ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Δυναμική Ρύθμιση Παραμέτρων Επιδημικών Αλγόριθμων στον Κινητό Υπολογισμό

Ειρήνη Σ. Φουντουκίδου

Επιβλέπων **Ευστάθιος Χατζηευθυμιάδης,** Αναπληρωτής Καθηγητής

ΑΘΗΝΑ

ΦΕΒΡΟΥΑΡΙΟΣ 2010

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Δυναμική Ρύθμιση Παραμέτρων Επιδημικών Αλγόριθμων στον Κινητό Υπολογισμό

Ειρήνη Σ. Φουντουκίδου
Α.Μ.: 1115200200057

ΕΠΙΒΛΕΠΩΝ: **Ευστάθιος Χατζηευθυμιάδης,** Αναπληρωτής Καθηγητής

ΠΕΡΙΛΗΨΗ

Η κινητικότητα αποτελεί το πιο διακριτικό γνώρισμα της σημερινής εποχής. Οι εφαρμογές χρειάζεται να λειτουργήσουν υπό συνθήκες απουσίας σταθερής υποδομής, ένεκα ασύμφορης ή αδύνατης δημιουργίας, ενώ η μετάδοση των δεδομένων γίνεται σε δίκτυα ευμετάβλητης σύνθεσης, πάνω από ασταθείς ζεύξεις και με τους κόμβους να εμφανίζουν συχνές μεταπομπές. Επιβάλλεται, επομένως, η υιοθέτηση ενός σχήματος που θα παρέχει ικανοποιητικές εγγυήσεις αξιόπιστης μετάδοσης της πληροφορίας αλλά ταυτόχρονα θα είναι ελαστικό σε προσωρινές αποτυχίες ή μόνιμες αποσυνδέσεις κόμβων. Αυτού του είδους τα χαρακτηριστικά εντοπίζονται συνδυαστικά στη συμπεριφορά των επιδημικών αλγόριθμων.

Πρόκειται για μοντέλα πιθανοθεωρητικής μετάδοσης της πληροφορίας, υπό την έννοια ότι τα δεδομένα δεν επιμολύνουν καθολικά το δίκτυο αλλά η επικοινωνία πραγματοποιείται επιλεκτικά, βάσει παραμέτρων διάχυσης. Αυτές οι παράμετροι μπορούν να ρυθμιστούν έτσι ώστε με αυθαίρετα υψηλή πιθανότητα να παρέχεται η εγγενής αξιοπιστία που παρέχουν και οι ντετερμινιστικοί αλγόριθμοι.

Με σκοπό τη βελτιστοποίηση της σχέσης κατανάλωσης ενέργειας – ποιότητας δεδομένων δικτύου και προκειμένου να ενισχυθεί η επιλεκτικότητα, το προτεινόμενο μοντέλο επιδημικού αλγόριθμου ανταμείβει τα σημαντικά δεδομένα μέσω ενίσχυσης της τιμής των σχετικών μεταβλητών διάχυσης ενώ αντίστοιχα η επικοινωνία λιγότερο σημαντικών δεδομένων γίνεται με μικρότερη πιθανότητα προώθησης, χάριν εξοικονόμησης ενέργειας. Η σπουδαιότητα της προς μετάδοση πληροφορίας ποσοτικοποιείται συναρτήσει της απόκλισης που η τελευταία εμφανίζει από τις προγενέστερες τιμές δειγματοληψίας και ως εκ τούτου, η επίδοση της επιδημικής διάχυσης συνδέεται άμεσα με την ποιότητα των δειγμάτων και κατά συνέπεια τη φύση του αλγόριθμου δειγματοληψίας.

Η πειραματική σύγκριση του ίδιου σχήματος επιδημικής μετάδοσης πληροφορίας για δύο διαφορετικούς αλγόριθμους δειγματοληψίας επαληθεύει την ανωτέρω αντίληψη. Πιο συγκεκριμένα, η εφαρμογή σταθερής συχνότητας δειγματοληψίας σε ένα τυχαίο ρεύμα δεδομένων παρουσιάζει χειρότερη αναλογία κόστους – οφέλους σε σχέση με το προτεινόμενο σχήμα μη ομοιόμορφης δειγματοληψίας, όπου η περίοδος δειγματοληψίας προσαρμόζεται στα χαρακτηριστικά του ρεύματος δεδομένων.

Με αυτό τον τρόπο καταλήγουμε σε μια τεχνολογία όπου το λογισμικό καθορίζει τις δυνατότητες που θα έχει το υλικό, προσφέροντας ιδανική παραμετροποίηση, προσαρμοσμένη στα μέτρα έκαστης εφαρμογής.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: κινητός υπολογισμός

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: επιδημικοί αλγόριθμοι, ρύθμιση παραμέτρων, πιθανοθεωρητική διάχυση πληροφορίας, κινητά δίκτυα, δειγματοληψία

ABSTRACT

Mobility is one of the most distinguishing traits of the present age. Applications need to operate under volatile conditions, where infrastructure is absent, impractical to establish or even expensive to build, while data have to be transmitted over unstable links, where no direct path from source to destination can be secured. It is therefore necessary to adopt a scheme that provides sufficient information dissemination guarantees and at the same time is flexible enough to sustain sensor network functionalities without interruption due to sensor node failures. Such features are typically encountered in the behavior of epidemic algorithms.

Epidemic algorithms are non-deterministic information dissemination algorithms, in the sense that not all data items are transmitted over the network but communication is carried out partly, based on values that epidemic algorithm parameters acquire. These parameters can be tuned so that with arbitrarily high probability the algorithm meets the guarantees that deterministic algorithms would provide. The inherent reliability of epidemic algorithms lies in a proactive mechanism, where redundancy and randomization are deployed in the first place to circumvent potential failures.

However, in order to optimize the energy consumption – data quality relationship, enforce selectivity and discourage spams, the proposed epidemic algorithm rewards important data items by enhancing the value of the diffusion variables and purges unnecessary ones. Evaluation on the importance of the observed piece of information is performed through the rate of change of the sampled data and thus success of the suggested epidemic variation is directly depended on the adequacy of the sampling algorithm applied.

Experimental results verify the above concept. In particular, use of constant sampling frequency ends up in worse cost – benefit ratio for the same epidemic algorithm implementation as opposed to the proposed non-uniform sampling scheme, where sampling period is adjusted to the data stream characteristics.

In this way, the new technological paradigm that is put forward forms the basis of an era where software determines hardware capabilities, resulting in a tuning schema that is well-tailored for every application.

SUBJECT AREA: mobile computing

KEYWORDS: epidemic algorithms, parameters' tuning, probabilistic information dissemination, wireless networks, sampling

ΠΕΡΙΕΧΟΜΕΝΑ

1. ΕΙΣΑΓΩΓΗ 10

2. ΑΝΑΛΥΤΙΚΗ ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ 12

3. ΕΠΙΣΚΟΠΗΣΗ ΠΕΔΙΟΥ 14

3.1 Μοντέλο SIR 15

3.2 Μοντέλο SIS 16

3.3 Μοντέλο SI 17

3.4 Μοντέλο SaIS 18

3.5 Μοντέλο aSIS 20

4. ΑΛΓΟΡΙΘΜΟΣ ΜΕΤΑΒΛΗΤΗΣ ΣΥΧΝΟΤΗΤΑΣ ΔΕΙΓΜΑΤΟΛΗΨΙΑΣ 24

4.1 Περιγραφή Αλγόριθμου 24

4.2 Αξιολόγηση Αλγόριθμου 25

4.2.1 Ορισμός Μετρικών 25

4.2.2 Μοντέλο Προσομοίωσης 26

4.2.3 Αποτελέσματα Προσομοίωσης 29

5. ΣΥΜΠΕΡΑΣΜΑΤΑ 41

ΠΙΝΑΚΑΣ ΟΡΟΛΟΓΙΑΣ 43

ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ – ΑΚΡΩΝΥΜΙΑ 44

ΠΑΡΑΡΤΗΜΑ Ι 45

ΠΑΡΑΡΤΗΜΑ ΙΙ 66

ΑΝΑΦΟΡΕΣ 68

ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

Σχήμα 1: Η (πιθανοθεωρητική) μετάδοση της πληροφορίας σε ένα ασύρματο δίκτυο αισθητήρων με έναν μοναδικό κόμβο - πηγή και κόμβους - αναμεταδότες. 15

Σχήμα 2: Η εναλλαγή των καταστάσεων ενός κόμβου (εξέλιξη στο χρόνο) 19

Σχήμα 3: Μεταβολή της τιμής χρονικής εγκυρότητας σε πηγή και αναμεταδότη, συναρτήσει του χρόνου 22

Σχήμα 4: Αρχικό, αναλογικό σήμα εισόδου στον κόμβο - πηγή 28

Σχήμα 5: Αποτέλεσμα εφαρμογής ομοιόμορφης και μη ομοιόμορφης δειγματοληψίας στο αρχικό σήμα από τον κόμβο - πηγή 30

Σχήμα 6: Μικροσκοπική ανάλυση του σήματος που αντιλαμβάνεται ο κόμβος – πηγή, ως αποτέλεσμα εφαρμογής ομοιόμορφης και μη ομοιόμορφης δειγματοληψίας 31

Σχήμα 7: Χρονική διακύμανση της περιόδου δειγματοληψίας r_s, με $r_s(0) = 5$ 31

Σχήμα 8: Αθροιστικός δείκτης φορών δειγματοληψίας συναρτήσει του χρόνου 32

Σχήμα 9: Απόκλιση και σχετική απόκλιση των δεδομένων του κόμβου – πηγή από την πραγματική τιμή, μετά την εφαρμογή δειγματοληψίας 33

