

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ & ΠΛΗΡΟΦΟΡΙΚΗΣ

ΒΑΣΙΚΕΣ ΑΡΧΕΣ ΔΙΚΤΥΩΝ ΚΙΝΗΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ

ΘΕΜΑΤΑ PROJECT

ΧΕΙΜΕΡΙΝΟ ΕΞΑΜΗΝΟ 2025-2026

Περιεχόμενα

1	Βέλτιστη τοποθέτηση σταθμών βάσης με υπολογιστική γεωμετρία.....	2
2	Αλγόριθμοι ανάθεσης συχνοτήτων LTE σε δίκτυο 4G.....	3
3	Εύρεση βέλτιστης διαδρομής διάδοσης πακέτων σε δίκτυο 4G	3
4	Προσομοίωση εναλλαγής κυψελίδων (handovers) σε δίκτυο 4G.....	4
5	Προσομοίωση δικτύου με NS-3.....	5
6	Προσομοίωση κινητού δικτύου με Python βάσει containers	8
7	Εντοπισμός παραβιάσεων SLAs και εξισορρόπηση φορτίου σε 4G	8
8	Έλεγχος χρησιμοποίησης πόρων σε δίκτυο με χρήση containers.....	9
9	Θεμελιώδεις αρχές και κατασκευή ενός συστήματος OFDM	11
10	Ανάλυση και προσομοίωση αλγόριθμων κατανομής πόρων	11
11	Βασικές έννοιες και προσαρμοστική διάδοση MIMO.....	12
12	Κάλυψη, ανταγωνισμός, και λήψη αποφάσεων σε δίκτυο 4G	13
13	Προσομοίωση event-driven δικτύου σε Python (και SimPy)	14
14	Προσομοίωση κυκλοφορίας δικτύου σε Python (και SimPy).....	15
15	Προσομοίωση δικτύου 4G κι εφαρμογή scheduling (PF, MT)	16
16	Προσομοίωση δικτύου 4G κι εφαρμογή scheduling (RR, WFQ)	17
17	Αλγόριθμοι scheduling για downlink σε LTE-4G δίκτυα σε MATLAB.....	18
18	Αλγόριθμοι scheduling για uplink σε LTE-4G δίκτυα σε MATLAB.....	19
19	Εφαρμογή clustering για τοποθέτηση σταθμών βάσης σε MATLAB.....	20
20	Προσομοίωση handover με τυχαίους περίπατους σε MATLAB	21
	Γενικές οδηγίες για όλα τα θέματα project.	22

1 Βέλτιστη τοποθέτηση σταθμών βάσης με υπολογιστική γεωμετρία

Γενική περιγραφή. Σε ένα δίκτυο LTE, η βέλτιστη τοποθέτηση σταθμών βάσης (Base Stations- BSs), είναι αναγκαία, καθώς κύριος σκοπός είναι η βέλτιστη κάλυψη του δικτυακού χώρου και η ικανοποίηση του μέγιστου αριθμού User Equipment- UEs, με τον πιο οικονομικό τρόπο, δηλαδή λιγότερα Base Stations. Στόχος της εργασίας είναι ο σχεδιασμός ενός αλγορίθμου βέλτιστης τοποθέτησης BSs σε ένα πεδίο δύο διαστάσεων (2D), για τον σκοπό της εξυπηρέτησης UEs, ενώ ταυτόχρονα λαμβάνεται υπόψη η κάλυψη (coverage) και η εξισορρόπηση φόρτου (load balancing). Η υλοποίηση του θέματος προτείνεται να γίνει σε Python.

Σχεδιασμός χώρου. Καλείστε να επιλέξετε dataset δύο διαστάσεων της αρεσκείας σας (το κάθε σημείο να αναπαριστά ένα UE). Το κάθε UE θα αποτελείται από συντεταγμένες (x, y). Πάνω σε αυτό το dataset θα τοποθετείτε αριθμό BSs N, με συντεταγμένες (x, y). Σας δίνετε κάτωθι link με datasets, να επιλέξτε όποιο προτιμάτε: <https://www.kaggle.com/datasets/samoilovmikhail/2d-clustering-dataset-collection>.

Αλγορίθμική υλοποίηση. Αναπτύξτε τον αλγόριθμο (μη επιβλεπόμενης) συσταδοποίησης (clustering) k-means. Με την αρωγή του τοποθετείστε BSs πάνω στο επίπεδο και προσδιορίστε τις ομάδες κάλυψης (clusters), στις οποίες θα ανήκουν τα UEs. Σε κάθε iteration του αλγόριθμου καταγράψτε τις αντίστοιχες τιμές αξιολόγησης. Για την καλύτερη κατανόηση της τοπολογίας και του πεδίου κάλυψης (coverage) του δικτύου, καλείστε να κάνετε αναπαράσταση του χώρου με Voronoi διαγράμματα. Σας δίνεται η βιβλιοθήκη `scipy.spatial.voronoi` για να χρησιμοποιήσετε:

<https://docs.scipy.org/doc/scipy/reference/generated/scipy.spatial.Voronoi.html>.

Επιπλέον, πολύ σημαντικό ρόλο σε ένα LTE δίκτυο διαδραματίζει και η αποτίμηση όλων των πιθανών παρεμβολών (interference). Ως interference ορίζεται η ανεπιθύμητη αλληλεπίδραση μεταξύ σημάτων διαφορετικών UEs ή BSs, η οποία οδηγεί σε μείωση της ποιότητας μετάδοσης. Για τον σκοπό αυτό, ζητείται η υλοποίηση αλγορίθμου εύρεσης γειτονικών κόμβων BSs, μα και η εκτίμηση της πιθανότητας παρεμβολών πάνω στο δίκτυο.

Αξιολόγηση. Για κάθε iteration των αλγορίθμων σας, υπολογίστε:

- Μέση απόσταση κάλυψης (Mean distance): Ο στόχος είναι η ελαχιστοποίηση της μέσης απόστασης όλων των UEs από τον πλησιέστερο BS.
- Ισορροπία φόρτου (Load balance): Η εξασφάλιση ότι τα UEs κατανέμονται ομοιόμορφα στους διαθέσιμους BSs, ώστε να αποφευχθεί η υπερφόρτωση μεμονωμένων σταθμών.
- Πιθανότητα παρεμβολών (Interference probability): Εκτίμηση της πιθανότητας παρεμβολών λόγω γειτονικών BSs που λειτουργούν στην ίδια συχνότητα.
- Περιοχές χωρίς κάλυψη (Dead areas): Ο υπολογισμός του ποσοστού των UEs που βρίσκονται εκτός περιοχής κάλυψης, δηλαδή με λαμβανόμενη ισχύ κάτω από ένα προκαθορισμένο κατώφλι, που θα ορίσετε.

Οπτικοποίηση. Παράγετε τα κατάλληλα plots (barplots, scatter plots, κοκ.) για το πως βελτιώνονται οι μετρικές αξιολόγησης ανά iteration. Οπτικοποιήστε το δίκτυο και κάνετε plot τα UEs και τα BSs, ώστε να είναι ξεκάθαρες οι περιοχές κάλυψης με Voronoi διαγράμματα.

Σημείωση: Για το τελευταίο ερώτημα, χρησιμοποιήστε τον τύπου του Friis

(https://en.wikipedia.org/wiki/Friis_transmission_equation):

$$P_r(d) = P_t \cdot G_t \cdot G_r \cdot \left(\frac{\lambda}{4\pi d} \right)^2 \approx \frac{1}{d^2}.$$

Για απλότητα θεωρούμε ότι όλες οι μεταβλητές είναι σταθερές, πλην της απόστασης.

Επιβλέπων: Γιάννης Κυριακόπουλος (ikyriakopoulos@isi.gr).

2 Αλγόριθμοι ανάθεσης συχνοτήτων LTE σε δίκτυο 4G

Γενική περιγραφή. Ο στόχος της άσκησης είναι η σχεδίαση αλγορίθμου που αναθέτει κανάλια σε User Equipments (UEs) σε ένα στατικό δίκτυο με Base Stations (BSs), με το οποίο να αναπαρίσταται σε 2D μορφή. Σκοπός είναι να μειωθούν πιθανές παρεμβολές (interference) μεταξύ UEs. Interference ορίζεται ως η ανεπιθύμητη αλληλεπίδραση μεταξύ σημάτων διαφορετικών χρηστών ή σταθμών βάσης, η οποία οδηγεί σε μείωση της ποιότητας μετάδοσης. Στην προκειμένη περίπτωση θα θεωρούμε interference όταν δύο γειτονικά UEs σε διαφορετικά BSs έχουν κάνει allocate το ίδιο κανάλι, και έχουν απόσταση μικρότερη από ένα threshold R. Επιπλέον, σημαντικό είναι να εξασφαλιστεί και η ισοκατανομή φόρτου (fairness) μεταξύ των BSs, δηλαδή κανένα BS δεν φορτώνεται δυσανάλογα. Η υλοποίηση του θέματος προτείνεται να γίνει σε Python.

System model. Υλοποιήστε ρουτίνα δημιουργίας 2D Plane (x, y), με είσοδο N UEs και K BSs. Κάθε BS θα αποτελείται από έναν αριθμό διαθέσιμων καναλιών [1, 2, 3, 4, ...], και μέγιστο αριθμό UEs που μπορεί να εξυπηρετήσει. Για απλούστευση, κάθε κανάλι θα αντιπροσωπεύει μια συχνότητα channel = |freq.| = 1, και θα γίνεται allocated από ένα UE κάθε φορά. Αν δύο UEs σε διαφορετικά BSs έχουν ίδιο κανάλι και απόσταση μικρότερη ενός threshold R, θεωρείται interference/conflict.

Πρόγραμμα δέσμευσης καναλιών. Υλοποιήστε πρόγραμμα που για κάθε UE βρίσκει τον πλησιέστερο BS με διαθέσιμο κανάλι, και το δεσμεύει. Σε περίπτωση που θα έχουν μείνει unallocated κανάλια σε κοντινούς UEs άλλων BSs, θα τα δεσμεύει σε αυτά. Σε περίπτωση που τα κανάλια ενός BS είναι γεμάτα, τα UEs θα παραμένουν unallocated. Επιπλέον, είναι επιθυμητό να γίνεται Fairness-aware assignment, δηλαδή να διατηρηθεί η ισοκατανομή του φόρτου, να μην να μην φορτωθεί ένας BS υπερβολικά, ενώ άλλα είναι άδεια. (Jane's fairness index)

Αξιολογήστε το πρόγραμμα που φτιάξατε με τις παρακάτω μετρικές.

- Unassigned UEs: πλήθος UEs που δεν εξυπηρετήθηκαν λόγω έλλειψης καναλιών στα Base Stations.
- Load per BS: πλήθος assigned UEs ανά BS.
- Interference metric: αριθμός conflicts-interference μεταξύ UEs διαφορετικών BS με ίδιο κανάλι και απόσταση μικρότερη από threshold R.
- Fairness index (Jain's): Χρησιμοποιήστε όποιο fairness index θέλετε (Jain's, Gini, κοκ.), και κρίνετε την κατανομή των UEs που κάνει το πρόγραμμα σας.
- Μέση απόσταση των assigned UEs ανά BS.

Ζητείται η οπτικοποίηση της τοπολογίας που δημιουργήσατε, χρωματισμένα UEs και BSs βάσει των allocations που έλαβαν χώρα, και οι κατάλληλες γραφικές αναπαραστάσεις της υλοποίησης.

Επιβλέπων: Γιάννης Κυριακόπουλος (ikyriakopoulos@isi.gr).

3 Εύρεση βέλτιστης διαδρομής διάδοσης πακέτων σε δίκτυο 4G

Γενική περιγραφή. Καλείστε να δημιουργήσετε ένα τηλεπικοινωνιακό δίκτυο με πολλαπλούς σταθμούς βάσης (BSs) με σκοπό την εύρεση βέλτιστης διαδρομής διάδοσης μηνυμάτων και πακέτων μεταξύ User Equipment (UEs). Θα χρησιμοποιήσετε γνωστούς αλγορίθμους εύρεσης βέλτιστης διαδρομής (shortest path algorithms) και θα τους αξιολογήσετε βάση των παρακάτω αναφερόμενων μετρικών. Η υλοποίηση του θέματος προτείνεται να γίνει σε Python.

System model. Δημιουργήστε δίκτυο 4G-LTE σε μορφή μη κατευθυνόμενου γράφου. Ο κάθε κόμβος θα αντιπροσωπεύει ένα BS, και η κάθε ακμή θα αντιπροσωπεύει κάποιο κανάλι διάδοσης πληροφορίας. Προσομοιώστε την καθυστέρηση διάδοσης πληροφορίας (latency) σε ένα κανάλι, με τη μορφή βάρους.

4G-LTE Projects

Μορφή μηνύματος. Θεωρείστε ότι UE θέλουν να στείλουν μηνύματα (src, dest, size, timestamp) μεταξύ τους. Τα src και dest αντιπροσωπεύουν UE (τον αποστολέα και τον παραλήπτη, αντίστοιχα). Size είναι το μέγεθος του μηνύματος, και timestamp η ώρα που απεστάλη. Θεωρήστε ως παραδοχή πως το κάθε κανάλι έχει την δυνατότητα να μεταφέρει μόνο ένα πακέτο σε κάθε timestep. Υλοποιήστε αλγορίθμους εύρεσης βέλτιστων μονοπατιών, όπως Dijkstra, A*, Bellman-Ford. Συγκρίνετε τους αλγορίθμους βάσει:

- Optimal Path: Εύρεση βέλτιστου μονοπατιού.
- Computational Time: Χρόνος που χρειάζεται ο αλγόριθμος για να βρει τη βέλτιστη διαδρομή.
- Packet Delivery Metrics.
 - Delay, χρόνος που χρειάζεται ένα πακέτο για να μεταδοθεί από τον πομπό στο δέκτη, συμπεριλαμβανομένων των delays των καναλιών.
 - Throughput, ρυθμός μετάδοσης δεδομένων.
 - Packet loss-timeouts, ορίστε threshold στην μετάδοση του πακέτου και υπολογίστε την απώλεια πακέτων.

