i=1:D3
    for j=1:D1
        for k=1:D2
            % YPOLOGISMOS THROUGHPUT GIA TO KATHE YPOSYNOLO
            throughput(i,j)=throughput(i,j) + squeue(j,k,i)*
                temp_rate_opt(j,k,i);
        end
    end
    if maxthrou <= throughput(i,j) % EYRESH YPOSYNOLOY POY EPITYGXANEI
        % MEGISTO THROUGHPUT
        maxthrou=throughput(i,j);
        indice1=i;
        indice2=j;
    end
end

```

```

        end
    end
end
rate_sic_opt=zeros(1,K);
for m=1:K
    for k=1:D2
        if d(indice2,k,indice1) == m % RYTHMOI METADOSHS OLWN TWN XTHSTWN
            rate_sic_opt(m) = temp_rate_opt(indice2,k,indice1);
        end
    end
end
end

```

---

```

% METHOD sic_queue_sched
% sic_queue_sched chooses the subset of users with longest queue lengths
function rate_sic_queue = sic_queue_sched(K, M, channel, queue);

q =queue; indice = zeros(1,M); temp_rate = zeros(1,M);
rate_sic_queue = zeros(1,K);
for i = 1:M
    maxq = 0;
    for j = 1:K
        if maxq <= q(j)
            maxq = q(j); % EPILOGH TWN M XRHSTWN ME TA MEGALYTERA QUEUES
            indice(M-i+1) = j;
        end
    end
end

```

```

end
schannel(:,M-i+1) = channel(:,indice(M-i+1));% TAKSINOMHSH TWN M XRHSTWN
q(indice(M-i+1)) = 0;          % SE SEIRA AYKSANOMENYOY MHKOYS OYRAS
end

temp_rate = sic(M, M, schannel);          % YPOLOGISMOS RATE
for j = 1:K
    for i = M:-1:1
        if indice(i)==j          % ANTISTOIXHSH RATE SE KATHE XRHSTH
            rate_sic_queue(j) = temp_rate(i);
        end
    end
end

end

-----

% METHOD sic_channel_sched
% sic_channel_sched chooses the subset of users with largest channel norm
function rate_sic_channel = sic_channel_sched(K, M, channel, queue);
h = channel; ind1 = zeros(1,M); ind2 = zeros(1,M); ind3 =
zeros(1,M); temp_rate = zeros(1,M); rate_sic_channel = zeros(1,K);

for i = 1:M
    max_channel = 0;
    for j = 1:K
        if max_channel <= (h(:,j))'*h(:,j))
            max_channel = h(:,j)' * h(:,j); % EPILOGH TWN M XRHSTWN ME

```

```

        ind1(i) = j;                                % TH MEGALYTERH NORMA KANALIOY
    end
end
if i == 1
    schannel = h(:,ind1(i));
    squeue = queue(ind1(i));
else
    schannel = [schannel h(:,ind1(i))];
    squeue = [squeue queue(ind1(i))];
end
h(:,ind1(i)) = 0;
end

for i = 1:M
    maxq = 0;
    for j = 1:M
        if maxq <= squeue(j)
            maxq = squeue(j);                       % TAKSINOMHSH XRHSTWN SE SEIRA
            ind2(i) = j;                             % AYKSANOMENOU MHKOYS OYRAS
        end
    end
    nqueue(M-i+1) = maxq;
    nchannel(:,M-i+1) = schannel(:,ind2(i));
    ind3(M-i+1) = ind1(ind2(i));
    squeue(ind2(i))=0;
end
end

```

```
temp_rate = sic(M, M, nchannel); % YPOLOGISMOS RATE
```

```
for j=1:K
    for i=M:-1:1
        if ind3(i)==j % ANTISTOIXISH RATE SE KATHE XRHSTH
            rate_sic_channel(j) = temp_rate(i);
        end
    end
end
end
```

```
-----
% METHOD sic_queue_channel_sched
```

```
% sic_queue_channel_sched chooses the subset of users
```

```
% with product of queue length times largest channel norm
```

```
function rate_sic_queue_channel = sic_queue_channel_sched(K, M,
channel, queue); h = channel; ind1 = zeros(1,M); ind2 = zeros(1,M);
ind3 = zeros(1,M); temp_rate = zeros(1,M); rate_sic_queue_channel =
zeros(1,K);
```