Σχήμα 10: Χρονική διακύμανση της πιθανότητας προώθησης β_s του κόμβου - πηγή ... 35

Σχήμα 11: Σήμα που αντιλαμβάνεται ο τυχαίος κόμβος – αναμεταδότης ως αποτέλεσμα εφαρμογής του ίδιου επιδημικού αλγόριθμου για πηγή ομοιόμορφης και μη ομοιόμορφης δειγματοληψίας 36

Σχήμα 12: Σχετική απόκλιση των δεδομένων του τυχαίου κόμβου – αναμεταδότη από την πραγματική τιμή, μετά την εφαρμογή του ίδιου επιδημικού αλγόριθμου 37

Σχήμα 13: Μέση τιμή σφάλματος – σύγκριση με το πρωτογενές σήμα 37

Σχήμα 14: Μέση τιμή σφάλματος – σύγκριση με το σήμα που αντιλαμβάνεται ο κόμβος – πηγή 38

Σχήμα 15: Αποτελεσματικότητα των σχημάτων ομοιόμορφης και μη ομοιόμορφης δειγματοληψίας συναρτήσει του χρόνου 39

Σχήμα 16: Ακρίβεια των σχημάτων ομοιόμορφης και μη ομοιόμορφης δειγματοληψίας – σύγκριση με το πρωτογενές σήμα και με το σήμα που αντιλαμβάνεται ο κόμβος – πηγή 39

Σχήμα 17: Σωρευτικό πλήθος διακινηθέντων μηνυμάτων συναρτήσει του χρόνου 40

Σχήμα 18: Αποδοτικότητα των σχημάτων ομοιόμορφης και μη ομοιόμορφης δειγματοληψίας συναρτήσει του χρόνου 40

Σχήμα 19: Απλοποιημένο διάγραμμα κλάσεων του προγράμματος προσομοίωσης 45

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

Εικόνα 1: Στιγμιότυπο της θέσης των κόμβων του δικτύου προσομοίωσης (αναπαράσταση μέσω Mobisim).....27

Εικόνα 2: Μέρος της διαδρομής του κόμβου – πηγή και ενός τυχαίου κόμβου - αναμεταδότη (αναπαράσταση μέσω Mobisim) 67

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

Πίνακας 1: Πίνακας Παραμέτρων Προσομοίωσης 29

Πίνακας 2: Χαρακτηριστικά Μοντέλου Κινητικότητας 66

1. ΕΙΣΑΓΩΓΗ

Τα τελευταία χρόνια υπήρξε παγκόσμιο ενδιαφέρον για την ανάπτυξη τεχνολογιών που επιτρέπουν την αποτελεσματική αλλά και αποδοτική διάχυση περιεχομένου σε *αυτοοργανούμενα* (ad-hoc) δίκτυα κινητών κόμβων. Η έννοια της "αποτελεσματικής διάχυσης" αναφέρεται στην ποιότητα της διακινούμενης πληροφορίας (υψηλή διαθεσιμότητα, χρονική εγκυρότητα, μικρά σφάλματα) ενώ η "αποδοτικότητα" ανάγεται σε όρους εξοικονόμησης ενέργειας και εύρους ζώνης.

Το παραδοσιακό υπολογιστικό μοντέλο *πελάτη – εξυπηρέτη* (client – server) αποδεικνύεται ακατάλληλο για εφαρμογή σε αυτού του είδους τα περιβάλλοντα, εξαιτίας των περιορισμών που υφίστανται στη διαθεσιμότητα δικτυακής υποδομής και στη γενικότερη αξιοπιστία του δικτύου.

Από την άλλη, οι αλγόριθμοι *πλημμύρας* (flooding), που χρησιμοποιούνται ευρέως στα *ομότιμα δίκτυα* (peer-to-peer), επιτρέπουν την αξιόπιστη αναπαραγωγή δεδομένων μεταξύ κόμβων, ωστόσο, η υιοθέτηση τους είναι ασύμφορη στα ασύρματα κινητά δίκτυα από τη σκοπιά της οικονομίας.

Είναι προφανές, επομένως, πως το σύστημα πρέπει να έχει την ικανότητα "*να ανακαλύπτει, να ερμηνεύει, να συμπεραίνει, να αξιοποιεί και να συλλογίζεται βάσει της περιρρέουσας πληροφορίας, ώστε να λαμβάνει αποφάσεις και να προσαρμόζεται σε διάφορες καταστάσεις*" [2]. Με αυτόν τον τρόπο θα μπορέσει να υλοποιηθεί επικοινωνία κατά την οποία θα γίνεται επιλεκτική μόνο αναμετάδοση του μηνύματος από τους νομαδικούς κόμβους, προκειμένου αυτοί να πετύχουν τον κοινό τους σκοπό: καθολική και ταυτόχρονα ενεργειακά αποδοτική διάχυση πληροφορίας ή γνώσης.

Μια τέτοια επικοινωνία θα βασίζεται σε τεχνικές *επιδημίας* (epidemical spreading). Η αρχή που κρύβεται πίσω από αυτού του τύπου τη διάδοση πληροφοριών μιμείται την τεχνική διάδοσης μιας επιδημίας. Οι γνώσεις μας από το πεδίο της βιολογίας για τη μετάδοση μολυσματικών ασθενειών σε τμήματα πληθυσμού προσφέρουν την πρώτη ύλη για τη διαμόρφωση αλγόριθμων που θα είναι ικανοί να διαχέουν πληροφορίες σε μεγάλα σύνολα κόμβων με αξιόπιστο και αποδοτικό τρόπο. Η ρύθμιση των παραμέτρων που υπεισέρχονται κατά τη μαθηματική μοντελοποίηση των επιδημικών αλγόριθμων επιτρέπει, με αυθαίρετα υψηλή πιθανότητα, την ικανοποίηση των εγγυήσεων αξιοπιστίας που παρέχουν οι αντίστοιχοι ντετερμινιστικοί αλγόριθμοι διάχυσης δεδομένων, ενώ η σύγκριση των κλασικών σχημάτων επιδημίας με άλλες τεχνικές διάδοσης πληροφορίας (π.χ. πλημμύρα) υπογραμμίζει την οικονομία που επιτυγχάνεται από ενεργειακής απόψεως.

Η στάθμιση της επιρροής των ως άνω παραμέτρων στην επίδοση και απόδοση των επιδημικών αλγόριθμων αποτελεί το επίκεντρο της παρούσας μελέτης. Πρόκειται για μια προσπάθεια αξιοποίησης της μεταβλητότητας του περιβάλλοντος και των δεδομένων διάχυσης με γνώμονα τη βέλτιστη δυνατή σχέση μεταξύ αξιοπιστίας και κατανάλωσης.

Η δομή της εργασίας έχει ως εξής:

Αρχικά εισάγεται η έννοια του επιδημικού αλγόριθμου. Δίδονται οι σχετικοί ορισμοί, αναλύονται οι υφιστάμενες παραλλαγές και παρουσιάζονται οι παράμετροι που υπεισέρχονται στη λειτουργία των διαφόρων μοντέλων επιδημίας.

Ακολουθεί το κεντρικό κεφάλαιο της εργασίας. Σε αυτό, περιγράφεται το προτεινόμενο σχήμα και η συλλογιστική που το συνοδεύει, προτείνονται μετρικές αξιολόγησής του και παρουσιάζονται τα αποτελέσματα της σχετικής μελέτης.

Στο επόμενο κεφάλαιο γίνεται αναφορά σε προβληματισμούς που αφορούν στα παραγόμενα συμπεράσματα ενώ προτείνονται περιοχές βελτίωσης και περαιτέρω έρευνας.

Τέλος, ακολουθούν τα Παραρτήματα, στα οποία παρατίθεται η υποδομή (κώδικας και εργαλεία) που αποτέλεσε τον πυρήνα σύνθεσης του παρόντος δοκιμίου.

2. ΑΝΑΛΥΤΙΚΗ ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ

Είναι πλέον κοινός τόπος πως η ανάπτυξη τόσο των ασύρματων επικοινωνιών όσο και των φορητών υπολογιστικών συσκευών δημιούργησε ένα νέο υπολογιστικό πρότυπο, τον *κινητό υπολογισμό* (mobile computing), σύμφωνα με το οποίο οι χρήστες φέρουν συσκευές που παρέχουν πρόσβαση σε υπηρεσίες πληροφόρησης, ανεξάρτητα από τη φυσική τους θέση ή τα χαρακτηριστικά της κίνησής τους.

Ένα τέτοιο νέο περιβάλλον εισάγει νέες προκλήσεις στον τομέα του διαμοιρασμού πληροφοριών. Οι παραδοσιακές τεχνικές πρόσβασης θεωρούν δεδομένη την ύπαρξη μόνιμης και σταθερής σύνδεσης, γεγονός εν γένει παραπλανητικό για την περίπτωση του κινητού υπολογισμού. Στον κινητό υπολογισμό, οι κόμβοι δημιουργούν συστάδες και ανταλλάσσουν πληροφορίες μέσω ασταθών ζεύξεων. Οι πόροι που διαθέτουν είναι περιορισμένοι (εύρος ζώνης, υπολογιστική δύναμη, μνήμη) και η επικοινωνία πραγματοποιείται στο πλαίσιο συχνών αποσυνδέσεων (π.χ. έξοδος από την εμβέλεια επικοινωνίας).