Επίσης απαιτείται μια οπτικοποίηση της τοπολογίας του δικτύου και των UEs. Τέλος, παρέχετε γραφική αναπαράσταση των βασικών μετρικών, αλλά και εξηγήστε ποιος αλγόριθμός βρίκε το καλύτερο path, ποιος παρήγαγε τις καλύτερες μετρικές πακέτων, και ποιος ήταν overall καλύτερος.

Packet switching. Επαναλάβετε την άσκηση, αυτή την φορά όμως εφαρμόστε μέθοδο packet switching, δηλαδή χωρίστε ένα μήνυμα σε επιμέρους πακέτα. Κάντε επέκταση σε οιονδήποτε εκ των αλγόριθμων προτιμάτε, αυτή τη φορά όμως να βρίσκει τα k-shortest paths. Πετύχετε την όσο πιο δυνατή βέλτιστη συνολική μετάδοση. Χρησιμοποιήστε εναλλακτικά μονοπάτια που θα σας βοηθήσουν στην ταχύτερη συνολική μετάδοση.

Επιβλέπων: Γιάννης Κυριακόπουλος (ikyriakopoulos@isi.gr).

4 Προσομοίωση εναλλαγής κυψελίδων (handovers) σε δίκτυο 4G

Στόχος του project. Σκοπός είναι η προσομοίωση της διαδικασίας εναλλαγής κυψελίδων (handover) σε ένα απλοποιημένο δίκτυο 4G, όπου ο εξοπλισμός χρήστη (UE) κινείται μεταξύ των περιοχών κάλυψης των διαφορετικών σταθμών βάσης (BSs). Το handover μας διασφαλίζει ότι οι συνεδρίες επικοινωνίας (φωνητικές κλήσεις / μετάδοση δεδομένων) δε διακόπτονται ενώ ο χρήστης μετακινείται. Η υλοποίηση του θέματος προτείνεται να γίνει σε Python.

Ανάπτυξη του Περιβάλλοντος Προσομοίωσης.

- Μοντέλο Δικτύου: Σχεδιάστε ένα απλό μοντέλο δικτύου με πολλαπλούς σταθμούς βάσης (π.χ. 4-6), που είναι διατεταγμένοι σε σχηματισμό πλέγματος, αντιπροσωπεύοντας τις περιοχές κάλυψης. Κάθε σταθμός βάσης θα έχει καθορισμένη ακτίνα κάλυψης.
- Εξοπλισμός Χρήστη: Μοντελοποιήστε πολλούς UEs που να κινούνται τυχαία ή σε προκαθορισμένες διαδρομές εντός δικτύου. Κάθε UE έχει μοναδικό αναγνωριστικό, τοποθεσία, κι ενεργές συνεδρίες επικοινωνίας (κάθε μια μπορεί να έχει μια μεταβλητή που αντιπροσωπεύει το φόρτο στο δίκτυο).
- Ισχύς Σήματος (SNR): Κάθε σταθμός βάσης θα εκπέμπει σήματα προς τα UE εντός της κάλυψης του. Η ισχύς σήματος θα υπολογίζεται για κάθε UE ανάλογα με την απόστασή του από τον σταθμό βάσης, μοντελοποιώντας την αντίστροφη σχέση μεταξύ απόστασης και ισχύος σήματος (SNR).

Υλοποίηση Αλγορίθμων Εναλλαγής Κυψελίδων.

- Εναλλαγή Κυψελίδων με Ισχυρότερο Σήμα (βάσει RSSI): Τα UE θα παρακολουθούν την ισχύ σήματος των σταθμών βάσης και θα ενεργοποιούν την εναλλαγή όταν ένας γειτονικός σταθμός βάσης έχει ισχυρότερο σήμα από τον τρέχοντα.

4G-LTE Projects

- Εναλλαγή Κυψελίδων με Όριο (Threshold-based): Ορίστε μια οριακή τιμή για την ισχύ σήματος. Η εναλλαγή ενεργοποιείται όταν η ισχύς σήματος από τον τρέχοντα σταθμό βάσης πέσει κάτω από αυτό το όριο και ένας γειτονικός σταθμός βάσης προσφέρει ισχυρότερο σήμα.
- Εναλλαγή Κυψελίδων με Κόστος: Υλοποιήστε έναν πιο σύνθετο αλγόριθμο εναλλαγής κυψελίδων που λαμβάνει υπόψη παράγοντες όπως η εξισορρόπηση φόρτου, η ισχύς σήματος και η απόσταση.

Μοντελοποίηση Κινητικότητας UE και Εναλλαγής Κυψελίδων.

- Μοντέλα Κινητικότητας: Προσομοιώστε διάφορα πρότυπα κινητικότητας για τα UE.
- Τυχαία Κίνηση: Τα UE κινούνται τυχαία εντός του δικτύου.
- Γραμμική Κίνηση: Τα UE ακολουθούν μια ευθεία διαδρομή (π.χ. Προσομοίωση ενός οχήματος που κινείται κατά μήκος ενός δρόμου).
- Προσαρμοσμένες Διαδρομές (Προαιρετικό): Δημιουργήστε δίκες σας πιο πολύπλοκες γεννήτριες διαδρομών (μπορείτε να αξιοποιήσετε δεδομένα από κάποιο Dataset)
- Εκτέλεση Εναλλαγής Κυψελίδων: Όταν πληρούνται οι συνθήκες για την εναλλαγή (με βάση τον επιλεγμένο αλγόριθμο), προσομοιώστε τη διαδικασία εναλλαγής επανατοποθετώντας το UE στον νέο σταθμό βάσης. Κάθε νέα σύνδεση πρέπει να συνοδεύτε από ένα delay που προκύπτει από ένα σταθερό delay για την ανταλλαγή σημάτων και ένα μεταβλητό delay που εξαρτάται από το φορτίο του νέου σταθμού βάσης και την απόσταση του UE από τον σταθμό (ισχύς σήματος).

Προσομοίωση Συνθηκών Δικτύου.

- Μεταβαλλόμενος Φόρτος Δικτύου: Εισαγάγετε τυχαίες ή προκαθορισμένες μεταβολές στον φόρτο που διαχειρίζεται κάθε σταθμός βάσης (π.χ. Αριθμός συνδεδεμένων UE ή φόρτος κάθε συνεδρίας που εξυπηρετείται από τον σταθμό βάσης).
- Μεταβαλλόμενη Ισχύς Σήματος: Μεταβάλλετε τυχαία την ισχύ σήματος με την πάροδο του χρόνου για να προσομοιώσετε συνθήκες του πραγματικού κόσμου όπως εξασθένηση, σκίαση και εμπόδια.

Ανάλυση Απόδοσης.

- Κύριες Μετρήσεις: Κατά τη διάρκεια της προσομοίωσης, καταγράψτε μετρήσεις όπως:
- Καθυστέρηση Εναλλαγής: Ο χρόνος που απαιτείται για την εκτέλεση της εναλλαγής.
- Αριθμός Εναλλαγών: Ο συνολικός αριθμός των εναλλαγών που εκτελούνται από κάθε UE.
- Αποσυνδέσεις (Προαιρετικό): Καταμετρήστε τον αριθμό των αποσυνδεδεμένων κλήσεων ή συνεδριών δεδομένων κατά τη διάρκεια των εναλλαγών λόγω κακής ποιότητας σήματος ή υπερβολικής καθυστέρησης (απαιτεί να ορίσετε αρκετές ακόμα παραμέτρους στην προσομοίωση).
- Μεταβολή Σεναρίου: Προσομοιώστε σενάρια αλλάζοντας τον αριθμό των UE, την πυκνότητα των σταθμών βάσης, και τα πρότυπα κίνησης, και παρατηρήστε πώς μεταβάλλεται η απόδοση κάθε αλγορίθμου εναλλαγής.

Επιβλέπων: Γιάννης Κυριακόπουλος (i.kyriakopoulos@isi.gr).

5 Προσομοίωση δικτύου με NS-3

Η εργασία εστιάζει στη δημιουργία προσομοίωσης δικτύου χρησιμοποιώντας το NS-3. Η προσομοίωση χωρίζεται σε δύο μέρη, (1) στην προσομοίωση μιας απλής σύνδεσης σημείου-προς-σημείο μεταξύ δύο κόμβων με UDP echo, και (2) προσομοίωση ενός πιο πολύπλοκου δικτύου με δυναμική δρομολόγηση, δημιουργία ροών πακέτων, εισαγωγή αποτυχίας συνδέσεων, και ανάλυση απώλειας πακέτων.

- **Μέρος 1. Προσομοίωση Σημείου-Προς-Σημείο:** Δημιουργήστε μια προσομοίωση με 2 κόμβους που συνδέονται μέσω σύνδεσης point to point. Ορίστε τα χαρακτηριστικά της σύνδεσης (ταχύτητα μετάδοσης δεδομένων και καθυστέρηση) και αντιστοιχίστε διευθύνσεις ipn4 στις διεπαφές των κόμβων. Εγκαταστήστε δύο servers UDP echo στον κόμβο 1 σε διαφορετικές θύρες. Δημιουργήστε

4G-LTE Projects

δύο πελάτες UDP echo στον κόμβο 0. Εκτελέστε την προσομοίωση και καταγράψτε τα δεδομένα ροής (πακέτα, καθυστέρηση, εύρος ζώνης).

- **Μέρος 2. Προσομοίωση Δυναμικής Δρομολόγησης:** Δημιουργήστε ένα δίκτυο με 5 κόμβους με την παρακάτω τοπολογία:
 - Κόμβος 0 με Κόμβο 1
 - Κόμβος 1 με Κόμβο 2
 - Κόμβος 0 με Κόμβο 3
 - Κόμβος 3 με Κόμβο 4
 - Κόμβος 4 με Κόμβο 2

Ορίστε τις ταχύτητες μετάδοσης δεδομένων και τις καθυστερήσεις για κάθε σύνδεση. Η καθυστέρηση πρέπει να είναι ρυθμιζόμενη μέσω παραμέτρων από τη γραμμή εντολών. Εγκαταστήστε το internet stack και ορίστε τις διευθύνσεις IP για κάθε διεπαφή. Ενεργοποιήστε τη δυναμική δρομολόγηση και εισάγετε συμβάντα αποτυχίας συνδέσεων (π.χ., αποτυχία της σύνδεσης μεταξύ Κόμβου 0 και Κόμβου 1 στο 2o δευτερόλεπτο και επαναφορά στο 3o δευτερόλεπτο). Δημιουργήστε ροές πακέτων CBR (Constant Bit Rate) από τον Κόμβο 0 προς τον Κόμβο 2, αλλά και από τον Κόμβο 0 προς τον Κόμβο 3, με διαφορετικές ταχύτητες και χρόνους εκκίνησης. Εγκαταστήστε packet sinks στους κόμβους προορισμού για τη λήψη της κίνησης. Εφαρμόστε μια γρήγορη “επικοινωνία” μεταξύ των κόμβων για να ρυθμιστούν οι πίνακες διευθύνσεών τους, ώστε να μπορούν να επικοινωνούν πιο γρήγορα και χωρίς καθυστερήσεις κατά τη διάρκεια της κύριας προσομοίωσης.

Χρησιμοποιήστε την παρακολούθηση ροής (Flow Monitoring) για να καταγράψετε στατιστικά όπως μεταδοθέντα πακέτα, απωλεσθέντα πακέτα, εύρος ζώνης. Υπολογίστε το ποσοστό απώλειας πακέτων σε τακτά χρονικά διαστήματα και καταγράψτε τα αποτελέσματα σε αρχείο. Στο τέλος, συνοψίστε το συνολικό αριθμό πακέτων (μεταδοθέντα, απωλεσθέντα, ποσοστό απώλειας).

Σημειώσεις. Χρησιμοποιήστε ως γλώσσα εκτέλεσης τη C++ ή Python μαζί με τα Python bindings για το ns-3. Συνίσταται η χρήση του docker για την εγκατάσταση ενός container, μαζί με όλες τις απαραίτητες εξαρτήσεις για το NS-3.