```
for i = 1:M
```

```
    max_queue_channel = 0;
```

```
    for j = 1:K
```

```
        if max_queue_channel <= (queue(j)*h(:,j)')*h(:,j))
```

```
            max_queue_channel = queue(j)*h(:,j)')*h(:,j);
```

```
            ind1(i) = j; % EPILOGH TWN M XRHSTWN ME TO MEGALYTERO
```

```
            % GINOMENO MHKOYS OYRAS EPI NORMA KANALIOY
```



```

        end
    end
    if i == 1
        schannel = h(:,ind1(i));
        squeue = queue(ind1(i));
    else
        schannel = [schannel h(:,ind1(i))];
        squeue = [squeue queue(ind1(i))];
    end
    end
    h(:,ind1(i)) = 0;
end

for i = 1:M
    maxq = 0;
    for j = 1:M
        if maxq <= squeue(j)
            maxq = squeue(j);           % TAKSINOMHSH XRHSTWN SE SEIRA
            ind2(i) = j;                 % AYKSANOMENOUY MHKOYS OYRAS
        end
    end
    end
    nqueue(M-i+1) = maxq;
    nchannel(:,M-i+1) = schannel(:,ind2(i));
    ind3(M-i+1) = ind1(ind2(i));
    squeue(ind2(i))=0;
end
temp_rate = sic(M, M, nchannel);      % YPOLOGISMOS RATE

```

```

for j=1:K
    for i=M:-1:1
        if ind3(i)==j                % ANTISTOIXISH RATE SE KATHE XRHSTH
            rate_sic_queue_channel(j) = temp_rate(i);
        end
    end
end
end

```

---

```

% METHOD sic_rand_sched
% dec_rand_sched chooses a random subset of users
function rate_sic_rand = sic_rand_sched(K, M, channel, queue); c =
combntns(1:K,M); [A,B] = size(c);
rand_value = rand(1,A); maxv = 0;

for i = 1:A
    if maxv <= rand_value(i)
        maxv = rand_value(i);      % TYXAIA EPILOGH
        ind1 = i;
    end
end

end

schannel = zeros(M,M);

for j = 1:M
    for k = 1:K

```

```

    if c(ind1,j) == k
        if schannel == 0
            schannel = channel(:,k);
            squeue = queue(k);
        else
            schannel = [schannel channel(:,k)];
            squeue = [squeue queue(k)];
        end
    end
end
end

for i = 1:M
    maxq = 0;
    for j = 1:M
        if maxq <= squeue(j)
            maxq = squeue(j);           % TAKSINOMHSH XRHSTWN SE SEIRA
            ind2(i) = j;                 % AYKSANOMENYOY MHKOYS OYRAS
        end
    end
end
nqueue(M-i+1) = maxq;
nchannel(:,M-i+1) = schannel(:,ind2(i));
ind3(M-i+1) = c(ind1,ind2(i));
squeue(ind2(i))=0;

end

```

```

temp_rate = sic(M, M, nchannel);    % YPOLOGISMOS RATE
rate_sic_rand = zeros(1,K);

for j=1:K
    for i=M:-1:1
        if ind3(i)==j                % ANTISTOIXISH RATE SE KATHE XRHSTH
            rate_sic_rand(j) = temp_rate(i);
        end
    end
end
end

```

---

```

% METHOD sic_no_sched
% sic_no_sched performs no scheduling
function rate_sic_no_sched = sic_no_sched(K, M, channel, queue);
squeue = queue; indice = zeros(1,M); temp_rate = zeros(1,M);
rate_sic_channel = zeros(1,K);

for i = 1:K
    maxq = 0;
    for j = 1:K
        if maxq <= squeue(j)
            maxq = squeue(j);        % TAKSINOMHSH XRHSTWN SE SEIRA
            indice(K-i+1) = j;      % AYKSANOMENOY MHKOYS OYRAS
        end
    end
end
end

```