Αναλυτικότερα, το κλασικό πρότυπο πελάτη – εξυπηρέτη προϋποθέτει την ύπαρξη ισχυρών υπολογιστικών κόμβων σε προκαθορισμένη τοποθεσία, όπου συνδέονται με - κατά κύριο λόγο- *ενακόλουθες συνόδους* (stateful sessions) συσκευές με θεωρητικώς απεριόριστους πόρους. Ωστόσο, στην περίπτωση του κινητού υπολογισμού, δε θα πρέπει να θεωρείται δεδομένη η ύπαρξη σταθερής υποδομής, καθώς είναι δυνατόν να δημιουργηθούν *αυτόνομα δίκτυα ειδικού σκοπού* (Mobile Ad-hoc NETworks - MANET) σε απροσδόκητα μέρη, χωρίς την παρουσία κεντρικής διαχείρισης. Επιπρόσθετα και δεδομένου ότι οι κόμβοι των εξυπηρετών αποτελούν *κρίσιμο σημείο αστοχίας* (single point of failure), η εισαγωγή πλεονασμού και του απαραίτητου συγχρονισμού που απαιτείται για τη διατήρηση της συνοχής των δεδομένων αποτελούν σημαντική πηγή *πλεονάζοντος κόστους* (overhead) σε ένα σκηνικό ούτως ή άλλως σφιχτού ενεργειακού προϋπολογισμού.

Από την άλλη, στο υπόδειγμα των ομότιμων δικτύων, τα κεντρικά σημεία αστοχίας εξαφανίζονται, όπως επίσης και τα σχετικά προβλήματα συμφόρησης και αξιοπιστίας. Όμως, οι πόροι των υπολογιστών επιβαρύνονται περαιτέρω, καθώς πρέπει να υποστηρίξουν όχι μόνο το χρήστη του σταθμού εργασίας, αλλά και τα αιτήματα των χρηστών του δικτύου.

Παρόλο που ολοένα και περισσότερες έρευνες προσανατολίζονται σε αρχιτεκτονικές εξοικονόμησης ενέργειας, οι προτεινόμενες λύσεις παρουσιάζουν περιθώρια βελτίωσης για αυτά τα συστήματα περιορισμένων πόρων. Η έννοια της βελτιστοποίησης της ενέργειας, στην περίπτωση των δικτύων κινητού υπολογισμού, είναι πολύ πιο πολύπλοκη, καθώς δεν αφορά μόνον τη μείωση της κατανάλωσης ενός και μόνον κόμβου, αλλά πρωτίστως αφορά τη διατήρηση της συνοχής του δικτύου και συνεπώς τη μεγιστοποίηση της ζωής του.

Επί παραδείγματι, τα δίκτυα αισθητήρων, που αποτελούν τον κυριότερο αντιπρόσωπο δικτύων κινητού υπολογισμού, απαιτούν διάρκεια ζωής μηνών έως και ετών, καθώς η αντικατάσταση των ενεργειακών στοιχείων εκατοντάδων διάσπαρτων κόμβων είναι ασύμφορη, αν όχι αδύνατη. Στην περίπτωση πρόωρης αστοχίας μιας μερίδας κόμβων (ενδεχομένως αυτών που εξυπηρετούν το μεγαλύτερο ποσοστό κίνησης του δικτύου), το εναπομείναν δίκτυο είναι πιθανό να διαχωριστεί σε μικρότερες, απομονωμένες μεταξύ τους, ομάδες κόμβων. Ακόμα και αν αυτές οι ομάδες αποτελούνται από κόμβους που λειτουργούν με εναλλακτικές πηγές ενέργειας (ηλιακή, γεωθερμική κ.ά.), είναι προφανές πως το αρχικό δίκτυο έχει κατακερματιστεί προτού ενδεχομένως να εκπληρώσει το σκοπό του.

Κρίνεται, επομένως, αναγκαίο να βρεθεί ένα σχήμα που θα εγγυάται αποτελεσματική αλλά και αποδοτική διάδοση πληροφορίας. Το νέο αυτό σχήμα θα πρέπει να εξισορροπεί το φορτίο που διαμοιράζεται στις διεργασίες του συστήματος, είτε αυτό αφορά απλή προώθηση μηνυμάτων, είτε αφορά πιο σύνθετη επεξεργασία, προκειμένου να εξισορροπείται ανάλογα και η καταναλισκόμενη ενέργεια. Ένα πρώτο βήμα προς αυτή την κατεύθυνση είναι η υιοθέτηση *συνεργατικών κόμβων* (collaborative node) υπό τη μορφή *επιδημικών αλγόριθμων* (epidemic algorithm).

Η γενική ιδέα πίσω από τους επιδημικούς αλγόριθμους είναι η περιοδική επικοινωνία της (μερικής) κατάστασης που αντιλαμβάνεται κάθε διεργασία για το σύστημα στις υπόλοιπες διεργασίες, συνεισφέροντας στη δημιουργία μιας κοινής αντίληψης για το περιβάλλον. Με άλλα λόγια, οι συνεργατικοί κόμβοι συλλέγουν τη διαχεόμενη πληροφορία, όχι μόνο για λογαριασμό τους, αλλά και για να τη μοιραστούν με τους γειτονικούς κόμβους. Αυτό σημαίνει πως κόμβοι που δε διαθέτουν αισθητήρες –άρα δεν είναι ικανοί να πάρουν μόνοι τους μετρήσεις- μπορούν να αποκτήσουν τα δεδομένα αυτά μετά την εφαρμογή επιδημικής διάχυσης πληροφορίας. Έτσι, επιτυγχάνεται οικονομία κλίμακας κατά την εγκατάσταση και λειτουργία του δικτύου, καθώς περιορίζεται η χρήση των ακριβών και ενεργοβόρων κόμβων αισθητήρων.

Επιπλέον, το μοντέλο επιδημίας είναι απλό στην εφαρμογή και ευπροσάρμοστο στις τοπολογικές και πληθυσμιακές αλλαγές. Εγγενώς η αξιοπιστία των επιδημικών αλγόριθμων βασίζεται σε έναν προνοητικό μηχανισμό, κατά τον οποίο ο πλεονασμός και η τυχαιότητα λειτουργούν έτσι ώστε να παρακάμπτονται πιθανές αστοχίες διαδικασιών ή/και συνδέσεων δικτύου. Αυτό συμβαίνει επειδή, εξ ορισμού, κάθε διαδικασία αποφασίζει πιθανοθεωρητικά την επικοινωνία με το σύνολο των γειτόνων της. Αυτοί επαναλαμβάνουν την ίδια διαδικασία και ούτω καθεξής. Καμία πρόνοια και κανείς μηχανισμός δεν απαιτείται για την ανίχνευση και το χειρισμό των αποτυχιών. Συνεπώς, όσο αυξάνεται η συχνότητα ή/και η ακτίνα επικοινωνίας ενός κόμβου με τη γειτονιά του, τόσο αυξάνεται και η αξιοπιστία του δικτύου. Από την άλλη, αύξηση της τιμής των παραμέτρων επικοινωνίας συνεπάγεται αύξηση και της ενέργειας που απαιτείται για την υλοποίηση της εν λόγω επικοινωνίας.

Είναι προφανές, επομένως, πως πρέπει να αναζητηθεί η ιδανική ισορροπία μεταξύ της κάλυψης του χώρου με ενημερωμένη πληροφορία και της επιμήκυνσης της ζωής του δικτύου. Παρόλο που το κλασικό σχήμα επιδημίας αποτελεί οικονομικότερη λύση από τη σκοπιά της ενέργειας σε σύγκριση με άλλες τεχνικές (π.χ. πλημμύρα), η ρύθμιση των παραμέτρων που υπεισέρχονται κατά τη μοντελοποίηση του σχήματος επιτρέπει μεγαλύτερα περιθώρια οικονομίας εις βάρος της ποιότητας της διακινούμενης πληροφορίας. Αντίθετα, εφόσον η εκάστοτε εφαρμογή επιτρέπει μικρότερα περιθώρια σφάλματος στη πληροφορία που κυκλοφορεί στο δίκτυο, οι αντίστοιχες παράμετροι δύνανται να μεταβληθούν κατάλληλα, προκειμένου να ενισχυθεί η αποτελεσματικότητα του δικτύου.

Προτού, όμως, κανείς προσπαθήσει να ερμηνεύσει τους παράγοντες που επηρεάζουν την αποτελεσματικότητα και αποδοτικότητα του επιδημικού δικτύου κινητού υπολογισμού, είναι σημαντικό να γνωρίσει τη θεωρία των επιδημικών αλγόριθμων, καθώς επίσης και τις διάφορες παραλλαγές τους.

3. ΕΠΙΣΚΟΠΗΣΗ ΠΕΔΙΟΥ

Όπως έγινε σαφές από τα ανωτέρω, η κατανάλωση ενέργειας αποτελεί μια πολύ σημαντική πτυχή στα δίκτυα κινητού υπολογισμού, καθώς η συντριπτική πλειοψηφία των κόμβων βασίζεται σε μπαταρίες για τη λειτουργία τους. Είναι, επομένως, αναγκαία η προσαρμογή των υπαρχόντων αλγόριθμων διάδοσης δεδομένων, έχοντας ως γνώμονα την οικονομία των πόρων.