Dockerfile για το container με ubuntu & NS-3:

```
FROM ubuntu:22.04
ENV DEBIAN_FRONTEND=noninteractive

RUN apt-get update && apt-get install -y \
    Software-properties-common \
    G++ \
    Python3 \
    Python3-pip \
    Git \
    Nano \
    Vim \
    Wget \
    Sudo \
    Net-tools \
    Iputils-ping \
    Cmake \
    Build-essential \
```

4G-LTE Projects

```
Pkg-config

#ns-3 dependencies
RUN apt-get install-y \
Libsqlite3-dev \
Libxml2-dev \
Libc6-dev \
Python3-setuptools \
Libgtk2.0-0 \
Qt5-default \
Gsl-bin \
Libgsl-dev \
Libgsl23 \
Libgslcblas0 \
Openmpi-bin \
Openmpi-common \
Openmpi-doc \
Libopenmpi-dev \
Autoconf \
Cvs \
Bzr \
Unrar \
Unrar-free \
Unar \
P7zip-full \
P7zip-rar \
Libgcrypt20-dev \
Clang

# GCC10
RUN add-apt-repository-y ppa:ubuntu-toolchain-r/test && \
Apt-get update && \
Apt-get install-y gcc-10 g++-10 && \
Update-alternatives--install /usr/bin/gcc gcc /usr/bin/gcc-10 10 && \
Update-alternatives--install /usr/bin/g++ g++ /usr/bin/g++-10 10

ENTRYPOINT ["/bin/bash"]
```

6 Προσομοίωση κινητού δικτύου με Python βάσει containers

Ο στόχος αυτής της εργασίας είναι να σχεδιαστεί και να εφαρμοστεί μια προσομοίωση ενός cellular network χρησιμοποιώντας Docker containers. Ζητείται έτσι να αναπτυχθεί η προσομοίωση συσκευών χρήστη και εφαρμογών με δυναμική κατανομή πόρων άλλα και δυναμικό placement των containers με μηχανισμούς παράδοσης (handover mechanism). Τέλος, ζητείται και μια οπτικοποίηση της τοπολογίας του δικτύου και η ανάλυση της απόδοσης του συστήματος.

- **Μέρος 1:** Προσομοίωση της υποδομής του δικτύου από containers, όπου το καθένα θα λειτουργεί ως συσκευή χρήστη που δημιουργεί συγκεκριμένους τύπους κίνησης. Δημιουργήστε εικόνες Docker για διαφορετικές εφαρμογές (π.χ. Streaming service κ.α.). Τα containers θα πρέπει να μιμούνται μία πραγματική κίνηση και χρήση του δικτύου. Δημιουργήστε σταθμούς βάσης ως δίκτυα Docker χρησιμοποιώντας Python και Docker SDK. Κάθε δίκτυο Docker θα αντιπροσωπεύει έναν μοναδικό σταθμό βάσης στο προσομοιωμένο δίκτυο.
- **Μέρος 2:** Αναπτύξτε ένα Python script που παρακολουθεί τα containers που είναι συνδεδεμένα σε κάθε base station. Εφαρμόστε μια πολιτική προγραμματισμού που εκχωρεί το εύρος ζώνης με βάση τις απαιτήσεις της εφαρμογής και το τρέχον φορτίο στο σταθμό βάσης. Ορίστε ένα τυχαίο μοντέλο κινητικότητας που αλλάζει τη «θέση» των container με την πάροδο του χρόνου. Υπολογίστε την απόσταση μεταξύ της τρέχουσας θέσης του container και του εκάστοτε σταθμού βάσης για να αποφασίσετε σε ποιο σταθμό βάσης θα συνδεθεί το container. Υπολογίστε την απόσταση για να προσδιορίσετε πότε ένα container πρέπει να μετακινηθεί σε νέο σταθμό βάσης. Σταματήστε το container, αποσυνδέστε από το τρέχον δίκτυο, και επανασυνδέστε το στο δίκτυο του πλησιέστερου σταθμού βάσης.

Hint: μπορείτε να αναπαραστήσετε τον εκάστοτε σταθμό βάσης ως ένα Python object που περιέχει πληροφορίες σχετικά με τους πόρους του (π.χ. εύρος ζώνης, αριθμό συνδεδεμένων container, κλπ) και τη θέση του (π.χ. σε 2D grid). Εάν θέλετε να προσθέσετε πολυπλοκότητα (bonus), μπορείτε να εισάγετε ένα απλό μοντέλο εξασθένησης της ισχύος του σήματος ή ακόμα να προσομοιώσετε handover failures.

- **Μέρος 3:** Ανάλυση απόδοσης και οπτικοποίηση. Συλλέξτε δεδομένα είτε μέσω του docker, είτε χρησιμοποιώντας άλλα εργαλεία συλλογής και οπτικοποίησης, για την απόδοση του δικτύου, συμπεριλαμβανομένων:
 - Χρήση πόρων και κίνηση δικτύου ανά σταθμό βάσης.
 - Αριθμός και ποσοστό επιτυχίας παραδόσεων (handover success rate).

Επιβλέπουσα: Ζωή Πάνου (zpanou@isi.gr).

7 Εντοπισμός παραβιάσεων SLAs και εξισορρόπηση φορτίου σε 4G

Σκοπός. Ο στόχος αυτής της εργασίας είναι η δημιουργία μιας προσομοίωσης δικτύου 4G-LTE, όπου οι φοιτητές καλούνται να εφαρμόσουν ένα rule-based σύστημα εντοπισμού παραβιάσεων SLAs (Service Level Agreements), αλλά και υλοποίηση ενός στοιχειώδους μηχανισμού εξισορρόπησης φορτίου (load balancing) για τη διαχείριση της ποιότητας υπηρεσίας (QoS). Η υλοποίηση μπορεί να αφορά σε Python, C, C++ (επιλέξτε ελεύθερα).

Μέρος 1: Η προσομοίωση του δικτύου. Στο 1^ο μέρος της εργασίας ζητείται η δημιουργία μιας χρονικά εξελισσόμενης προσομοίωσης ενός δικτύου 4G. Η προσομοίωση αυτή θα πρέπει να περιλαμβάνει τις παρακάτω παραμέτρους:

- Σταθμούς βάσης.
- Έναν τυχαίο αριθμό χρηστών που κατανέμονται στοχαστικά στους σταθμούς βάσης.

4G-LTE Projects

- Η χρονική προσομοίωση θα πρέπει να γίνεται με τρόπο που να κατανέμονται όλοι οι χρήστες στους σταθμούς, και σε κάθε βήμα να ενημερώνεται η τρέχουσα κατανομή, η χρήση πόρων ανά σταθμό, και διάφοροι δείκτες απόδοσης. Η δραστηριότητα των χρηστών θα πρέπει να μεταβάλλεται (π.χ. αλλαγή εύρους ζώνης).

Για κάθε σταθμό βάσης ανά χρονική στιγμή δημιουργούνται μετρήσεις:

- Αριθμός συνδεδεμένων χρηστών.
- Καθυστέρηση (latency σε ms).
- Μέσος ρυθμός μετάδοσης
- Προαιρετικά όποια άλλη μετρική θεωρείτε χρήσιμη

Σημείωση: Είναι απαραίτητο να ενσωματωθεί μια συνάρτηση τυχαίων γεγονότων, η οποία στο κάθε χρονικό βήμα θα μεταβάλει δυναμικά τη συμπεριφορά των χρηστών και των σταθμών βάσης, ώστε να προσομοιώνεται η απρόβλεπτη φύση ενός πραγματικού δικτύου. Η συνάρτηση αυτή θα πρέπει να αλλάζει τυχαία τις απαιτήσεις εύρους ζώνης των χρηστών, και να τροποποιεί την ευαισθησία τους σε καθυστέρηση και να προκαλεί έκτακτα γεγονότα, όπως μείωση της χωρητικότητας ενός σταθμού βάσης ή προσωρινή υποβάθμισή του. Οι αλλαγές αυτές επηρεάζουν την κατανομή πόρων, και δημιουργούν συνθήκες συμφόρησης ή παραβίασης SLAs και ενεργοποιούν τον μηχανισμό ανίχνευσης παραβιάσεων και εξισορρόπησης φορτίου, καθιστώντας την προσομοίωση ρεαλιστική και λειτουργικά απαιτητική.

Μέρος 2: Ορισμός κανόνων παραβίασης SLAs. Στην προσομοίωση οι παραβιάσεις εντοπίζονται βάσει συγκεκριμένων ορίων που ορίζονται για τις βασικές μετρικές ποιότητας υπηρεσίας (QoS). Οι φοιτητές καλούνται να καθορίσουν κατάλληλα κατώφλια για μετρικές, λαμβάνοντας υπόψη τις απαιτήσεις και τα χαρακτηριστικά του δικτύου 4G. Επίσης, πρέπει να εξεταστεί η επίδραση της υπερφόρτωσης στους σταθμούς βάσης, με βάση τον αριθμό χρηστών που εξυπηρετούνται. Οι παραβιάσεις που εντοπίζονται σε αυτές τις μετρικές θα λειτουργούν ως ενεργοποιητές για την εφαρμογή κανόνων ελέγχου SLAs, και πιθανών ενεργειών εξισορρόπησης φορτίου στη διάρκεια της προσομοίωσης. Αν κάποια τιμή υπερβεί τα όρια, θεωρείται παραβίαση SLA και καταγράφεται.

Μέρος 3: Λογική ισορροπίας φορτίου. Θα πρέπει να εφαρμοστεί μηχανισμός εξισορρόπησης φορτίου, που να ανταποκρίνεται δυναμικά στις συνθήκες του δικτύου. Πέρα από τον εντοπισμό σταθμών βάσης με χαμηλό φορτίο και την προσομοίωση μεταφοράς χρηστών από υπερβολικά φορτωμένους σταθμούς σε λιγότερο φορτωμένους, μπορούν να ενσωματωθούν και επιπλέον ενέργειες, όπως:

- Κατάταξη των χρηστών βάσει προτεραιότητας ή ευαισθησίας QoS, ώστε να μεταφέρονται πρώτα εκείνοι που επηρεάζονται περισσότερο.
- Προοδευτική μεταφορά χρηστών, όχι όμως όλων ταυτόχρονα, αλλά σταδιακά σε διαδοχικά χρονικά βήματα για αποφυγή νέας αστάθειας ή παραβίασης SLAs.
- Έλεγχος γειτνίασης σταθμών: να επιτρέπεται handover μόνο μεταξύ γειτονικών σταθμών βάσης, ώστε να διατηρείται ρεαλιστική γεωγραφική κατανομή.

Μέρος 4: Καταγραφή και οπτικοποίηση. Στην εκτέλεση της προσομοίωσης, είναι πάρα πολύ σημαντικό να διατηρείται αρχείο με τα γεγονότα που σχετίζονται με παραβιάσεις SLAs, και ενέργειες ώστε να είναι εφικτή η ανάλυση και οπτικοποίηση της προσομοίωσης μέσω διαγραμμάτων ή dashboards, με στόχο την αξιολόγηση της απόδοσης του συστήματος και την αποτύπωση των δυναμικών αλλαγών.

Επιβλέπουσα: Ζωή Πάνου (zpanou@isi.gr).

8 Έλεγχος χρησιμοποίησης πόρων σε δίκτυο με χρήση containers

Σκοπός. Ο στόχος αυτής της εργασίας είναι η κατανόηση βασικών αρχών λειτουργίας και διαχείρισης κινητών δικτύων 4G μέσω της προσομοίωσης διαφορετικών περιπτώσεων χρήσης σε containers. Οι

4G-LTE Projects

φοιτητές καλούνται να παρακολουθούν μετρικές συστήματος (όπως CPU και RAM), στοχεύοντας στην καλύτερη εξοικονόμηση πόρων, και να σχεδιάσουν μηχανισμό δυναμικής κατανομής πόρων, με σκοπό τη βελτιστοποίηση της ποιότητας υπηρεσίας (QoS) και τη σταθεροποίηση του συστήματος.

Μέρος 1: Η προσομοίωση του δικτύου. Ζητείται εδώ να υλοποιηθεί ένα σενάριο προσομοίωσης που να περιλαμβάνει τη δημιουργία 10–20 containers, καθένα από τα οποία θα αναπαριστά διαφορετικό τύπο υπηρεσίας σε ένα κινητό δίκτυο 4G-LTE. Τα containers θα εκτελούν διεργασίες με διαφορετικά χαρακτηριστικά κατανάλωσης πόρων, π.χ.

- Video streaming.
- Data processing.
- Voice call.
- Gaming.

Η υλοποίηση της δημιουργίας και κατανομής φορτίου μπορεί να βασιστεί σε εργαλεία όπως stressing, ή custom scripts που δημιουργούν το κατάλληλο φορτίο. Η συνολική διάρκεια της προσομοίωσης θα πρέπει να είναι επαρκής, ώστε να καταγραφούν αρκετά δεδομένα για την εξαγωγή συμπερασμάτων σχετικά με τη χρησιμοποίηση πόρων (CPU/RAM). Οι τιμές αυτές θα πρέπει να καταγράφονται διαρκώς ώστε να εντοπίζονται αποκλίσεις από τις προκαθορισμένες τιμές.

Μέρος 2: Ορισμός ορίων. Θα πρέπει να καθοριστούν κατάλληλα όρια για τις μετρικές CPU και RAM, βάσει των απαιτήσεων κάθε κατηγορίας χρήσης. Εάν οποιαδήποτε μετρική υπερβεί το καθορισμένο όριο για ένα container, θεωρείται ότι υπάρχει παραβίαση και πρέπει να καταγραφεί. Οι παραβιάσεις αυτές θα ενεργοποιούν το μηχανισμό δυναμικής διαχείρισης πόρων, ο οποίος πρόκειται να υλοποιηθεί στο επόμενο μέρος.

Μέρος 3: Κατανομή πόρων. Όταν εντοπίζεται παραβίαση σε κάποιο container, θα πρέπει να καλείται μια συνάρτηση που να επιχειρεί να βελτιώσει την κατανομή των πόρων, έτσι ώστε να αποκατασταθεί η ποιότητα υπηρεσίας. Η συνάρτηση αυτή μπορεί να περιλαμβάνει.

- Ανακατανομή φορτίου.
- Δυναμική δημιουργία ή τερματισμό container / processes.
- Εκκαθάριση ή απελευθέρωση μνήμης.
- Άλλες βελτιστοποιήσεις που μπορεί να σκεφτείτε.