```
nqueue(K-i+1) = maxq;
nchannel(:,K-i+1) = channel(:,indice(K-i+1));
squeue(indice(K-i+1))=0;
end
temp_rate = sic(K, M, nchannel);    % YPOLOGISMOS RATE

for j = 1:K
    for i = 1:K
        if indice(K-i+1) == j      % ANTISTOIXISH RATE SE KATHE XRHSTH
            rate_sic_no_sched(j) = temp_rate(K-i+1);
        end
    end
end
end
```

---

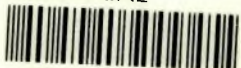
# Βιβλιογραφία

- [1] Anastasios Giannoulis, Konstantinos P. Tsoukatos and Leandros Tassioulas, *Cross-layer power control in wireless networks*, (2005).
- [2] ———, *Maximum Throughput Power Control in CDMA Wireless Networks*, (2005).
- [3] D. Tse and S. Hanly, *Multi-access fading channels: Part i: Polymatroid structure, optimal resource allocation and throughput capacities*, IEEE Transactions on Information Theory vol. **44** (1998), pp. 2796–2815.
- [4] David Tse and Pramond Viswanath, *Fundamentals of wireless communication*, Cambridge University Press, 2004.
- [5] Edwin K. P. Chong and Stanislaw H. Zak, *An introduction to optimization*, John Wiley and sons, Inc., New York, 2001.
- [6] H. Boche and M. Schubert, *Solution of the SINR downlink beamforming problem*, Proc. 36th Conf. Inf. Sci. Syst. (2002).
- [7] H. Boche and M. Wiczanowski, *Optimal scheduling for high speed uplink packet access - a cross-layer approach*, Proceedings VTC Spring (2004).
- [8] H. Boche, E. A. Jorswieck and T. Haustein, *Channel Aware Scheduling for Multiple Antenna Multiple Access Channels*, (2003).
- [9] Holger Boche and Marcin Wiczanowski, *Stability Region of Arrival Rates and Optimal Scheduling for MIMO-MAC - A Cross Layer Approach*, (2004).
- [10] Holger Boche and Martin Schubert, *Iterative Multiuser Uplink and Downlink Beamforming Under SINR Constraints*, IEEE Transactions on Signal Processing (2005).
- [11] ———, *Resource Allocation for Multi-Antenna Multi-User Systems*, (2005).
- [12] Iordanis Koutsopoulos, Konstantinos P. Tsoukatos and Konstantinos Aggelis, *Physical Layer Techniques and Maximum Throughput Scheduling with Antenna Arrays*, (2005).
- [13] Krishnan Kumaran and Lijun Qian, *Uplink Scheduling in CDMA Packet-Data Systems*, IEEE INFOCOM 2003.
- [14] L. Tassioulas and A. Ephremides, *Stability Properties of Constraint Queueing Systems and Scheduling Policies for Maximum Throughput in Multihop Radio Networks*, IEEE Trans. Aut. Control vol. **37** (1992), pp. 1936–1948.

- [15] M. Schubert and H. Boche, *Solution of the multi-user downlink beamforming problem with individual SINR constraints*, IEEE Trans. Veh. Technol. **vol. 53** (2004), pp. 18–28.
- [16] Martin Schubert and Holger Boche, *An Efficient Algorithm for Optimum Joint Downlink Beamforming and Power Control*, (2002).
- [17] Mung Chiang, *Balancing Transport and Physical Layers in Wireless Multihop Networks: Jointly Optimal Congestion Control and Power Control*, IEEE Jour. on Select. Areas in Communications **vol. 23** (2005), pp. 104–116.
- [18] Pramod Viswanath and David N. C. Tse, *Sum Capacity of the Vector Gaussian Broadcast Channel and Uplink - Downlink Duality*, IEEE International Symposium on Information Theory (2002).
- [19] Pramod Viswanath, Venkat Anantharam, David N. C. Tse, *Optimal Sequences, Power Control, and User Capacity of Synchronous CDMA Systems with Linear MMSE Multiuser Receivers*, IEEE Transactions on Information Theory (1999).
- [20] Stephen Boyd and Lieven Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.



ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΘΕΣΣΑΛΙΑΣ



004000085896