Οι γνώσεις μας από το πεδίο της βιολογίας για τη μετάδοση μολυσματικών ασθενειών προσφέρουν την πρώτη ύλη για τη διαμόρφωση αλγόριθμων που προσανατολίζονται στην αντιμετώπιση των σχετικών προκλήσεων ενέργειας. Η αρχή που κρύβεται πίσω από αυτού του τύπου τη διάδοση πληροφοριών μιμείται την τεχνική διάδοσης της επιδημίας, έτσι αναφερόμαστε σε επιδημική διάδοση πληροφορίας. Ανάλογο παράδειγμα αποτελεί και η *διάδοση φήμης* στη γειτονιά (rumor spreading). Ο όρος αυτός περιγράφει τη μετάδοση πληροφοριών στους γειτονικούς κόμβους, οι οποίοι ενδιαφέρονται για το συγκεκριμένο περιεχόμενο (δηλαδή, το κουτσομπολιό).

Στο φυσικό κόσμο, ως εξάπλωση μιας επιδημίας ορίζεται η διάδοση μολυσματικής ασθένειας στα μέλη ενός πληθυσμού. Η ευκολία με την οποία μια επιδημία μετατρέπεται σε πανδημία (καθολική μόλυνση του πληθυσμού) εξαρτάται τόσο από τα δημογραφικά χαρακτηριστικά του πληθυσμού (κινητικότητα, μέγεθος γειτονιάς) όσο και από τις ιδιότητες και την ανθεκτικότητα της παθογένειας. Ωστόσο, είτε η επιδημία εμφανίζει επιθετικά χαρακτηριστικά είτε τείνει να εξασθενεί με την πάροδο του χρόνου, η διάδοσή της προϋποθέτει κάποιου είδους επαφή μεταξύ ενός τουλάχιστον μολυσμένου ατόμου και των μελών του πληθυσμού.

Από την άλλη, στο πλαίσιο των επιδημικών αλγόριθμων, ως εξάπλωση μιας επιδημίας ορίζεται η διάδοση ενός *ρεύματος δεδομένων* (data stream) στους κόμβους ενός δικτύου κινητού υπολογισμού. Η ευκολία με την οποία η πληροφορία καλύπτει το σύνολο του δικτύου καθορίζεται τόσο από την πυκνότητα και τα χαρακτηριστικά κίνησης των κόμβων όσο και από τη διακύμανση των τιμών της πηγής δεδομένων, ενώ αντίστοιχα, η διάδοση των πληροφοριών προϋποθέτει επικοινωνία μεταξύ ενός τουλάχιστον μολυσμένου κόμβου και των λοιπών κόμβων του δικτύου.

Για το σκοπό αυτό και λόγω της ασύρματης φύσης της επικοινωνίας και των επακόλουθων περιορισμών στην ακτίνα επικοινωνίας (r), είναι απαραίτητο οι κινητοί κόμβοι να παραμένουν σε κοντινή απόσταση μεταξύ τους, προκειμένου να εξασφαλίζεται η συνοχή του δικτύου.

Το υποσύνολο των κόμβων του πληθυσμού που βρίσκεται εντός της εμβέλειας επικοινωνίας ενός κόμβου αποτελούν τη *γειτονιά* (neighborhood) του κόμβου αυτού.

Ανά πάσα στιγμή, οι κόμβοι του δικτύου επιδημίας χωρίζονται σε τρεις ομάδες: στους *υγιείς* κόμβους (susceptible), στους *μολυσμένους* (infected) και στους *άνοσους* (removed) και ισχύει ότι:

$$S_t + I_t + R_t = N_t, \forall t \; \kappa\alpha\iota \; I_t \neq \emptyset$$

όπου S_t το σύνολο των υγιών κόμβων του δικτύου, I_t το σύνολο των μολυσμένων κόμβων του δικτύου, R_t το σύνολο των ανοσοποιημένων κόμβων του δικτύου και N_t ο πληθυσμός του δικτύου κατά τη χρονική στιγμή t.

Εν γένει, στην ομάδα των υγιών ανήκουν οι κόμβοι που δεν είναι εξοπλισμένοι με αισθητήρα (συνεπώς δε διαθέτουν άμεση πρόσβαση στο ρεύμα των δεδομένων) και ταυτόχρονα η πληροφορία που φέρουν είναι *απαρχαιωμένη* (stale).

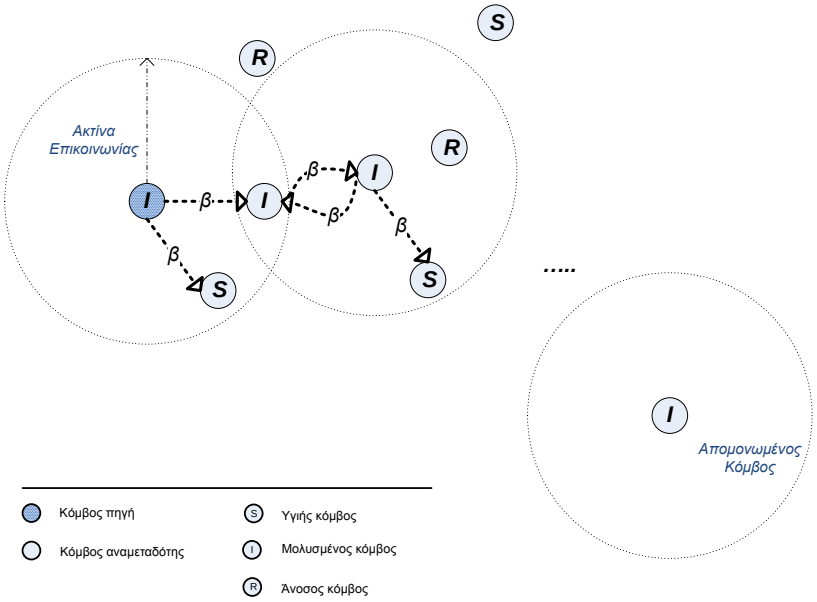
Στην κατηγορία των μολυσμένων, περιλαμβάνονται οι κόμβοι εκείνοι του πληθυσμού που φέρουν ενημερωμένη πληροφορία, είτε λόγω της ικανότητάς τους να τη συλλέξουν

οι ίδιοι (*κόμβοι πηγή* – source nodes), είτε λόγω της συναναστροφής τους με άλλους, μολυσμένους κόμβους (*κόμβοι αναμεταδότες* – relay nodes). Αυτοί μπορούν να μολυνθούν εκ νέου από μολυσμένους κόμβους που φέρουν πιο πρόσφατη πληροφορία πλαισίου και με τη σειρά τους να μολύνουν μολυσμένους κόμβους που φέρουν παλαιότερη πληροφορία ή/και υγιείς κόμβους.

Το συμπληρωματικό υποσύνολο του πληθυσμού αποτελείται από τους άνοσους κόμβους. Πρόκειται για περιπτώσεις μολυσμένων κόμβων, οι οποίοι δεν επέστρεψαν ποτέ σε υγιή κατάσταση, αλλά ούτε παρέμειναν στη μολυσμένη κατάσταση επ' άπειρον. Αντίθετα, οι κόμβοι αυτοί ανοσοποιούνται, δεν μπορούν να μολυνθούν ξανά με πληροφορία πλαισίου, ούτε δύνανται να μεταδώσουν την πληροφορία τους σε άλλους.

Αυτή είναι η γενική ιδέα πίσω από τους επιδημικούς αλγόριθμους. Με λίγα λόγια, οι κόμβοι του πληθυσμού που ανήκουν στην ίδια γειτονιά ανταλλάσσουν στοχαστικά πληροφορίες, προκειμένου να ενημερωθούν και να συμπεράνουν για την κατάσταση του δικτύου με ένα συνεργατικό τρόπο. Οι πληροφορίες αυτές είναι διαθέσιμες είτε μέσω αυτόνομης δειγματοληψίας, είτε κατόπιν επικοινωνίας με άλλους κόμβους.

Η πιθανότητα με την οποία ο μολυσμένος κόμβος αποφασίζει για τη μετάδοση της επιδημίας στη γειτονιά του καλείται *πιθανότητα μετάδοσης* (forwarding probability) και συμβολίζεται με β.



Σχήμα 1: Η (πιθανοθεωρητική) μετάδοση της πληροφορίας σε ένα ασύρματο δίκτυο αισθητήρων με έναν μοναδικό κόμβο - πηγή και κόμβους - αναμεταδότες.

Αντίστοιχα, η πιθανότητα ανάρρωσης ενός μολυσμένου κόμβου (δεδομένου ότι δε φέρει πλέον πλαίσιο εν ισχύ) ονομάζεται *ρυθμός ίασης* (recovery rate) και συμβολίζεται με δ, ενώ η πιθανότητα ανοσοποίησης ενός μολυσμένου κόμβου καλείται *ρυθμός ανοσοποίησης* (removal rate) και συμβολίζεται με α.

Καθώς είναι λογικό, κάθε κόμβος χαρακτηρίζεται από μία και μόνο κατάσταση ανά πάσα στιγμή, επομένως, ένας κόμβος που τη μία χρονική στιγμή ήταν μολυσμένος, την επόμενη χρονική στιγμή είτε θα αναρρώσει (με πιθανότητα δ), είτε θα ανοσοποιηθεί (με πιθανότητα α), είτε θα παραμείνει μολυσμένος (με πιθανότητα 1-δ-α).