Η εκάστοτε ενέργεια που γίνεται στο πλαίσιο της βελτίωσης κατανομής, θα πρέπει να καταγράφεται με χρονική σήμανση (timestamp). Η κατανομή πόρων αφορά τη δυναμική διαχείριση των διαθέσιμων CPU & RAM για κάθε container, με σκοπό τη διατήρηση της ποιότητας υπηρεσίας (QoS). Αν ανιχνεύεται παραβίαση κάποιου ορίου (π.χ., υπέρβαση της CPU > 80%), ενεργοποιείται αντίστοιχα μηχανισμός που αναλαμβάνει την ανακατανομή πόρων, είτε αυξάνοντας τους διαθέσιμους πόρους στο προβληματικό container (docker update), είτε μειώνοντάς τους από άλλα- λιγότερο ενεργά- containers, είτε, τέλος, δημιουργώντας ή τερματίζοντας containers αναλόγως του φορτίου. Όλες οι ενέργειες αυτές πρέπει να καταγράφονται σε log αρχείο, με χρονική σήμανση για σκοπούς παρακολούθησης και αξιολόγησης.

Μέρος 4: Καταγραφή και οπτικοποίηση. Στο μέρος αυτό πρέπει να υλοποιηθεί σύστημα καταγραφής όλων των σημαντικών γεγονότων σε αρχείο log (CSV ή TXT), ώστε να είναι δυνατή η μετέπειτα ανάλυση και οπτικοποίηση. Κάθε εγγραφή πρέπει να περιλαμβάνει.

- Χρονική στιγμή.
- Είδος γεγονότος (violation ή resource adjustment).
- Όνομα container.
- Τιμή μετρικής.

Οπτικοποίηση. Σκεφτείτε χρήσιμες γραφικές παραστάσεις που θα απέδιδαν όσο το δυνατόν καλύτερα το utilization και τη κατανομή πόρων.

4G-LTE Projects

Σημείωση. Η ανάκτηση των στατιστικών από το docker δεν παρέχει άμεσα το % χρήσης CPU. Αντίθετα, επιστρέφει ακατέργαστες μετρικές χρήσης CPU, και πρέπει να υλοποιηθεί κάποια ειδική συνάρτηση υπολογισμού του ποσοστού. Η συνάρτηση αυτή θα πρέπει να καλείται κατά την παρακολούθηση κάθε container, ώστε να γίνεται σωστός έλεγχος και να καταγράφονται έγκυρα δεδομένα στα αρχεία log. Σαν εναλλακτική μπορεί να χρησιμοποιηθεί το Prometheus.

Επιβλέπουσα: Ζωή Πάνου (zpanou@isi.gr).

9 Θεμελιώδεις αρχές και κατασκευή ενός συστήματος OFDM

Εισαγωγή. Το Orthogonal Frequency Division Multiplexing (OFDM) είναι μια βασική τεχνολογία φυσικού επιπέδου (physical layer), η οποία θεμελιώνει τα σύγχρονα πρότυπα ασύρματης επικοινωνίας υψηλής ταχύτητας, όπως 4G/LTE, 5G, και Wi-Fi. Το παρόν project ζητά από τους φοιτητές να σχεδιάσουν ένα πλήρες σύστημα επικοινωνίας από το μηδέν (from scratch). Με την ανάπτυξη και την προσομοίωση του κάθε υποσυστήματος θα ανακαλύψουν όχι μόνο τι είναι το OFDM, αλλά και γιατί η συγκεκριμένη αρχιτεκτονική είναι απαραίτητη για αξιόπιστη επικοινωνία σε ρεαλιστικά ασύρματα περιβάλλοντα. Επιτρέπεται η χρήση Python, numpy, scipy, και όλων των basic libraries. **Οτιδήποτε άλλο πρέπει πρώτα να συζητηθεί με τον επιβλέποντα.**

Μεθοδολογία και ενδιάμεσοι στόχοι. Το project χωρίζεται σε έξι (6) βήματα υλοποίησης, και μια τελική φάση ανάλυσης και περιγραφής. Σε ότι ακολουθεί, οποιαδήποτε τιμή παραμέτρου ορίζετε, θα πρέπει να την αιτιολογείτε.

- **Βήμα 1:** Θα υλοποιήσετε ένα απλό single-carrier σύστημα με χρήση QPSK modulation, και ύστερα θα αποδείξετε την αποτυχία του σε ένα προσομοιωμένο multipath κανάλι με AWGN θόρυβο. Ο στόχος είναι να παραχθούν σαφείς αποδείξεις του inter-symbol interference με παραμορφωμένα constellation diagrams, και υψηλό bit error rate (BER).
- **Βήμα 2:** Θα κατασκευάσετε τη βασική αρχιτεκτονική OFDM χρησιμοποιώντας μετασχηματισμούς IFFT/FFT για τη διαχείριση παράλληλων data streams. Ο στόχος εδώ είναι να επιβεβαιωθεί η τέλεια λειτουργία του σε ιδανικό κανάλι, αποδεικνύοντας την εγκυρότητα της βασικής ιδέας.
- **Βήμα 3:** Επανεισάγοντας το multipath κανάλι με AWGN θόρυβο, αποκαλύπτεται η ευπάθεια του OFDM σε inter-carrier interference. Ο στόχος είναι η υλοποίηση και η ανάλυση μιας διαισθητικής μα αναποτελεσματικής λύσης- frequency guard bands- και η μελέτη του trade-off μεταξύ κέρδους απόδοσης και απώλειας φασματικής αποδοτικότητας.
- **Βήμα 4:** Υλοποιήστε την καθιερωμένη λύση: το cyclic prefix. Αποδείξτε την αποτελεσματικότητα του σε σχέση με το προηγούμενο βήμα.
- **Βήμα 5:** Περιγράψτε τις επιπλέον αλλαγές που πρέπει να γίνουν στο υπάρχον σύστημα, ώστε να μετατραπεί σε OFDMA. Η περιγραφή σας πρέπει να αποσαφηνίζει τις αλλαγές στο subcarrier mapping του transmitter, και στη διαδικασία subcarrier de-mapping του receiver, ώστε να υποστηρίζεται ταυτόχρονη μετάδοση σε πολλούς χρήστες.
- **Βήμα 6:** Μεταδώστε το “hello world” σε binary μορφή και άλλες φράσεις της επιλογής σας.

Επιβλέπων: Κυριάκος Στρατάκος (kstratakos@isi.gr).

10 Ανάλυση και προσομοίωση αλγόριθμων κατανομής πόρων

Εισαγωγή. Η κατανομή πόρων (resource scheduling) αποτελεί ένα σημαντικό στοιχείο των προηγμένων τηλεπικοινωνιακών συστημάτων (4G-LTE, 5G). Οι αλγόριθμοι scheduling είναι κρίσιμοι για την εύρυθμη

4G-LTE Projects

λειτουργία του δικτύου. Το παρόν project ζητά από τους φοιτητές να μελετήσουν βασικές στρατηγικές scheduling, να αποκαλύψουν τις αδυναμίες συστημάτων και αλγορίθμων και να εφαρμόσουν γνωστούς αλγορίθμους βελτιστοποίησης, συνδέοντας θεωρία μα και πρακτική προσομοίωση. Επιτρέπεται χρήση Python, numpy, scipy, matplotlib, pandas, ή οτιδήποτε άλλο μετά από επικοινωνία με τον επιβλέποντα. Ενθαρρύνεται η υποβολή καλά τεκμηριωμένου και reproducible κώδικα (seeds για την τυχαιότητα).

Μεθοδολογία και ενδιάμεσοι στόχοι. Το project χωρίζεται σε τρία (3) μέρη υλοποίησης, ενώ υπάρχουν και ερωτήσεις ανάλυσης και κατανόησης του προβλήματος, και αξιολογείται θετικά η όσο το δυνατό πιο ρεαλιστική αναπαράσταση των μεταβλητών και παραμέτρων.

- **Μέρος 1:** Θεωρήστε Ν χρήστες, K resource blocks και πίνακα $C_{N \times K}$, όπου κάθε στοιχείο του C_{ij} είναι τυχαία μεταβλητή που αντιπροσωπεύει την ποιότητα του καναλιού j για τον χρήστη i. Θεωρούμε πως το επιτεύχιμο data rate για το κάθε resource block είναι ανάλογο της ποιότητας του καναλιού. Υλοποίήστε έναν αλγόριθμο που να μεγιστοποιεί το στιγμαίο συνολικό data rate (Max C/I), και που να έχει μέγιστη πολυπλοκότητα της τάξης του $O(N \cdot K)$. Δείξτε με γραφήματα το πώς κατανέμονται οι πόροι μεταξύ χρηστών.
- **Μέρος 2:** Επεκτείνετε το πρόβλημα στο χρόνο, όπου ο πίνακας C αλλάζει σε κάθε timestep. Ορίστε πίνακα ζητούμενου data rate για κάθε χρήστη Q, που να αλλάζει κι αυτός σε κάθε timestep. Κάθε στοιχείο τους ακολουθεί κανονική κατανομή γύρω από τυχαία μέση τιμή. Τρέξτε προσομοίωση για $T = 1000$ TTIs, εφαρμόζοντας τον αλγόριθμο του μέρους 1. Καταγράψτε και απεικονίστε την εξέλιξη ουρών, το στιγμαίο και μέσο throughput, και το Jain's Fairness Index. Τι πρόβλημα παρατηρείτε; Δώστε ποσοτική και γραφική αιτιολόγηση.
- **Μέρος 3:** Ορίστε μια συνάρτηση για το ιστορικό μέσο throughput των χρηστών με παράμετρο a (forgetting factor) R_i .
 - Λύστε το πρόβλημα με τον Proportional Fair (PF) αλγόριθμο.
 - Εξηγήστε το ρόλο της παραμέτρου a. Πως αυτή συνδέεται με το συγκεκριμένο πρόβλημα; Ποια είναι η σχέση της με τα coding schemes QPSK, N-QAM; Γιατί δε συμφέρει να παίρνει πάντα την μέγιστη τιμή;
 - Εξηγήστε ποια είναι η ποσοτική και ποιοτική διαφορά του αποτελέσματος σε σχέση με το προηγούμενο ερώτημα.

Επιβλέπων: Κυριάκος Στρατάκος (kstratakos@isi.gr).

11 Βασικές έννοιες και προσαρμοστική διάδοση MIMO

Εισαγωγή. Το Multiple-Input Multiple-Output (MIMO) είναι μια θεμελιώδης τεχνολογία των σύγχρονων δικτύων 4G/LTE και 5G, που αξιοποιεί πολλαπλές κεραίες στον πομπό και στο δέκτη για να μετατρέψει το multipath propagation σε πλεονέκτημα. Το εν λόγω project στοχεύει και καθοδηγεί τους φοιτητές σε κατασκευή προσομοίωσης σε Python, για την ποσοτική και ποιοτική ανάλυση των πλεονεκτημάτων αυτής της τεχνολογίας σε σχέση με την προγενέστερη τεχνολογία, Single-Input Single-Output (SISO). Στο τέλος, οι φοιτητές θα είναι εξοικειωμένοι με την τεχνολογία MIMO και τις θεμελιώδεις αρχές της. Επιτρέπεται η χρήση Python, numpy, scipy και όλων των basic libraries. **Οτιδήποτε άλλο πρέπει πρώτα να συζητηθεί με τον επιβλέποντα.**

Μεθοδολογία και ενδιάμεσοι στόχοι.

- **Βήμα 1:** Μοντελοποίηση ενός καναλιού μετάδοσης: παράγετε μια συνάρτηση που προσομοιώνει το ασύρματο μέσο με τη χρήση του Rayleigh fading model. Η συνάρτηση θα πρέπει να παράγει ένα μιγαδικό πίνακα καναλιού (channel matrix) H (1x1 για SISO, 2x2 για MIMO 2x2, κλπ.). Υλοποίήστε το Shannon capacity formula. Θεωρήστε πως η συνολική ισχύς εκπομπής είναι ίδια σε όλα τα υπό μελέτη συστήματα. Παρουσιάστε την χωρητικότητα καναλιού (bps/Hz) ως συνάρτηση του signal-to-noise ratio (SNR) για τα ακόλουθα συστήματα: 1x1 SISO, 2x2 MIMO, και 4x4 MIMO.

4G-LTE Projects

- **Βήμα 2:** Μοντελοποίηση ενός κινούμενου χρήστη: παράγετε timeseries simulation του κινούμενου χρήστη στο χώρο (2 sec). Σε κάθε timeframe υπολογίστε τον H, ώστε να αναπαρασταθεί η κίνηση του χρήστη. Για δεδομένο SNR παρουσιάστε κατάλληλα διαγράμματα στιγμιαίου throughput/t, για τις περιπτώσεις SISO και 2x2 MIMO. Αναλύστε το reliability των 2 συστημάτων. Βρείτε το ποσοστό του χρόνου που το throughput πέφτει κάτω από ένα minimum required level. Ακόμη, επιλέξτε το επιθυμητό Mbps level ώστε να φαίνεται η διαφορά των 2 συστημάτων “diversity gain”.
- **Βήμα 3:** Προσαρμοστική διάδοση MIMO: τα προηγμένα συστήματα μετάδοσης εκμεταλλεύονται πληροφορία που έχουν για το κανάλι διάδοσης H, Channel State Information (CSI) για τη βελτίωση της μετάδοσης. Αυτή η πληροφορία παράγεται από τον χρήστη κάθε στιγμή. Ο πομπός μπορεί να χρησιμοποιήσει αυτή την πληροφορία, ώστε να διαμοιράσει το περίπλοκο MIMO κανάλι σε κάποια παράλληλα μικρότερα απλά υπο-κανάλια, με τη χρήση του Singular Value Decomposition (SVD). Φτιάξτε μια συνάρτηση η οποία παίρνει σαν δεδομένο τον πίνακα H, κάνει SVD, και βρίσκει τη συνολική χωρητικότητα ως άθροισμα των χωρητικοτήτων όλων των υπο-καναλιών. Προσθέστε στο διάγραμμα χωρητικότητας καναλιού του βήματος 1 την περίπτωση 4x4 MIMO with SVD Precoding. Σχολιάστε το αποτέλεσμα. Μπορεί κάποια εναλλακτική μέθοδος MIMO να παρουσιάσει καλύτερα αποτελέσματα;

Επιβλέπων: Κυριάκος Στρατάκος (kstratakos@isi.gr).