3.1 Μοντέλο SIR

Το σχήμα που αποτυπώνει την ανωτέρω γενική λειτουργία διακίνησης μηνυμάτων του επιδημικού πρωτοκόλλου είναι το εξής:

```
if STATE is not REMOVED
then
        if MSG Mi RECEIVED FOR THE FIRST TIME
        then
                STATE ← INFECTED
                M ← f(Mi, M)
        endif
        if STATE is INFECTED
        then
                if random1 < β
                then
                        TRANSMIT M TO ALL NEIGHBORS
                endif
                if random2 < δ
                then
                        STATE ← SUSCEPTIBLE
                elseif δ ≤ random2 < α+δ
                then
                        STATE ← REMOVED
                endif
                endif
        endif
endif
```

αναφέρεται συχνά ως μοντέλο SIR (λόγω των τριών πιθανών καταστάσεων των κόμβων – Susceptible, Infected, Removed) και αποτελεί βάση για τη διαμόρφωση μιας πληθώρας παραλλαγών επιδημικών αλγόριθμων, που προοδευτικά στοχεύουν στην εξισορρόπηση των απαιτήσεων για αξιόπιστη και αποδοτική διάδοση πληροφοριών.

3.2 Μοντέλο SIS

Στην οριακή περίπτωση όπου α=0 (μηδενική πιθανότητα ανοσοποίησης), κανείς κόμβος του δικτύου δε βρίσκεται στην κατάσταση Removed και οι μονάδες του πληθυσμού αμφιταλαντεύονται ανάμεσα στις καταστάσεις Susceptible και Infected. Ένας υγιής κόμβος έρχεται σε επαφή με έναν μολυσμένο και εντάσσεται στο σύνολο των μολυσμένων, όπου παραμένει μέχρι να θεραπευτεί, ενώ με πιθανότητα δ≠0, ο κόμβος θεραπεύεται και επανέρχεται στο σύνολο των υγιών.

Εξαιτίας της παλινδρόμησης αυτής (Susceptible → Infected → Susceptible), το σχήμα ονομάσθηκε SIS και η λειτουργικότητα του τυχαίου κόμβου του δικτύου περιγράφεται ως κάτωθι:

```
if MSG Mi RECEIVED FOR THE FIRST TIME
then
        STATE ← INFECTED
        M ← f(Mi, M)
endif
if STATE is INFECTED
then
        if random1 < β
        then
                TRANSMIT M TO ALL NEIGHBORS
        endif
        if random2 < δ
        then
                STATE ← SUSCEPTIBLE
        endif
endif
```

3.3 Μοντέλο SI

Θέτοντας δ=0 (μηδενική πιθανότητα ανάρρωσης) προκύπτει η απλούστερη και κλασικότερη κατηγορία επιδημικού αλγόριθμου. Όπως υποδεικνύει και η ονομασία SI (Susceptible, Infected), σε αυτή την κλάση, αρχικά, σχεδόν κάθε κόμβος είναι υγιής. Όταν κάποια πληροφορία καταφέρει να μολύνει έναν κόμβο, τότε αυτός παραμένει μολυσμένος για όσο διάστημα αποτελεί μέλος του πληθυσμού του δικτύου:

```
if MSG Mi RECEIVED FOR THE FIRST TIME
then
        STATE ← INFECTED
        M ← f(Mi, M)
endif
if STATE is INFECTED
then
        if random1 < β
        then
                TRANSMIT M TO ALL NEIGHBORS
        endif
endif
```

Συνεπώς, με την πάροδο του χρόνου, όλοι οι κόμβοι θα γίνουν εν τέλει αποδέκτες της πληροφορίας, δηλαδή θα μολυνθούν.

Εξειδίκευση του μοντέλου SI αποτελεί ο αλγόριθμος πλημμύρας. Σε αυτόν, κάθε κόμβος που έχει στη διάθεσή του ενημερωμένο πλαίσιο, το επικοινωνεί ντετερμινιστικά σε όλους τους γειτονικούς κόμβους, ήτοι β=1.

```
if MSG Mi RECEIVED FOR THE FIRST TIME
then
        STATE ← INFECTED
        M ← f(Mi, M)
endif
if STATE is INFECTED
then
        TRANSMIT M TO ALL NEIGHBORS
endif
```

Είναι προφανές πως ο αλγόριθμος πλημμύρας παρέχει τις εγγυήσεις ποιότητας που υπόσχονται και οι λοιποί ντετερμινιστικοί αλγόριθμοι, ωστόσο η λειτουργία του προσθέτει σημαντικό φόρτο σε ένα δίκτυο με ούτως ή άλλως περιορισμένους πόρους.

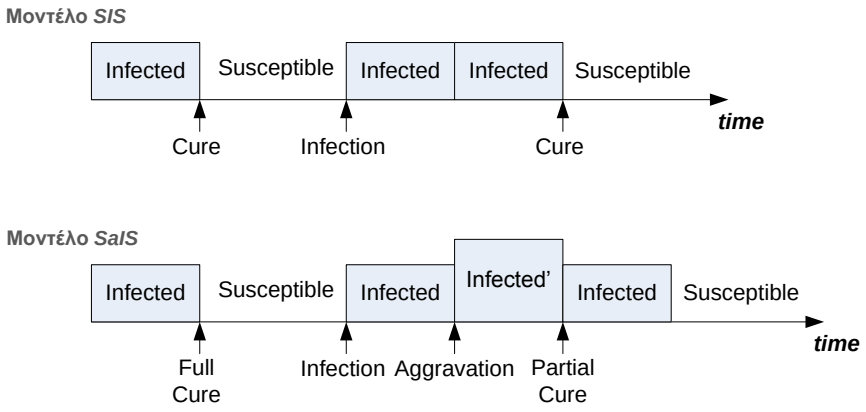
3.4 Μοντέλο SaIS

Κατά την παρουσίαση των ανωτέρω μοντέλων έγινε η σιωπηρή παραδοχή ότι ένας μολυσμένος κόμβος δύναται να μολυνθεί εκ νέου, δεδομένου ότι, στη γενικότερη περίπτωση, υφίστανται συσχετίσεις μεταξύ των δεδομένων πλαισίου (χρονικές ή σημασιολογικές εξαρτήσεις), οι οποίες πρέπει να ληφθούν υπόψη.

Διαισθητικά, ένας κόμβος παύει να ενδιαφέρεται για παρωχημένη πληροφορία ή πληροφορία γενικού περιεχομένου. Προκύπτει, συνεπώς, η ανάγκη ορισμού ισχυρότερων επιδημιών σε σχέση με αυτή που έχει ήδη μολύνει τον κόμβο. Οι ισχυρότερες επιδημίες βελτιώνουν την υφιστάμενη γνώση σύμφωνα με τη σχετική σημασιολογική διάρθρωση, δηλαδή με βάση τη σημασιολογική ιεραρχία ή/και τις χρονικές εξαρτήσεις (καλύτερης ποιότητας πληροφορία).

Το φαινόμενο της επαναμόλυνσης ενός κόμβου με μόλυνση ανώτερου επιπέδου (*μετάλλαξη*- transmutation) αναφέρεται μεταφορικά ως *επιδείνωση* (aggravation) της κατάστασης του κόμβου αυτού. Από την άλλη, η ανάρρωση ενός μολυσμένου κόμβου αναφέρεται στη μερική καλυτέρευση της κατάστασής του.

Το μοντέλο που περιγράφει το χαρακτηριστικό της επιδείνωσης καλείται SaIS (Susceptible- aggravation- Infected- Susceptible) και απεικονίζεται ως εξής:



Σχήμα 2: Η εναλλαγή των καταστάσεων ενός κόμβου (εξέλιξη στο χρόνο)

Στο μοντέλο SaIS η κατάσταση του κόμβου δύναται να επιδεινωθεί, ως αποτέλεσμα επαναμόλυνσης, λόγω μεταστοιχείωσης της επιδημίας [3]

Σύμφωνα με το ανωτέρω Σχήμα, ένας κόμβος είναι δυνατό να μολυνθεί με μια πληροφορία πλαισίου M_i είτε αν ο κόμβος αυτός ήταν υγιής την προηγούμενη χρονική στιγμή, είτε αν υπήρξε μολυσμένος με λιγότερο ποιοτική πληροφορία M (π.χ. η πληροφορία M δύναται να παραχθεί από τη M_i).

Η απλούστερη εκδοχή ποιοτικής αξιολόγησης της εισερχόμενης πληροφορίας βασίζεται στη διατήρηση του πλέον πρόσφατου διαθέσιμου δείγματος, επιτυγχάνοντας με αυτόν τον τρόπο καλύτερο συγχρονισμό του αποδέκτη με τα δεδομένα του κόμβου – πηγή, συνεπεία της μικρότερης χρονικής υστέρησης στο παρατηρούμενο μέγεθος. Αντιστοίχως, η υιοθέτηση πιο εκλεπτυσμένων μεθόδων προσανατολίζεται στην «παραγωγή» γνώσης, μέσω τεχνικών *συμπερασμού* (inference) και *συγχώνευσης δεδομένων* (data fusion) και έχει ως στόχο το σχηματισμό βελτιωμένης αντίληψης για το περιβάλλον συνολικά (*συνεργατική επίγνωση πληροφορίας πλαισίου* – collaborative context awareness). Και αυτό διότι, οι κόμβοι ανώτερου επιπέδου παράγουν διανθισμένη γνώση βάσει της κατανόησης που έχουν για το περιβάλλον και την επικοινωνούν σε γειτονικούς κόμβους, ορισμένοι από τους οποίους δεν διαθέτουν δυνατότητα μετασχηματισμού της προσλαμβάνουσας πληροφορίας.