12 Κάλυψη, ανταγωνισμός, και λήψη αποφάσεων σε δίκτυο 4G

Εισαγωγή. Η δημιουργία ενός σύγχρονου δικτύου 4G LTE είναι ένα στοίχημα υψηλού ρίσκου, που εκτός από μηχανική χρειάζεται και στρατηγικό επιχειρηματικό σχεδιασμό. Η επιτυχία του εξαρτάται από την ικανότητα μιας εταιρείας να αναπτύσσει έξυπνα τους πόρους της, με κρίσιμη απόφαση την τοποθεσία εγκατάστασης των σταθμών βάσης (BSs- eNodeBs). Αυτό το project προσομοιώνει σε ανταγωνιστικό περιβάλλον τη δημιουργία ενός δικτύου σε μια πόλη. Αρχικά, θα δράσετε ως ένας ενιαίος, αποδοτικός σχεδιαστής. Στη συνέχεια, θα χωριστείτε σε δύο αντίπαλες εταιρείες, εξοπλισμένες με budget αλλά και τεχνολογικές επιλογές, ώστε να πολεμήσετε για το μερίδιο της αγοράς σε ένα παιχνίδι τοποθέτησης BSs βασισμένο σε γύρους. Στα παρακάτω πρέπει να υπάρχει αναλυτική αιτιολόγηση των στρατηγικών που ακολουθήθηκαν, και η περιγραφή των αποτελεσμάτων κάτω από διάφορα περιβάλλοντα. Το simulation και οι αλγόριθμοι θα πρέπει να είναι recreateable, οπότε χρησιμοποιήστε ένα array από seeds όπου χρησιμοποιείτε κατανομές και πιθανότητες. **Απαγορεύεται η χρήση μεθόδων ενισχυτικής ή μηχανικής μάθησης.** Η υλοποίηση του θέματος προτείνεται να γίνει σε Python.

Μεθοδολογία και Ενδιάμεσοι Στόχοι.

- **Βήμα 1:** Ως ομάδα, είστε σχεδιαστές σε μια εταιρεία με αποκλειστική άδεια κάλυψης μιας νέας αστικής περιοχής, η οποία αναπαρίσταται ως ένα πλέγμα (grid) 100x100 units. Στόχος σας είναι να παρέχετε κάλυψη σε 1000 πιθανούς πελάτες (UEs), χρησιμοποιώντας τον ελάχιστο δυνατό αριθμό eNodeBs.
 - Δημιουργία πόλης: σχεδιάστε τις τοποθεσίες 1000 UEs με ένα μείγμα ομαδοποιημένων (4 συστάδες των 150 χρηστών) και ομοιόμορφων (400 χρήστες) κατανομών, αναπαριστώντας με ρεαλιστικό τρόπο έναν αστικό ιστό.
 - Εύρεση βέλτιστων τοποθεσιών: Χρησιμοποιήστε τον αλγόριθμο clustering k-Means για να βρείτε ιδανικές τοποθεσίες για τα eNodeBs (κεντροειδή των clusters). Πειραματιστείτε με διάφορους αριθμούς k και διάφορα instances των distributions για να βρείτε το μικρότερο k που απαιτείται για να καλυφθεί έστω το 95% των UEs. Υποθέστε ότι τυπικά η ακτίνα ενός eNodeB είναι 15 μονάδες.
- **Βήμα 2:** Κάθε φοιτητής είναι υπεύθυνος για τη στρατηγική τοποθέτηση eNodeB δύο εταιριών που ανταγωνίζονται μεταξύ τους σε turn based περιβάλλον. Στο χάρτη από το προηγούμενο ερώτημα

4G-LTE Projects

Θα ανταγωνιστείτε για το ποιος θα καλύψει τους περισσότερους πελάτες. Θεωρήστε ότι ο πελάτης θα επιλέξει τον πάροχο που έχει το κοντινότερο eNodeB στην τοποθεσία του. Η κάθε εταιρία έχει budget 1200K ευρώ, και μπορεί να επιλέξει να τοποθετήσει σε κάθε γύρο ένα απ' τα ακόλουθα:

- Micro Cell: κόστος 150 K | ακτίνα 10 μ.μ. | μέγιστη εξυπηρέτηση 100 πελάτες.
- Standard Cell: κόστος 250 K | ακτίνα 17 μ.μ. | μέγιστη εξυπηρέτηση 240 πελάτες.
- High Power Cell: κόστος 500 K | ακτίνα 25 μ.μ. | μέγιστη εξυπηρέτηση 500 πελάτες.

Ο φοιτητής καλείται να γράψει έναν turn based αλγόριθμο που θα επιλέγει την επόμενη τοποθεσία και τύπου του eNodeB της εταιρείας του με βάση την τρέχουσα κατάσταση του χάρτη και το budget του. Τρέξτε την προσομοίωση 100 φορές και παρουσιάστε τα αποτελέσματα των αλγορίθμων σας. Αξιολογείται η ευρηματικότητα και η στρατηγική του αλγόριθμου, ενώ η νίκη δίνει bragging rights.

Επιβλέπων: Κυριάκος Στρατάκος (kstratakos@isi.gr).

13 Προσομοίωση event-driven δικτύου σε Python (και SimPy)

Αυτή η εργασία αφορά τη μοντελοποίηση και προσομοίωση λειτουργίας ενός απλού δικτύου με χρήση της γλώσσας προγραμματισμού Python. Αφού ολοκληρωθεί η μοντελοποίηση, καλείστε να συλλέξετε και να απεικονίσετε διάφορες μετρικές του δικτύου, όπως είναι η καθυστέρηση πακέτων, packet drops, throughput, κ.α. Για την εκτέλεση της εργασίας συνιστάται η βιβλιοθήκη SimPy (προτείνεται ενδεικτικά, όμως μπορείτε να χρησιμοποιήσετε όποια βιβλιοθήκη προσομοίωσης δικτύων της Python επιθυμείτε).

<https://simpy.readthedocs.io/en/latest/>

Η απλή δομή τους δικτύου μπορεί για παράδειγμα να περιέχει δύο hosts (source host και destination host), και ένα router το οποίο θα διασυνδέει τους hosts και θα επιτρέπει την μεταφορά πακέτων. Όλες οι συνδέσεις μεταξύ των hosts και του router ορίζονται από συγκεκριμένο bandwidth, καθώς και από την καθυστέρηση διάδοσης (propagation delay) για κάθε σύνδεση.

1. **Βασική μοντελοποίηση και προσομοίωση:** Χρησιμοποιείστε την Python (μαζί με τη SimPy), ώστε να ορίσετε το δίκτυο. Συγκεκριμένα, θα χρειαστεί να ορίσετε τους κόμβους (nodes), όπως host και router, και τις συνδέσεις (links) μεταξύ τους. Θα ορίσετε, επίσης, τα bandwidth, και καθυστέρηση διάδοσης (propagation delay) για κάθε link.
2. **Μεταφορά πακέτων:** Προσομοιώστε με την Python πακέτα που μεταφέρονται μέσω του δικτύου. Θα πρέπει να ορίσετε το μέγεθος των πακέτων (π.χ. σε bytes), μα και την καθυστέρηση μεταφοράς (transmission delay), και να υπολογίσετε το συνολικό χρόνο που το κάθε πακέτο χρειάζεται για να φτάσει στον τελικό στόχο (π.χ. destination host) δηλαδή το συνολικό end-to-end delay. Ισχύει ότι:

$$\text{Transmission Delay} = \frac{\text{Packet Size}}{\text{Bandwidth}}.$$

Προσοχή στο να είναι κατάλληλες οι μονάδες των μεγεθών (π.χ. το packet size μετρημένο σε bytes και το bandwidth μετρημένο σε bytes/sec, αντίστοιχα).

3. **Προσομοίωση ουράς (queue) στον router:** Προσθέστε μια ουρά στο router, όπου υπάρχει αναμονή των πακέτων. Το μέγεθος της ουράς θα είναι περιορισμένο (π.χ. 10 πακέτα), και αν και η ουρά είναι γεμάτη θα έχουμε packet drop.
4. **Συλλογή και απεικόνιση μετρικών:** Προσομοιώστε την μεταφορά πολλαπλών πακέτων (π.χ. n=2000 πακέτα) και συλλέξτε end-to-end delay για κάθε πακέτο, throughput (συνολικό ποσό δεδομένων σε bits / χρόνος που χρειάστηκε για να μεταφερθούν), packet loss rate (ποσοστό των πακέτων που έγιναν drop εξαιτίας υπερχείλισης της ουράς), και το queue size (μέγεθος ουράς). Αποτυπώστε γραφικά αυτές τις μετρικές ως προς τον χρόνο.
5. **Συμφόρηση δικτύου (network congestion) – συσχέτιση bandwidth και καθυστέρησης:** Εκτελέστε την προσομοίωση για διαφορετικές τιμές του bandwidth (π.χ. 1 Mbps, 10 Mbps, κι 100 Mbps).

4G-LTE Projects

Υπολογίστε και απεικονίστε γραφικά τις παραπάνω μετρικές του ερωτ. 4 για τις διάφορες τιμές του bandwidth. Παρατηρείτε μείωση της απόδοσης του δικτύου και συμφόρηση ; Εξηγήστε.

6. **Μέγεθος πακέτου:** Εκτελέστε την ίδια διαδικασία στέλνοντας τον ίδιο αριθμό πακέτων αλλά με διαφορετικό μέγεθος πακέτων (π.χ. 100 bytes, 1000 bytes, 3000 bytes, 5000 bytes κ.λπ.). Δείξτε και εξηγήστε αντίστοιχα και με γραφικές παραστάσεις πώς επηρεάζονται οι μετρικές του δικτύου ανάλογα με το μέγεθος των πακέτων.
7. **Επέκταση του δικτύου:** Επεκτείνετε το δίκτυο με περισσότερους hosts και routers. Εκτελέστε τα παραπάνω ερωτήματα για το νέο δίκτυο και εξηγήστε πώς η πολυπλοκότητα του νέου δικτύου επηρεάζει τις επιδόσεις και τις μετρικές σε σχέση με τον χρόνο και σε σύγκριση με το προηγούμενο δίκτυο, όταν θέλουμε να στείλουμε τον ίδιο όγκο πληροφορίας μέσω πακέτων.
8. **Packet Scheduling Algorithms:** Εφαρμόστε δύο διαφορετικούς αλγόριθμους scheduling πακέτων στην προσομοίωση σας: First-Come-First-Serve (FCFS), και Round-Robin (RR). Για την προσομοίωση χρησιμοποιείστε ένα από τα δίκτυα που έχετε ήδη μοντελοποιήσει, με συνδυασμό των μικρών και μεγάλων πακέτων που φτάνουν με ποικίλους ρυθμούς. Αναλύστε πώς κάθε αλγόριθμος scheduling επηρεάζει τις μετρικές όπως η καθυστέρηση πακέτων (delay), και το throughput. Εξηγήστε επίσης ποιος αλγόριθμος αποδίδει καλύτερα κάτω από διαφορετικά φορτία κυκλοφορίας (π.χ. ελαφρύ-μικρά πακέτα, μέτριο-μεσαία πακέτα, βαρύ-μεγάλα πακέτα).

Επιβλέπουσα: Πουλχερία Ζερβού (pzervou@isi.gr).

14 Προσομοίωση κυκλοφορίας δικτύου σε Python (και SimPy)

Αυτή η εργασία αφορά τη μοντελοποίηση και την προσομοίωση της λειτουργίας ενός δικτύου κινητών επικοινωνιών με χρήση της γλώσσας προγραμματισμού Python. Βασικό στοιχείο της εργασίας είναι η αλληλεπίδραση μεταξύ των σταθμών βάσης (Base Stations- BS) και των συσκευών των χρηστών (User Equipment- UE). Μέσω της προσομοίωσης των BS-UE μα και προσομοίωσης της αποστολής πακέτων-πληροφορίας μεταξύ τους, καλείστε να συλλέξετε και να απεικονίσετε διάφορες μετρικές του δικτύου. Για την εκτέλεση της εργασίας συνιστάται η βιβλιοθήκη SimPy (προτείνεται ενδεικτικά, όμως μπορείτε να χρησιμοποιήσετε όποια βιβλιοθήκη προσομοίωσης δικτύων της Python επιθυμείτε).

<https://simpy.readthedocs.io/en/latest/>.

1. **Βασική μοντελοποίηση του χώρου:** Χρησιμοποιείστε την Python για να ορίσετε τους χρήστες και τους σταθμούς βάσης σε έναν ορισμένο χώρο. Οι χρήστες θα κινούνται εντός του ορισμένου χώρου και δεν θα μπορούν να κινηθούν εκτός αυτού. Η κίνηση για κάθε UE θα γίνεται με τυχαίο τρόπο που θα επιλέξετε εσείς (π.χ. 1 βήμα προς τυχαία κατεύθυνση σε κάθε χρονική στιγμή, 1 χρονική στιγμή ακινησία και ύστερα 1 βήμα κίνηση, κοκ.). Οι σταθμοί βάσης προφανώς θα είναι ακίνητοι. Η ισχύς του σήματος από τον σταθμό βάσης προς το UE θα ορίζεται ως:

$$\text{Signal Strength} = \frac{1}{1 + \text{distance}^2}.$$

Σε κάθε χρονική στιγμή της προσομοίωσης, κάθε UE θα πρέπει να είναι συνδεδεμένο στον BS από τον οποίο λαμβάνει το ισχυρότερο σήμα. Μετά από κάθε χρονική στιγμή/βήμα θα ελέγχετε την ισχύ για κάθε UE και αν δέχεται ισχυρότερο σήμα από κάποιο άλλο BS, θα πρέπει να συνδεθεί με αυτό το BS (διαδικασία handover). Στην μοντελοποίηση και στην προσομοίωσή σας θα πρέπει να περιέχονται 3 BSs και 10 UEs.

2. **Απεικόνιση προσομοίωσης:** Χρησιμοποιείστε μία απλή διαδικασία απεικόνισης (για παράδειγμα με την βιβλιοθήκη Matplotlib), όπου για ένα επιλεγμένο χρόνο προσομοίωσης (π.χ. 100 seconds = 100 timesteps) θα φαίνεται η κίνηση των UEs στον χώρο, αλλά και με ποιο BS είναι συνδεδεμένα (π.χ. BS και UE να συνδέονται με μία γραμμή και κάθε UE μπορεί να απεικονίζεται με το ίδιο χρώμα με το BS στο οποίο είναι συνδεδεμένο κάθε timestep).

4G-LTE Projects

3. **Κίνηση Δεδομένων- Data Traffic:** Κάθε χρήστης θα πρέπει να δημιουργεί μια κίνηση δεδομένων σε χρονικά βήματα της επιλογής σας. Υποθέστε ότι το εκάστοτε UE δημιουργεί κίνηση συγκεκριμένης κατανομής επιλογής σας (π.χ. τυχαία κατανομή, κανονική κατανομή, κατανομή Poisson κλπ.), που αντιπροσωπεύει τον αριθμό των πακέτων δεδομένων ανά χρονικό βήμα. Επιλέξτε επίσης ελεύθερα το μέγεθος του κάθε πακέτου (π.χ. σταθερό μέγεθος). Ορίστε συγκεκριμένο και σταθερό εύρος ζώνης (Bandwidth) για κάθε BS και χωρίστε το με ίσο τρόπο σε όλους τους συνδεδεμένους χρήστες κάθε timestep. Για κάθε χρήστη υπολογίστε τον χρόνο αποστολής (latency) με βάση τα πακέτα που έχει στείλει και το διαθέσιμο bandwidth ως εξής:

$$\text{Latency (s)} = \frac{\text{Packet Size (bits)}}{\text{Assigned Bandwidth} (\frac{\text{bits}}{\text{s}})}.$$

Για κάθε χρήστη υπολογίστε το Throughput ως εξής:

$$\text{Throughput} = \frac{\text{Total Data Transmitted (bits)}}{\text{Time (s)}}.$$

Απεικονίστε γραφικά τα latency και throughput ως προς τον χρόνο.

4. **Περιορισμένο bandwidth και Packet Loss:** Επιλέξτε κατάλληλες τιμές bandwidth και packet size ώστε να υπάρχουν χρονικές στιγμές όπου το bandwidth ενός ή περισσότερων BS δεν επαρκεί για μεταφορά όλων των πακέτων. Θεωρήστε ότι όταν συμβαίνει αυτό τα πακέτα που στέλνονται εκείνη τη στιγμή χάνονται (packet loss).
5. **Γραφική απεικόνιση:** Υπολογίστε και απεικονίστε γραφικά σε σχέση με τον χρόνο το Throughput, Latency και τον αριθμό χαμένων πακέτων για κάθε χρήστη. Επίσης υπολογίστε και απεικονίστε σε σχέση με τον χρόνο, το bandwidth κάθε BS ώστε να φαίνεται πόσο κοντά είναι στο να εξαντληθεί το εύρος ζώνης του.
6. **Δυναμική κατανομή Bandwidth:** Κατανέμετε με μέθοδο της επιλογής σας το Bandwidth κάθε BS στους UEs που εξυπηρετεί ώστε να πετύχετε καλύτερη εξυπηρέτηση χρηστών και μεταφορά πακέτων (π.χ. κατανομή με βάση ποιος χρήστης έχει στείλει τα περισσότερα ή τα μεγαλύτερα πακέτα). Προσομοιώστε το σενάριο του προηγούμενου ερωτήματος χρησιμοποιώντας την νέα μέθοδο κατανομής εύρους ζώνης και απεικονίστε τις ίδιες μετρικές. Βάσει των αποτελεσμάτων εξηγήστε αν και γιατί υπάρχει βελτίωση στην απόδοση του δικτύου.

Επιβλέπουσα: Πουλχερία Ζερβού (pzervou@isi.gr).

15 Προσομοίωση δικτύου 4G κι εφαρμογή scheduling (PF, MT)

Η παρούσα εργασία έχει ως στόχο την ανάπτυξη μιας προσομοίωσης ενός δικτύου 4G με Python. Θα χρειαστεί να προσομοιώσετε την συμπεριφορά του δικτύου υπό διάφορες συνθήκες φορτίου και να αξιολογήσετε την απόδοση του με βάση διάφορες μετρικές απόδοσης.

Προσομοίωση Δικτύου 4G: Αναπτύξτε μια προσομοίωση ενός δικτύου 4G χρησιμοποιώντας Python, το οποίο περιλαμβάνει ένα σταθμό βάσης (BS) και χρήστες (UE). Κάθε UE έχει μοναδικά χαρακτηριστικά, όπως αναγνωριστικό (ID), διαφορετικές απαιτήσεις κίνησης, τυχαίους δείκτες ποιότητας καναλιού (CQI), μα και διαφορετικά επίπεδα προτεραιότητας. Απαιτείται η υλοποίηση μοντέλου κινητικότητας, όπου τα UEs μετακινούνται τυχαία εντός της περιοχής κάλυψης του σταθμού βάσης. Οι UEs πρέπει να έχουν ποικίλες ταχύτητες και μοτίβα κίνησης (πεζοί, οχήματα, κα) για να αντικατοπτρίζουν διαφορετικά real-world σενάρια.

Υλοποίηση Αλγορίθμων Χρονοπρογραμματισμού:

- **Proportional Fair (PF):** Ισορροπεί την απόδοση με τη δικαιοσύνη, προσπαθώντας να διασφαλίσει ότι όλοι οι χρήστες παίρνουν δίκαιη πρόσβαση στο κανάλι, ενώ επιτρέπει σε χρήστες με καλύτερη ποιότητα καναλιού να παίρνουν περισσότερους πόρους.

4G-LTE Projects

- **Max Throughput (MT):** Κατανέμει δυναμικά πόρους με βάση τις τρέχουσες συνθήκες του καναλιού. Ο αλγόριθμος δίνει προτεραιότητα στους χρήστες με την καλύτερη ποιότητα καναλιού. Καθώς οι UEs μετακινούνται και η ποιότητα του σήματος αλλάζει, ο αλγόριθμος χρονοπρογραμματισμού θα προσαρμόσει την κατανομή πόρων για να μεγιστοποιήσει τη συνολική απόδοση του δικτύου.

Δημιουργία Μοντέλων Παραγωγής Κινητικότητας: Προσομοιώστε διάφορα μοντέλα κινητικότητας για τα UEs:

- **Constant Bit Rate (CBR):** Σταθερός ρυθμός ροής δεδομένων, όπως voice calls.
- **Variable Bit Rate (VBR):** Μεταβλητός ρυθμός ροής δεδομένων, όπως video streaming ή multimedia.
- **HTTP-Like traffic:** Προσομοίωση της συμπεριφοράς περιήγησης στον ιστό: τα δεδομένα ζητούνται σε ριπές, όπως λήψη ιστοσελίδων.

Μοντελοποίηση Συνθηκών Καναλιού: Η ποιότητα καναλιού του εκάστοτε UE θα επηρεάζεται από την απόσταση του από τον σταθμό βάσης. Τα UEs που βρίσκονται κοντύτερα στο σταθμό βάσης σε κάθε χρονική στιγμή θα έχουν και την καλύτερη ποιότητα καναλιού.

Μετρικές Αξιολόγησης και Ανάλυση Απόδοσης: Κατά τη διάρκεια εκτέλεσης της προσομοίωσης, θα μετρήσετε βασικές μετρικές αξιολόγησης του συστήματος, όπως είναι το συνολικό throughput του συστήματος (system throughput), ο δείκτης δικαιοσύνης (user fairness index), η καθυστέρηση πακέτων (packet delay), η απώλεια πακέτων, καθώς και όποιες άλλες θεωρείτε εσείς απαραίτητες. Αξιολογήστε την απόδοση των αλγορίθμων χρονοπρογραμματισμού βάσει των παραπάνω μετρικών αξιολόγησης.

Επιβλέπουσα: Πουλχερία Ζερβού (pzervou@isi.gr).

16 Προσομοίωση δικτύου 4G κι εφαρμογή scheduling (RR, WFQ)

Η παρούσα εργασία έχει ως στόχο την ανάπτυξη μιας προσομοίωσης ενός δικτύου 4G με Python. Θα χρειαστεί να προσομοιώσετε την συμπεριφορά του δικτύου υπό διάφορες συνθήκες φορτίου και να αξιολογήσετε την απόδοση του με βάση διάφορες μετρικές απόδοσης.

Προσομοίωση Δικτύου 4G: Αναπτύξτε μια προσομοίωση ενός δικτύου 4G χρησιμοποιώντας Python, το οποίο περιλαμβάνει ένα σταθμό βάσης (BS) και χρήστες (UE). Κάθε UE έχει μοναδικά χαρακτηριστικά, όπως αναγνωριστικό (ID), διαφορετικές απαιτήσεις κίνησης, τυχαίους δείκτες ποιότητας καναλιού (CQI), μα και διαφορετικά επίπεδα ποιότητας της υπηρεσίας (QoS). Απαιτείται η υλοποίηση μοντέλου κινητικότητας, όπου τα UEs μετακινούνται τυχαία εντός της περιοχής κάλυψης του σταθμού βάσης. Οι UEs θα έχουν ποικίλες ταχύτητες και μοτίβα κίνησης (όπως πεζοί, οχήματα, κα) για να αντικατοπτρίζουν διαφορετικά real-world σενάρια.

Υλοποίηση Αλγορίθμων Χρονοπρογραμματισμού:

- **Round Robin (RR):** Υλοποιήστε τον αλγόριθμο Round Robin, ο οποίος κατανέμει τους πόρους του δικτύου εξίσου σε όλους τους χρήστες με κυκλική σειρά, ανεξάρτητα από τις συνθήκες καναλιού ή τις απαιτήσεις κίνησης.
- **Weighted Fair Queuing (WFQ):** Υλοποιήστε τον αλγόριθμο Weighted Fair Queuing, που κατανέμει τους πόρους του δικτύου ανάλογα με βάση τα βάρη που έχουν ανατεθεί στους χρήστες. Αναθέστε βάρη στους χρήστες με βάση την προτεραιότητα ή τις απαιτήσεις Quality of Service (QoS).

Δημιουργία Μοντέλων Παραγωγής Κινητικότητας: Προσομοιώστε διάφορα μοντέλα κινητικότητας για τα UEs:

- **Constant Bit Rate (CBR):** Σταθερός ρυθμός ροής δεδομένων, όπως voice calls.
- **Variable Bit Rate (VBR):** Μεταβλητός ρυθμός ροής δεδομένων, όπως video streaming ή multimedia.

4G-LTE Projects

- **HTTP-Like traffic:** Προσομοίωση της συμπεριφοράς περιήγησης στον ιστό: τα δεδομένα ζητούνται σε ριπές, όπως λήψη ιστοσελίδων.

Μοντελοποίηση Συνθηκών Καναλιού: Η ποιότητα καναλιού του εκάστοτε UE θα επηρεάζεται από την απόσταση του από τον σταθμό βάσης. Τα UEs που βρίσκονται κοντύτερα στο σταθμό βάσης σε κάθε χρονική στιγμή θα έχουν και την καλύτερη ποιότητα καναλιού.

Μετρικές Αξιολόγησης και Ανάλυση Απόδοσης: Κατά τη διάρκεια εκτέλεσης της προσομοίωσης, θα μετρήσετε βασικές μετρικές αξιολόγησης του συστήματος, όπως είναι το συνολικό throughput του συστήματος (system throughput), ο δείκτης δικαιοσύνης (user fairness index), η καθυστέρηση πακέτων (packet delay), η απώλεια πακέτων, καθώς και όποιες άλλες θεωρείτε εσείς απαραίτητες. Αξιολογήστε την απόδοση των αλγορίθμων χρονοπρογραμματισμού βάσει των παραπάνω μετρικών αξιολόγησης.

Επιβλέπουσα: Πουλχερία Ζερβού (pzervou@isi.gr).