Παραγωγοί και καταναλωτές παραμένουν λοιμώδεις μέχρι να καταστούν ανίσχυρα τα δεδομένα που φέρουν, οπότε λέγεται ότι *αναρρώνουν μερικώς* (partial cure). Στην περίπτωση αυτή, οι κόμβοι καταλήγουν με ανθεκτικότερη αλλά λιγότερο ποιοτική πληροφορία M', οι δε κόμβοι αυτοί παραμένουν μολυσματικοί με την πληροφορία M' έως ότου, είτε μολυνθούν εκ νέου με (ίδιου ή διαφορετικού επιπέδου) επιδημία, είτε αναρρώσουν, παρελθόντος κάποιου χρονικού διαστήματος. Στην τελευταία περίπτωση, οι κόμβοι έχουν αναρρώσει πλήρως και βρίσκονται σε κατάσταση υγιή.

Κάτωθι εμφανίζεται ο αλγόριθμος που περιγράφει το SaIS σχήμα:

```
if MSG Mi RECEIVED FOR THE FIRST TIME
then
        if M not VALID or
          Mi MORE VALUABLE THAN M              /* aggravation or context renewal */
        then
                STATE ← INFECTED
                M ← f(Mi, M)
```

```
        endif
endif
if STATE is INFECTED
then
        if random < β
        then
                TRANSMIT M TO ALL NEIGHBORS
        endif
        if M not VALID
        then
                if ∃ VALID M': M→M'
                then
                        M ← M'                          /* partial cure */
                else
                        STATE ← SUSCEPTIBLE        /* full cure */
                endif
        endif
endif
```

3.5 Μοντέλο aSIS

Κατά την παραμονή του κόμβου σε κατάσταση μολυσμένη είναι σημαντικό να επιβραβεύεται ή να περιορίζεται η μετάδοση της φέρουσας πληροφορίας –αναλόγως του επιπέδου αυτής– προκειμένου να ενισχυθεί η επιλεκτικότητα και να αποτρέπονται άσκοπες αποστολές μηνυμάτων. Και αυτό γιατί, εξαιτίας του πλήθους των *αλμάτων* (hops) που παρεμβάλλονται μεταξύ της γεννήτριας του δείγματος δεδομένων και του κόμβου προορισμού μέχρι την απόκτηση των δεδομένων από τον τελευταίο, η ποιότητα της τρέχουσας πληροφορίας ενδεχομένως να μην είναι αποδεκτά υψηλή. Για παράδειγμα, ένα μήνυμα που αφορά σε κυκλοφοριακή συμφόρηση είναι λογικό να θεωρηθεί χρήσιμο εντός μιας ακτίνας μερικών χιλιομέτρων από την πηγή πληροφόρησης και για χρονικό διάστημα μικρότερο της μιας ώρας.

Για να υπάρξει, συνεπώς, ελεγχόμενη και ουσιαστική αναπαραγωγή πληροφορίας, η εξάπλωση των δεδομένων πρέπει να διέπεται από τοπολογικούς/ χρονικούς κανόνες.

Αναφορικά με τους χωροταξικούς περιορισμούς, υποτιθέσθω ότι το δίκτυο αισθητήρων αναπτύσσεται σε περιβάλλον με διαχωριστικό, το οποίο όμως δεν αποτρέπει την επικοινωνία των δύο ομάδων κόμβων που σχηματίζονται από αυτό (λόγου χάρη, μια πόρτα που χωρίζει δύο δωμάτια). Το γεγονός ότι ο κόμβος που φέρει την ποιοτικότερη –σύμφωνα με τα ανωτέρω- πληροφορία ανήκει στην μία περιοχή θα πρέπει να ληφθεί υπόψη κατά τη διανομή των δεδομένων, καθώς το περιβάλλον στο οποίο αναφέρεται η εν λόγω πληροφορία ενδέχεται να μην ανταποκρίνεται στην κατάσταση της δεύτερης περιοχής. Η αντιμετώπιση τέτοιου είδους περιστατικών απαιτεί, εξ ορισμού, γνώση της τοπολογίας και των χαρακτηριστικών κίνησης των κόμβων του δικτύου, θέματα εκτός εμβέλειας της παρούσας εργασίας.

Όσον αφορά στους χρονικούς περιορισμούς, είναι προφανές πως οι παρωχημένες πληροφορίες δεν υπάρχει λόγος να μεταδίδονται στο δίκτυο, εξοικονομώντας επιπλέον ενέργεια αλλά και αυξάνοντας τη «χρονική ποιότητα». Κατ' αναλογία, οι πιο έγκυρες τιμές θα πρέπει να μεταδίδονται γρήγορα σε έναν μεγάλο αριθμό κόμβων, με στόχο τον απόλυτο συγχρονισμό των δεδομένων των κόμβων αναμετάδοσης με την τρέχουσα τιμή της πηγής.

Για την προσαρμογή της συμπεριφοράς του επιδημικού σχήματος στην ποιότητα της διακινούμενης πληροφορίας απαιτείται αντίστοιχη προσαρμογή των παραμέτρων του αλγόριθμου στην εντροπία του ρεύματος δεδομένων. Διότι, διαισθητικά, ραγδαία διακύμανση της υπό παρατήρηση μεταβλητής θα πρέπει να υποβάλει την πηγή σε αποστολή του δείγματος με μεγαλύτερη πιθανότητα μόλυνσης β_s, προκειμένου να υπάρξει ιδανικά καθολική μετάδοση της νέας, διαφοροποιημένης πληροφορίας στους γειτονικούς κόμβους. Αντιθέτως, ένα ομαλότερο ρεύμα δεδομένων θα πρέπει να οδηγεί τον πληθυσμό σε λιγότερο επιθετική συμπεριφορά εξάπλωσης, μειώνοντας μεν τη διεισδυτικότητα της νέας –όχι σημαντικά διαφοροποιημένης– πληροφορίας, προς όφελος όμως του ενεργειακού ισοζυγίου.

Ορίζεται, επομένως, αύξουσα συνάρτηση $f(.)$ για τον υπολογισμό της πιθανότητας εκπομπής της πηγής β_s, ως εξής:

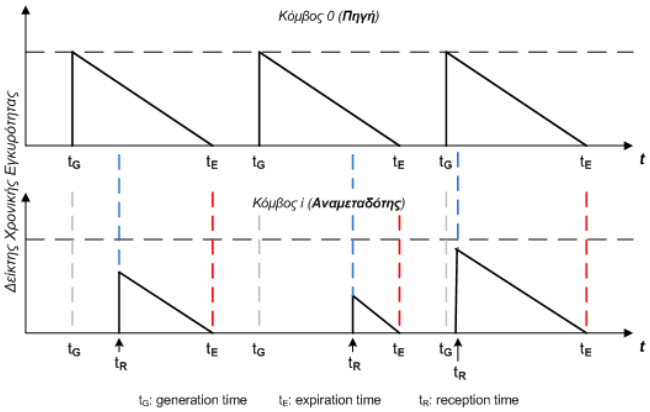
$$\beta_s(t) = f(\frac{\Delta x_0(t)}{\Delta t}),\ \ f: \mathbb{R} \rightarrow (0, 1]$$

όπου $x_0(t)$ η φέρουσα στον κόμβο 0 (πηγή) πληροφορία τη χρονική στιγμή t. Μεγάλη αλλαγή σε διαδοχικά δείγματα αυξάνει την πιθανότητα προώθησης του $x_0(t)$ στους γειτονικούς κόμβους, προκειμένου αυτοί να διαθέτουν ανά πάσα στιγμή όσο το δυνατό ποιοτικότερο αντίγραφο $x_i(t)$ του $x_0(t)$. Αντίθετα, η προώθηση μικρών μεταβολών του $x_0(t)$ δύναται να πραγματοποιείται με χαμηλότερο $\beta_s(t)$ χάριν οικονομίας στην κατανάλωση ενέργειας. Επισημαίνεται ότι δεν προβλέπεται ανάλογη προσαρμογή του ρυθμού μόλυνσης β_r των κόμβων αναμετάδοσης, καθώς, λόγω της στοχαστικής φύσης των επιδημιών, τα χαρακτηριστικά του ρεύματος $x_0(t)$ που αντιλαμβάνεται ο κόμβος πηγή διαφέρουν εν γένει από το εισερχόμενο στον τυχαίο κόμβο *i* ρεύμα $x_i(t)$. Ελλείψει ακριβών στατιστικών στοιχείων στους κόμβους αναμετάδοσης, η χρήση της συνάρτησης *f* θα αύξανε το συνολικό σφάλμα στο ασύρματο δίκτυο. Έτσι, η προώθηση των μηνυμάτων από τους κόμβους αναμετάδοσης γίνεται με σταθερό, προκαθορισμένο ρυθμό β_r.

Υπό το ίδιο πρίσμα επιχειρείται και η προσαρμογή του ρυθμού ίασης *δ* στα χαρακτηριστικά του περιβάλλοντος της επιδημίας.

Για το σκοπό αυτό, ορίζεται η παράμετρος χρονικής εγκυρότητας $v_i(t)$ της πληροφορίας $x_i(t)$ ως $v_i(t) = t_E - t$, όπου t_E (expiration time) ο χρόνος εκπνοής της ισχύος της πληροφορίας. Η εν λόγω μεταβλητή λαμβάνει τη μέγιστη τιμή της κατά την παραγωγή του δείγματος από τον κόμβο-πηγή τη χρονική στιγμή t_G (generation time, $t_G < t_E$) και αποστέλλεται στους κόμβους αναμεταδότες μαζί με το μήνυμα $x_0(t_G)$. Ο τυχαίος κόμβος αναμετάδοσης *i*, φορέας της πληροφορίας $x_i(t-1)$, λαμβάνει το μήνυμα $x_r(t)$ χρονικής εγκυρότητας $v_r(t)$, από έτερο κόμβο αναμετάδοσης ή απευθείας από την πηγή και αποφασίζει, βάσει κάποιου κριτηρίου ποιοτικής αξιολόγησης του εισερχόμενου μηνύματος (βλ. αλγόριθμο SaIS), για τη μόλυνση $x_i(t)$ που θα επικρατήσει. Εφόσον επιλεγεί επικαιροποίηση της φέρουσας πληροφορίας ($x_i(t) \leftarrow x_r(t)$), ενημερώνεται αντιστοίχως και η παράμετρος χρονικής εγκυρότητας ($v_i(t) \leftarrow v_r(t)$), ενώ στην περίπτωση που διατηρηθεί η τρέχουσα πληροφορία ($x_i(t) = x_i(t-1)$), αναπροσαρμόζεται σχετικά η υφιστάμενη εναπομένουσα διάρκεια ζωής ($v_i(t) = v_i(t-1) - 1$).

Σε κάθε περίπτωση, ο μολυσμένος κόμβος προωθεί με σταθερή πιθανότητα β_r ή/και επεξεργάζεται την πληροφορία $x_i(t)$ για όσο διάστημα παραμένει έγκυρη, δηλαδή εφόσον $v_i(t) > 0 \Leftrightarrow t_E > t$.



Σχήμα 3: Μεταβολή της τιμής χρονικής εγκυρότητας σε πηγή και αναμεταδότη, συναρτήσει του χρόνου [1]

Με την έλευση της χρονικής στιγμής $t = t_E$, οπότε και $v_i(t_E) = 0$, η πληροφορία θεωρείται παρωχημένη, παύει να ανακυκλώνεται στο δίκτυο και ο κόμβος αναρρώνει ντετερμινιστικά, εν αντιθέσει με τους απλούς επιδημικούς αλγόριθμους στους οποίους η ίαση γίνεται πιθανοθεωρητικά με πιθανότητα *δ*.

Προφανώς, υψηλή τιμή της παραμέτρου μέγιστης χρονικής εγκυρότητας $v_0(t_G)$ επηρεάζει αρνητικά την απόδοση του επιδημικού σχήματος, υποστηρίζει όμως την εξάπλωση σημαντικών δεδομένων σε όλο το ασύρματο δίκτυο αισθητήρων. Από την άλλη, δυσανάλογη ελάττωση της τιμής $v_0(t_G)$ έχει ως άμεση συνέπεια τον περιορισμό των περιττών αναμεταδόσεων, με τον κίνδυνο της ατελούς κάλυψης του δικτύου να ενισχύεται.

Προκύπτει, συνεπώς, η ανάγκη ρύθμισης της παραμέτρου $v_0(t_G)$ με γνώμονα την απόσταση που απαιτείται να διανύσει η νέα τιμή για την πλήρη κάλυψη του δικτύου. Η αποτίμηση της ακτίνας του δικτύου σε όρους χρονικής διάρκειας εμπλέκει την πιθανότητα μετάδοσης πηγής και αναμεταδοτών, δεδομένου του μη κατευθυνόμενου χαρακτήρα του επιδημικού μοντέλου και προς τούτο ορίζεται συνάρτηση $g(.)$ ως εξής:

$$v(t) = g(\beta_s(t), \beta_r), \;\; g: (0,1] \times (0,1] \rightarrow \mathbb{R}$$

Δημιουργείται, έτσι, μια δεδομενο-κεντρική παραλλαγή του κλασικού επιδημικού μοντέλου, η οποία καλείται aSIS (adaptive SIS) και περιγράφεται ως κάτωθι:

```
(πηγή)
if M ← SENSE()
then
        βs ← f(ΔM/Δt)
        V  ← g(βs, βr)                  /* temporal validity indicator tuning */
        if random < βs
        then
                TRANSMIT (M, V) TO ALL NEIGHBORS
        endif
```

```
endif

(αναμεταδότης)
V ← V-1                                        /* information ageing */
if MSG (Mi, Vi) RECEIVED
then
        if Mi MORE VALUABLE THAN M
        then
                STATE ← INFECTED
                M ← Mi
                V ← Vi
        else
            if V is less than or equal to 0            /* information expiration */
            then
                STATE ← SUSCEPTIBLE
            endif
        endif

        if STATE is INFECTED
        then
                if random < βr
                then
                        TRANSMIT (M, V) TO ALL NEIGHBORS
                endif
        endif
endif
```

Η επιλογή της κατάλληλης κάθε φορά υλοποίησης για τις $f(.)$ και $g(.)$ δύναται να πραγματοποιείται από μια βάση αποθηκευμένων μεθόδων που τηρεί η πηγή, ανάλογα με την υπολογιστική της δεινότητα (μνήμη, επεξεργαστική ισχύ) αλλά και την ευρύτερη φύση της εφαρμογής (κρισιμότητα, χρονική ευαισθησία κ.ά.).

4. ΑΛΓΟΡΙΘΜΟΣ ΜΕΤΑΒΛΗΤΗΣ ΣΥΧΝΟΤΗΤΑΣ ΔΕΙΓΜΑΤΟΛΗΨΙΑΣ

4.1 Περιγραφή Αλγόριθμου

Ανεξαρτήτως του επιπέδου αποτελεσματικότητας και αποδοτικότητας που διαμορφώνει ο αλγόριθμος μόλυνσης στο δίκτυο αισθητήρων, οι τιμές που αντιλαμβάνονται οι κόμβοι – αναμεταδότες ακολουθούν στην καλύτερη περίπτωση το πρωτογενές, αναλογικό ρεύμα δεδομένων με τον ίδιο βαθμό πιστότητας που ορίζεται από τον τρόπο επιλογής των δειγμάτων από τον κόμβο – πηγή.

Αυτό σημαίνει ότι ο ρυθμός δειγματοληψίας επηρεάζει σημαντικά την ποιότητα της διακινούμενης πληροφορίας, διότι η επιλογή μεγάλης περιοδικότητας οδηγεί νομοτελειακά σε φτωχή αποτύπωση του υπό παρατήρηση μεγέθους (τόσο για τον κόμβο – πηγή, πόσω δε μάλλον για τους λοιπούς κόμβους, στους οποίους κατά κύριο λόγο διοδεύεται μέρος του δειγματικού χώρου), η δε συχνή δειγματοληψία οριακά ωθεί την πηγή σε συνεχή παρακολούθηση του ρεύματος δεδομένων.

Οι περισσότεροι αλγόριθμοι υλοποιούν σταθερή συχνότητα δειγματοληψίας, με τακτικές, περιοδικές μετρήσεις του φυσικού μεγέθους, χωρίς να λαμβάνουν υπόψη τη μεταβλητότητα του σήματος. Ωστόσο, δεδομένου ότι η ποιότητα της διακινούμενης στο δίκτυο πληροφορίας προσδιορίζεται από την απόκλιση που παρατηρείται από την αντίστοιχη τιμή της μεταβλητής πραγματικού χρόνου, η εύστοχη επιλογή των αντιπροσωπευτικότερων δειγμάτων από τον κόμβο – πηγή (και για λογαριασμό του υπόλοιπου δικτύου) αποτελεί κρίσιμο παράγοντα επιτυχίας του σχήματος.

Με σκοπό την επιβολή της επιλεκτικότητας και την αποφυγή άσκοπων μεταδόσεων, προτείνεται η εφαρμογή μη ομοιόμορφης δειγματοληψίας, κατά την οποία τα χρονικά σημεία λήψης των τιμών δείγματος καθορίζονται από την αντίληψη που έχει σχηματίσει ο δειγματολήπτης για τις ιδιότητες του παρατηρούμενου μεγέθους. Ειδικότερα, εφόσον τη στιγμή της δειγματοληψίας εντοπισθεί έντονη διαφοροποίηση της παρατηρούμενης τιμής σε σχέση με το αμέσως προηγούμενο δείγμα, ο κόμβος – πηγή λαμβάνει απόφαση επόμενης δειγματοληψίας στο εγγύς μέλλον. Αντιστοίχως, εισερχόμενη τιμή με μικρή απόκλιση από την προηγούμενη έχει ως συνέπεια την επιμήκυνση του χρονικού διαστήματος επόμενης δειγματοληψίας. Με τον τρόπο αυτό ο ρυθμός δειγματοληψίας υπαγορεύεται από τη μεταβλητότητα του σήματος υπό παρακολούθηση, με συχνή δειγματοληψία στις περιπτώσεις που το ρεύμα δεδομένων παρουσιάζει έντονη διακύμανση και σποράδην δειγματοληψία σε σήματα με μικρή αβεβαιότητα τιμών.