17 Αλγόριθμοι scheduling για downlink σε LTE-4G δίκτυα σε MATLAB

Γενική περιγραφή. Στα LTE-4G δίκτυα τηλεπικοινωνιών, ένα κυρίαρχο ζήτημα στο δικτυακό επίπεδο του MAC είναι αυτό του scheduling των downlink και uplink requests. Χωριστά στις δύο αυτές περιπτώσεις έχουν αναπτυχθεί οι αντίστοιχες scheduling policies (χρονοδρομολογητές). Η παρούσα εργασία απαιτεί την υλοποίηση schedulers για **downlink** σε MATLAB και μια συγκριτική αποτίμησή τους βάσει μετρικών. Το σενάριο που έχουμε είναι ποικίλα user equipments (UEs) να κάνουν **downlink από** base station (BS).

Στόχοι του project. Ο στόχος του εν λόγω project είναι να αποτιμηθεί η λειτουργία πέντε (5) schedulers που επιλύουν το παραπάνω ζήτημα. Για το **downlink** ειδικά, προτείνονται οι ακόλουθοι schedulers: δύο (2) πιο απλοί, εκ των round robin, proportional fair, και maximum throughput, καθώς και τρεις (3) κάπως πιο πολύπλοκοι, εκ των maximum carrier-to-interference, quality guaranteed, packet loss ratio, earliest deadline first, modified earliest deadline first – proportional fair, modified largest weighted delay first, opportunistic packet loss fair, και exponential proportional fair. Η αποτίμηση θα γίνει βασικά με χρήση ως ανεξάρτητης μεταβλητής (input) το πλήθος των χρηστών (βλέπε ενδεικτικά n=50,100,...1000), και με χρήση εξαρτημένων μεταβλητών (outputs) τις μετρικές throughput, delay, fairness της λειτουργίας του δικτύου ως προς το scheduling. Πρέπει στο τέλος να παραχθούν όμορφα γραφήματα (scatter plots, bar plots, κλπ.), που να αποτυπώνουν το αποτέλεσμα της συγκριτικής αποτίμησης. Δίνεται η ελευθερία να επιλεγούν και άλλες μεταβλητές input, που να συνδέονται με mobility patterns (αν και ένα στατικό δίκτυο αρκεί για την εργασία) ή παραμέτρους που ορίζονται από τον εκάστοτε scheduling αλγόριθμο. Το «Survey Scheduling Paper» στο οποίο περιγράφεται μια μεγάλη γκάμα αλγόριθμων έχει αναρτηθεί στο e-class (<https://arxiv.org/pdf/2204.11369.pdf>), ενώ οι μετρικές αποτίμησης είναι οι κοινές και συνήθεις για τα δίκτυα τηλεπικοινωνιών, με αποτέλεσμα οι ορισμοί τους να μπορούν εύκολα να αναζητηθούν.

Αρχές υλοποίησης. Η υλοποίηση οφείλει να γίνει σε MATLAB, προτείνεται from scratch, και εναλλακτικά είτε με το toolbox LTE είτε με το toolbox 5G (είναι πιθανό αυτά τα toolboxes να περιπλέκουν, όμως, την υλοποίηση, παρά να τη διευκολύνουν). Ο κάθε πιο σύνθετος scheduler από τους τρεις (3) που είναι να επιλεγούν, ενδέχεται βάσει των δικών του χαρακτηριστικών να περιέχει και επιπλέον παραμέτρους. Για παράδειγμα, ο earliest deadline first προαπαιτεί γνώση των deadlines. Οι παράμετροι αυτοί κάνουν την αρχικοποίηση του δικτύου πιο σύνθετη, και δημιουργούν και νέες ανεξάρτητες μεταβλητές (inputs). Ο φοιτητής ή η φοιτήτρια έχει το ελεύθερο να κάνει παραδοχές που να στέκουν, με τις παραδοχές αυτές να φαίνονται διακριτά στην υλοποίηση και στα αποτελέσματα (όλα τα outputs πρέπει για όλους τους αλγόριθμους να αποτιμηθούν ως προς όλα τα inputs). Τα demands/requests των UEs από τον BS πρέπει να αρχικοποιηθούν ως τυχαία διανύσματα, που όμως να αποτυπώνουν ρεαλιστικά την κατάσταση στο δίκτυο (για παράδειγμα, πόσο πιθανό είναι όλοι οι UEs ταυτόχρονα να έχουν demand/request;). Το ίδιο ισχύει και για τα επιπρόσθετα πιθανά διανύσματα- παραμέτρους, βλέπε π.χ. διάνυσμα των deadlines.

4G-LTE Projects

Σημαντικό: Η τυχαιότητα στην όλη μεθοδολογία προϋποθέτει α) πολλαπλές επαναλήψεις από runs των πειραμάτων για την εξαγωγή μέσων τιμών και τυπικών αποκλίσεων των output μετρικών, καθώς και β) κατανομές πιθανοτήτων. Ποια διακριτή κατανομή πιθανότητας (π.χ. uniform, binomial, Poisson, κλπ.) είναι η πιο κατάλληλη για το κάθε τυχαίο αρχικό διάνυσμα και γιατί; Πώς οι μετρικές αλλάζουν με βάση τις στατιστικές παραμέτρους αυτών των κατανομών (βλέπε π.χ. το λ για την Poisson, που ίσως να είναι επιπρόσθετο input στο πείραμα). Οποιαδήποτε παραδοχή γίνεται δεκτή, αρκεί να τεκμηριωθεί σωστά. Τέλος, θα πρέπει οπωσδήποτε να γνωρίζετε τη θεωρία και τα μαθηματικά των αλγόριθμων scheduling.

Επιβλέπων: Γιάννης Λακουμέντας (lakoumen@upatras.gr).

18 Αλγόριθμοι scheduling για uplink σε LTE-4G δίκτυα σε MATLAB

Γενική περιγραφή. Στα LTE-4G δίκτυα τηλεπικοινωνιών, ένα κυρίαρχο ζήτημα στο δικτυακό επίπεδο του MAC είναι αυτό του scheduling των downlink και uplink requests. Χωριστά στις δύο αυτές περιπτώσεις έχουν αναπτυχθεί οι αντίστοιχες scheduling policies (χρονοδρομολογητές). Η παρούσα εργασία απαιτεί την υλοποίηση schedulers για uplink στη MATLAB, και μια συγκριτική αποτίμησή τους βάσει μετρικών. Το σενάριο που έχουμε είναι πολλαπλά user equipments (UEs) να κάνουν uplink προς base station (BS).

Στόχοι του project. Ο στόχος του εν λόγω project είναι να αποτιμηθεί η λειτουργία πέντε (5) schedulers που επιλύουν το παραπάνω ζήτημα. Για το uplink πιο ειδικά προτείνονται οι ακόλουθοι schedulers: δύο (2) πιο απλοί, εκ των round robin, proportional fair, και maximum throughput, καθώς και τρεις (3) κάπως λιγότερο απλούκοι, εκ των low complexity, low complexity delay, proportional fair delay, first maximum expansion, adaptive resource allocation based packet scheduling. Η αποτίμηση θα γίνει πρώτα με χρήση ως ανεξάρτητης μεταβλητής (input) το πλήθος των χρηστών (βλέπε ενδεικτικά $n=50, 100, \dots, 1000$), και με χρήση εξαρτημένων μεταβλητών (outputs) τις μετρικές throughput, delay, fairness της λειτουργίας του δικτύου ως προς το scheduling. Πρέπει στο τέλος να παραχθούν όμορφα γραφήματα (scatter plots, bar plots, κλπ.), που να αποτυπώνουν το αποτέλεσμα της συγκριτικής αποτίμησης. Δίνεται η ελευθερία να επιλεγούν και άλλες μεταβλητές input, που να συνδέονται με mobility patterns (αν και ένα στατικό δίκτυο αρκεί για την εργασία) ή παραμέτρους που ορίζονται από τον εκάστοτε scheduling αλγόριθμο. Το «Survey Scheduling Paper» στο οποίο περιγράφεται μια μεγάλη γκάμα αλγόριθμων έχει αναρτηθεί στο e-class (<https://arxiv.org/pdf/2204.11369.pdf>), ενώ οι μετρικές αποτίμησης είναι οι κοινές και συνήθεις για τα δίκτυα τηλεπικοινωνιών, με αποτέλεσμα οι ορισμοί τους να μπορούν εύκολα να αναζητηθούν.

Αρχές υλοποίησης. Η υλοποίηση οφείλει να γίνει σε MATLAB, προτείνεται from scratch, και εναλλακτικά είτε με το toolbox LTE είτε με το toolbox 5G (είναι πιθανό αυτά τα toolboxes να περιπλέκουν, όμως, την υλοποίηση, παρά να τη διευκολύνουν). Ο κάθε πιο σύνθετος scheduler από τους τρεις (3) που είναι να επιλεγούν, ενδέχεται βάσει των δικών του χαρακτηριστικών να περιέχει και επιπλέον παραμέτρους. Για παράδειγμα, ο first maximum expansion προαπαιτεί γνώση από channel conditions (που μπορούν με απλό τρόπο να μοντελοποιηθούν βάσει σχετικού distance των UEs από τον BS). Οι παράμετροι αυτοί κάνουν την αρχικοποίηση του δικτύου πιο σύνθετη, και δημιουργούν και νέες ανεξάρτητες μεταβλητές (inputs). Ο φοιτητής ή η φοιτήτρια έχει το ελεύθερο να κάνει παραδοχές που να στέκουν: οι παραδοχές αυτές πρέπει να φαίνονται διακριτά στην υλοποίηση και στα αποτελέσματα (όλα τα outputs πρέπει για όλους τους αλγόριθμους να αποτιμηθούν ως προς όλα τα inputs). Τα demands/requests των UEs προς τον BS θα πρέπει να αρχικοποιηθούν ως τυχαία διανύσματα, που όμως να αποτυπώνουν ρεαλιστικά την κατάσταση στο δίκτυο (π.χ., πόσο πιθανό είναι όλοι οι UEs ταυτόχρονα να έχουν demand/request;). Το ίδιο ισχύει και για τα επιπρόσθετα πιθανά διανύσματα - παραμέτρους, βλέπε π.χ. relative distances.

Σημαντικό: Η τυχαιότητα στην όλη μεθοδολογία προϋποθέτει α) πολλαπλές επαναλήψεις από runs των πειραμάτων για την εξαγωγή μέσων τιμών και τυπικών αποκλίσεων των output μετρικών, καθώς και β) κατανομές πιθανοτήτων. Ποια διακριτή κατανομή πιθανότητας (π.χ. uniform, binomial, Poisson, κλπ.) είναι η πιο κατάλληλη για το κάθε τυχαίο αρχικό διάνυσμα και γιατί; Πώς οι μετρικές αλλάζουν με βάση

τις στατιστικές παραμέτρους αυτών των κατανομών (βλέπε π.χ. το λ για την Poisson, που ίσως να είναι επιπρόσθετο input στο πείραμα). Οποιαδήποτε παραδοχή γίνεται δεκτή, αρκεί να τεκμηριωθεί σωστά. Τέλος, θα πρέπει οπωσδήποτε να γνωρίζετε τη θεωρία και τα μαθηματικά των αλγόριθμων scheduling.

Επιβλέπων: Γιάννης Λακουμέντας (lakoumen@upatras.gr).

19 Εφαρμογή clustering για τοποθέτηση σταθμών βάσης σε MATLAB

Γενική περιγραφή. Στη μη επιβλεπόμενη μηχανική μάθηση (unsupervised machine learning), μάλλον το πιο κοινό πρόβλημα που αντιμετωπίζεται αλγορίθμικά είναι η συσταδοποίηση (clustering). Ένα σύνολο οντοτήτων σε πολυδιάστατο χώρο πρέπει να διαμεριστεί σε συστάδες με τρόπο που οντότητες σε κοινή συστάδα να είναι πιο κοντά μεταξύ τους από τις οντότητες άλλων συστάδων. Μια εφαρμογή της ίσως να είναι και η βέλτιστη τοποθέτηση σταθμών βάσης (BSs) σε ένα χώρο που καλύπτεται από δίκτυο, και για το οποίο υπάρχουν ιστορικά δεδομένα εκτίμησης της κατανομής των χρηστών (UEs) όπως πιο κάτω.

Σχεδιασμός χώρου και αρχικό πείραμα. Δίνεται ένας κυβικός (3D) χώρος, με διαστάσεις, για απλότητα, στο [0, 1]. Μια συμμετρική κατανομή σημείων που αντιστοιχούν σε UEs σε αυτόν, σχηματίζουν ίσες με 9 συστάδες, 1 στο κέντρο του κύβου και 8 στις γωνίες του. Είναι εύκολο να υλοποιηθεί αυτό, με τυχαίες ισάριθμες με τις συστάδες Gaussian κατανομές, προκαθορισμένες μέσες τιμές, και επιλογή μιας κοινής και όχι σχετικά μεγάλης τυπικής απόκλισης, ώστε να διασφαλίζεται ότι οι συστάδες δεν κάνουν overlap. Προφανώς, θα πρέπει να ορίζεται ένας κανόνας οι συντεταγμένες των UEs να παραμένουν στο [0, 1]. Έστω $n=100$ το πλήθος των UEs ανά συστάδα, και σημειωθείτε ότι οι συστάδες δεν είναι πλήρεις. Καλό είναι να απεικονιστεί ο σχεδιασμός χώρου με τα UEs, για επιβεβαίωση ορθότητας του αρχικού setting.

Clustering και σταθμοί βάσης. Υπάρχει ένα σύνολο αλγόριθμων, όπως ο silhouette (αλλά και άλλοι που μπορούν να επιλεγούν), που βρίσκει το βέλτιστο πλήθος από συστάδες k. Στην περίπτωσή μας, πρέπει να επιβεβαιώθει ότι $k=9$. Υστερα, πρέπει να εφαρμοστούν τρεις (3) αλγόριθμοι clustering, και οι οποίοι με είσοδο το αρχικό setting και $k=9$ θα τοποθετήσουν βέλτιστα τους BSs (ως τα κεντροειδή συστάδων), αλλά και θα αναθέσουν σε αυτούς τα εκάστοτε κοντινότερα UEs. Οι 3 αλγόριθμοι είναι οι ακόλουθοι.

- K-means. Πρέπει να τρέξει πολλαπλές φορές (ως τυχαιοποιημένος), και να επιλεγεί το βέλτιστο του run. Παρέχει άμεσα τα κεντροειδή, τα μεγέθη των συστάδων, και τις εντός-συστάδων αποστάσεις.
- Ιεραρχικός αλγόριθμος clustering, agglomerative προσανατολισμού, με τη Ward-D linkage μέθοδο, και φυσικά με χρήση της Euclidean distance απόστασης. Θα ήταν και χρήσιμο να δοθεί ως απεικόνιση το δενδρόγραμμα που παράγεται, και κόβεται σε ορισμένο ύψος για την παραγωγή των k συστάδων.
- Gaussian mixture models που περιέχουν εσωτερικά τον expectation-maximization (EM) algorithm.

Περιγράψτε επιγραμματικά τους διάφορους αλγόριθμους, τα πλεονεκτήματα και μειονεκτήματά τους. Επιβεβαιώστε την ορθή αποτελεσματικότητα αλλά και την ισοδυναμία των αλγόριθμων στο αρχικό μας setting, με απεικόνιση του αποτελέσματος clustering, τις συγκριτικές συντεταγμένες των κεντροειδών, τα μεγέθη των σχηματισμένων συστάδων (ισούνται με n), και τη μέση απόσταση BS-UEs ανά συστάδα. Αν ορίσουμε ως «παρεμβολή σήματος» τη συνθήκη ότι ένα UE να έχει απόσταση από το BS της συστάδας του μεγαλύτερη από διπλάσια από την απόσταση από το δεύτερο πλησιέστερο BS, υπάρχουν 0 τέτοιες;

Εξέλιξη πειράματος και συμπεράσματα. Επαναλάβετε το αρχικό setting με τις παρακάτω τροποποιήσεις και όλους τους αλγόριθμους, και τις προαναφερθείσες μετρικές απότιμησης του clustering, για κάθε n στο [100, 200, 300, 400, 500], αλλά και για την κάθε τυπική απόκλιση στο [σ, 2σ, 3σ, 4σ, 5σ], με το k μα και τις μέσες τιμές των συστάδων να μένουν σταθερές. Πώς επηρεάζεται το silhouette; Τα μεγέθη των συστάδων; Οι συντεταγμένες των κεντροειδών; Μπορείτε να παρέχετε γραφήματα για τις μεταβολές των αναφερθεισών μετρικών (μέσων εντός-συστάδων αποστάσεων, πληθών «παρεμβολών σήματος»), για τους τρεις αλγόριθμους συγκριτικά, σε συνάρτηση με τη εκάστοτε n και την κάθε τυπική απόκλιση. Σαν τελικό πείραμα, δοκιμάστε στον κύβο μια ενιαία ομοιόμορφη (όχι Gaussian) κατανομή των αρχικών

900 σημείων που αντιστοιχούν σε UEs, και τοποθετήστε το βέλτιστο πλήθος BSs στα βέλτιστα σημεία, εκτιμώντας παράλληλα και όλες τις προαναφερθείσες μετρικές clustering, με τους τρεις αλγόριθμους.

Επιβλέπων: Γιάννης Λακουμέντας (lakoumen@upatras.gr).

20 Προσομοίωση handover με τυχαίους περίπατους σε MATLAB

Γενική περιγραφή. Σε ένα τυπικό δίκτυο 4G-LTE, handover λέγεται η διαδικασία αλλαγής εξυπηρέτησης από σταθμό βάσης (BSx) σε σταθμό βάσης (BSy) ενός κινούμενου χρήστη (UE). Η διαδικασία εναλλαγής αυτή (λέγεται και εναλλαγή κυψελίδων), διασφαλίζει πως οι τρέχουσες συνεδρίες επικοινωνίας (όπως είναι οι φωνητικές κλήσεις ή η μετάδοση δεδομένων) δε διακόπτονται καθώς ο χρήστης μετακινείται. Στο ακόλουθο project θα μελετηθούν μοτίβα κινητικότητας σε ένα πρότυπο setting δικτύου σε MATLAB.

Σχεδιασμός χώρου και πείραμα. Δίνεται ένα κυβικό (3D) mesh (grid), στο οποίο τοποθετούνται σημεία που αντιστοιχούν σε UEs, με ακέραιες συντεταγμένες από 0 ως 10, με τυχαίο τρόπο. Αρχικά, τα σημεία είναι $n=30$, και οι συντεταγμένες τους επιλέγονται τυχαία και ομοιόμορφα στο mesh. Στο mesh έχουμε και 9 σταθμούς βάσης: 8 στις γωνίες του, αλλά κι έναν στο κέντρο, προφανώς με συντεταγμένες (5, 5, 5). Στο αρχικό πείραμα οι BSs έχουν κοινή ισχύ, και άρα κάθε UE «παίρνει σήμα» από τον πλησιέστερο BS, με βάση την Ευκλείδεια απόσταση. Οι UEs κινούνται πάνω στο grid, με το κάθε UE να έχει δικαίωμα ή να μείνει στο ίδιο σημείο (1 επιλογή) είτε, όμως, και να κινηθεί μπροστά ή όπισθεν μια μονάδα προς κάποια από τις διαστάσεις του mesh (2x3=6 επιλογές). Όπως είναι προφανές δεν μπορεί να γίνει κίνηση εκτός του mesh. Η προσομοίωση θα αφορά $t=100$ χρονικά βήματα, και προτείνονται οι ακόλουθες τρεις (3) στρατηγικές κίνησης, που καλούνται παραλλαγές ενός τυχαίου περίπατου. Εδώ οι εξεταζόμενοι θα μπορούσαν- προαιρετικά- να μελετήσουν τον Metropolis-Hastings sampler (algorithm) και γιατί όχι και να υλοποιήσουν κάποια παραλλαγή του, πιο σύνθετη από τυχαίο περίπατο, ως περαιτέρω στρατηγική.

- **Στρατηγική πιθανής μη κίνησης.** Η πιθανότητα να μείνει στάσιμο το UE είναι 40%. Η πιθανότητα να κινηθεί το UE είναι 60%, και που ισοκατανέμεται προς κάθε πιθανή αναφερθείσα κατεύθυνση. Εάν η κίνηση τείνει να προκαλεί έξοδο από το mesh, πρέπει αντί αυτού το UE να μένει και πάλι στάσιμο.
- **Στρατηγική ισόποσων σεναρίων.** Η πιθανότητα να μείνει στάσιμο το UE είναι 1/7. Η πιθανότητα να κινηθεί το UE είναι 6/7, και που ισοκατανέμεται προς κάθε πιθανή αναφερθείσα κατεύθυνση. Και τώρα, αν αυτή η κίνηση τείνει να προκαλεί έξοδο από το mesh, προτείνεται το UE να μένει στάσιμο.
- **Στρατηγική κίνησης αναγκαστικά.** Η πιθανότητα να μείνει στάσιμο το UE είναι πια 0. Η πιθανότητα να κινηθεί το UE είναι 1, και η οποία ισοκατανέμεται προς κάθε πιθανή αναφερθείσα κατεύθυνση. Προσοχή, όμως, γιατί τώρα πια αν η κίνηση τείνει να προκαλεί έξοδο από το mesh δεν κρίνεται ως επιλογή μη κίνησης: η πιθανότητα κίνησης πια μπορεί να διαφέρει (default 1/6, μα ίσως 1/4 ή 1/3).

Περαιτέρω πειράματα. Δοκιμάστε σενάρια με η εντός του [30, 40, 50, 60, 70] με τυχαία και ομοιόμορφη τοποθέτηση. Οι BSs παραμένουν σε σταθερή θέση, όπως έχει περιγραφεί. Για να ολοκληρωθεί όλο το πείραμα απαιτείται και μια αποτίμηση. Πρέπει σε κάθε setting και κάθε βήμα να υπολογίζονται όλα τα handovers (ποσοστό επί του συνόλου): πόσα UEs αλλάζουν πλησιέστερο BS; Πόσες φορές έχουμε εδώ το φαινόμενο ping-pong, δηλαδή ένα UE να εναλλάσσει σερί BSs για τρία (3) time steps; Απαιτούνται απεικονίσεις των handovers και ping-pongs για κάθε setting (n), κάθε στρατηγική, και στην εξέλιξή τους με το χρόνο. Καλό θα είναι να απεικονιστούν και τα grids στην αρχή, στο τέλος, και σε τρία (3) μεσαία χρονικά σημεία της διαδικασίας ή αν γίνεται να κατασκευαστεί ένα demo που να παρουσιάζει όλη την εξέλιξη στο χρόνο. Τα UEs θα πρέπει να χρωματίζονται ανάλογα με τον BS τους, όπως σε ένα clustering scatterplot. Για $n=70$, τέλος, μπορείτε να υλοποιήσετε έναν αλγόριθμο αποφυγής του ping-pong, όπως να γίνεται ανάθεση σε έναν BS μόνο ύστερα από δύο (2) βήματα όπου το UE είναι πλησιέστερο αυτού;

Επιβλέπων: Γιάννης Λακουμέντας (lakoumen@upatras.gr).

Γενικές οδηγίες για όλα τα θέματα project.

Όλα τα θέματα project απαιτούν υλοποίηση. Η υλοποίηση περιγράφεται σε κάθε θέμα, και κατά κανόνα αφορά γλώσσα Python και βιβλιοθήκες της, που μπορούν να χρησιμοποιηθούν κατά το δοκούν. Μόνο τα θέματα 17-20 είναι στο περιβάλλον της MATLAB, και ενδεχόμενα toolboxes του. Για όλα τα θέματα ισχύει, ότι ο κώδικας υλοποίησης θα πρέπει να διθεί στον αντίστοιχο επιβλέποντα ως παραδοτέο, είτε με τη μορφή αρχείου ή αρχείων, είτε με κάποιο link σε cloud (π.χ. Google Colab, GitHub, κλπ.). Με βάση οδηγίες που θα δίνονται τακτικά στο e-class του μαθήματος, η δομή αξιολόγησης των θεμάτων project είναι ως εξής (ενδιάμεσα επικοινωνείτε ελεύθερα με τον επιβλέποντά σας για καθοδήγηση και αρωγή):

Ενδιάμεση παρουσίαση. Καλείστε σε ενδιάμεσο σημείο του εξαμήνου (θα οριστούν ημερομηνίες) να παρουσιάσετε κάποια πρόσδοτης υλοποίησης, ενδεικτικά έως και 50%. Η παρουσίαση γίνεται χρήσει λογισμικού παρουσίασης και με ενδεχόμενη επίδειξη εκτέλεσης της υλοποίησης (demo), εντός το πολύ 30 λεπτών της ώρας, ενώπιον του επιβλέποντα. Η φάση αυτή δεν αφορά επίσημη εξέταση (είναι, όμως, υποχρεωτική για τη λήψη προβιβάσιμου βαθμού στο μάθημα), μα αλληλεπίδραση με τον επιβλέποντα, παροχή βοήθειας και συμβουλών από αυτόν, και γενικότερο brainstorming για τη συνέχιση του project.

Τελική εξέταση. Καλείστε σε τελικό σημείο του εξαμήνου (θα οριστούν ημερομηνίες) να παρουσιάσετε όλη την υλοποίηση. Η παρουσίαση γίνεται χρήσει λογισμικού παρουσίασης και με ενδεχόμενη επίδειξη εκτέλεσης της υλοποίησης (demo), εντός το πολύ 30 λεπτών της ώρας, ενώπιον του επιβλέποντα. Εδώ γίνεται επίσημη εξέταση, η φάση προφανώς και είναι υποχρεωτική, όπως άλλωστε και η παρουσίαση στο ενδιάμεσο στάδιο, και πραγματοποιούνται ερωτήσεις αξιολόγησης κατανόησης του προβιβλήματος που πραγματεύεται το project, και οι οποίες αφορούν τόσο θεωρία όσο και λεπτομέρειες υλοποίησης.

Περίπου την περίοδο που αρχίζει η εξεταστική χειμερινού εξαμήνου, υπάρχει **και το deadline** μας για την παράδοση και μιας αναφοράς (θα οριστεί ημερομηνία). Η αναφορά πρέπει να περιγράφει την όλη διαδικασία αντιμετώπισης του θέματος αναλυτικά, με μεθόδους, αποτελέσματα, συμπεράσματα, και να είναι εντός των 10-20 σελίδων. Το τελικό παραδοτέο για προβιβάσιμο βαθμό αποτίθεται στο e-class, και περιέχει τελική αναφορά, τελική παρουσίαση (υποσύνολο της οποίας αποτελεί η ενδιάμεση), και αρχεία κώδικα (εναλλακτικά, link για cloud σε ευδιάκριτο σημείο της αναφοράς και της παρουσίασης).

- **Συμβουλή #1:** Επικοινωνείτε μέσω email ενδιάμεσα με τον επιβλέποντά σας για πάσα καθοδήγηση.
- **Συμβουλή #2:** Δίνετε το παρόν σε όσες δυνατόν παρουσιάσεις μπορείτε, κι όχι μόνο στις δικές σας.

Καλή επιτυχία, εκ των επιβλεπόντων - συμβούλων και του διδάσκοντα του μαθήματος!