Για τον υπολογισμό του ιδανικού κάθε φορά ρυθμού δειγματοληψίας $\mathrm{r_s}$ απαιτείται ο ορισμός φθίνουσας συνάρτησης $z(\cdot)$ ως εξής:

$$\mathrm{r_s(t)} = \mathrm{z}\left(\frac{\Delta \mathrm{x_0(t)}}{\Delta \mathrm{t}}\right), \mathrm{z}: \mathbb{R} \rightarrow \mathbb{R}^{+}, \text{ όπου}$$

- η εξαρτημένη μεταβλητή συμβολίζει τις χρονικές μονάδες που απαιτείται να μεσολαβήσουν μεταξύ διαδοχικών γεγονότων δειγματοληψίας
- το σύνολο τιμών της συνάρτησης δύναται να είναι φραγμένο. Δηλαδή, αναλόγως της φύσης της εφαρμογής, μπορούν να ορισθούν τιμές $\mathrm{r_{min}}$ ή/και $\mathrm{r_{max}}$, $\mathrm{r_{min}} \leq \mathrm{r_s(t)} \leq \mathrm{r_{max}}$, προκειμένου να διασφαλισθεί ότι η επόμενη δειγματοληψία θα συμβεί το νωρίτερο σε $\mathrm{r_{min}}$ ή/και το αργότερο σε $\mathrm{r_{max}}$ χρονικές μονάδες
- η υλοποίηση της συνάρτησης $z(\cdot)$ εξαρτάται από την επεξεργαστική ισχύ του δειγματολήπτη και τις απαιτήσεις για ταχύτητα της εφαρμογής. Σε κάθε περίπτωση, όμως, θα πρέπει να πληρούνται οι προϋποθέσεις του θεωρήματος Nyquist – Shannon.

4.2 Αξιολόγηση Αλγόριθμου

Δεδομένου ότι η επιλογή του μοντέλου δειγματοληψίας είναι ανεξάρτητη του αλγόριθμου επιδημίας που εφαρμόζεται κατά τη διάδοση των δειγμάτων στο δίκτυο αισθητήρων, η αποτίμηση της αποτελεσματικότητας και αποδοτικότητας του σχήματος μη ομοιόμορφης δειγματοληψίας δύναται να πραγματοποιηθεί μέσω σύγκρισης με τις αντίστοιχες επιδόσεις σχήματος ομοιόμορφης δειγματοληψίας για τον ίδιο αλγόριθμο επιδημικής μετάδοσης της πληροφορίας.

Προκειμένου, επομένως, να εκτιμηθούν τα σημεία υπεροχής του αλγόριθμου, κρίνεται απαραίτητος ο προσδιορισμός συγκεκριμένων μετρικών αξιολόγησης της επίδοσης, η ποσοτικοποίησή τους μέσω σχετικών προσομοιώσεων και η εν συνεχεία ευθεία αντιπαραβολή των παραγόμενων τιμών με τις αντίστοιχες του ανάλογου αλγόριθμου ομοιόμορφης δειγματοληψίας.

4.2.1 Ορισμός Μετρικών

Αρχικά, ως παράμετρος αποτελεσματικότητας του μοντέλου $\mathrm{n(t)}$ τη χρονική στιγμή *t* ορίζεται το κλάσμα των μολυσμένων κόμβων I(t) έναντι του πληθυσμού N

$$\mathrm{n(t)} = \frac{|\mathrm{I(t)}|}{\mathrm{N}}, \text{ όπου } \mathrm{I(t)} = \{\mathrm{i} \in \mathrm{N}, \mathrm{v_i(t)} > 0\}$$

Δεδομένου ότι ο λόγος αυτός περιγράφει το πλήθος των κόμβων που φέρουν χρήσιμη πληροφορία, ιδεατά οι τιμές που λαμβάνει θα πρέπει να τείνουν στη μονάδα. Σε κάθε περίπτωση, όμως, η παράμετρος είναι πάντα μη μηδενική, υπό το πρίσμα ότι ο κόμβος – πηγή εξορισμού διαθέτει έγκυρα δεδομένα.

Η τιμή της μετρικής $\mathrm{n(t)}$ αποτελεί τη βάση και για την αποτίμηση της αποδοτικότητας $\mathrm{u(t)}$. Πιο συγκεκριμένα, ως αποδοτικότητα του δικτύου τη χρονική στιγμή *t* ορίζεται ο λόγος των μολυσμένων κόμβων $\mathrm{n(t)}$ σε σχέση με το πλήθος των μηνυμάτων που έχουν μέχρι στιγμής διακινηθεί στο δίκτυο. Δηλαδή,

$$\mathrm{u(t)} = \frac{\mathrm{n(t)}}{\mathrm{m(t)}}, \text{ όπου } \mathrm{m(t)} = \sum_{\mathrm{r=1}}^{\mathrm{t}} \frac{1}{\mathrm{N}} * \sum_{\mathrm{i=1}}^{\mathrm{N}} \mathrm{m_i(r)}$$

και $\mathrm{m_i(t)}$ το πλήθος των μηνυμάτων $\mathrm{x_i(t)}$ που προωθεί ο κόμβος *i* το χρόνο *t*. Υψηλές τιμές $\mathrm{u(t)}$ υποδηλώνουν μεγάλη αποτελεσματικότητα με περιορισμένες ζεύξεις επικοινωνίας, ενώ οι αιτίες χαμηλής αποδοτικότητας θα πρέπει να αναζητηθούν είτε στη χαμηλή αποτελεσματικότητα του αλγόριθμου είτε στην αλόγιστη διακίνηση της πληροφορίας στο δίκτυο.

Ωστόσο, η αξιολόγηση των δύο ανωτέρω παραμέτρων δεν εξετάζει την πληροφορία από τη σκοπιά της ποιότητας. Είναι δυνατό να υπάρξει μεγάλο ποσοστό μολυσμένων κόμβων οι οποίοι όμως να ανακυκλώνουν απαρχαιωμένη, χαμηλού επιπέδου πληροφορία.

Για το λόγο αυτό διαμορφώνεται παράμετρος $\mathrm{e_i(t)}$, η οποία συμβολίζει το βαθμό σφάλματος που εισάγει κάθε κόμβος στο δίκτυο. Λαμβάνοντας υπόψη ότι πρωταρχικός στόχος των κόμβων σε ένα δίκτυο αισθητήρων είναι η όσο το δυνατό πιστότερη αποτύπωση του υπό παρατήρηση φυσικού μεγέθους, ως δείκτης σφάλματος $\mathrm{e_i(t)}$ ορίζεται η ποσοστιαία απόκλιση της τιμής $\mathrm{x_i(t)}$ του κόμβου *i* τη χρονική στιγμή *t* από την πραγματική τιμή του φυσικού μεγέθους $\mathrm{x(t)}$ την ίδια χρονική στιγμή, ήτοι:

$$\mathrm{e_i(t)} = \left|\frac{\mathrm{x_i(t)} - \mathrm{x(t)}}{\mathrm{x(t)}}\right| \ \forall \mathrm{i} \in [0, \mathrm{N} - 1]$$

Είναι προφανές ότι κατά τις χρονικές στιγμές δειγματοληψίας t_s, ισχύει ότι $e_0(t_s)$=0, αφού $x_0(t_s) = x(t_s)$. Κατά τα λοιπά, η μεταβλητή σφάλματος $e_0(t)$ λαμβάνει τόσο μεγαλύτερες τιμές όσο μεγαλώνει η μεταβλητότητα του ρεύματος δεδομένων. Διότι, κατά το διάστημα Δt < $r_s(t_s)$ που μεσολάβησε από την προηγούμενη λήψη δείγματος τη χρονική στιγμή t_s, η τιμή του συνεχούς ρεύματος δεδομένων διαμορφώθηκε σε x(t_s + Δt), με το δείκτη σφάλματος να υπολογίζεται σε

$$e_0(t_s + \Delta t) = \left|\frac{x_0(t_s + \Delta t) - x(t_s + \Delta t)}{x(t_s + \Delta t)}\right| = \left|\frac{x(t_s) - x(t_s + \Delta t)}{x(t_s + \Delta t)}\right|$$

Γίνεται έτσι φανερό ότι το σφάλμα που αναπόφευκτα εισάγεται στο δίκτυο κατά την επιλογή δείγματος εξαρτάται από τη διαφοροποίηση του $x(t_s + \Delta t)$ που επήλθε κατά το διάστημα Δt επί της τιμής $x(t_s)$.

Μεγαλύτερες τιμές σφάλματος αναμένεται να εντοπισθούν στους λοιπούς κόμβους του δικτύου, καθώς πέραν της ανωτέρω απόκλισης της πιο πρόσφατης τιμής από το πραγματικό ρεύμα δεδομένων, εισάγεται πρόσθετο σφάλμα λόγω του χρονικού διαστήματος που παρέρχεται για τη λήψη της τιμής t_s από τον τυχαίο κόμβο – αναμεταδότη. Καθώς είναι λογικό, όσο ταχύτερη είναι η μετάδοση του δείγματος $x_0(t_s)$ στους κόμβους – αναμεταδότες, τόσο η τιμή της μετρικής σφάλματος θα συγκλίνει στην αντίστοιχη τιμή της πηγής, $e_i(t) \rightarrow e_0(t)$.

Βάσει του δείκτη σφάλματος των επιμέρους κόμβων ορίζεται και η μέση τιμή σφάλματος του δικτύου τη χρονική στιγμή t, e(t), ως εξής

$$e(t) = \frac{1}{t} * \sum_{r=1}^{t} (\frac{1}{N} * \sum_{i=1}^{N} e_i(r))$$

Η παράμετρος $e(t)$ πρέπει να λαμβάνει μικρές τιμές, προκειμένου ο τυχαίος κόμβος *i* να είναι επαρκώς συγχρονισμένος με το περιβάλλον.

Τέλος, για να προσδιορισθεί το ποσοστό των κόμβων του δικτύου που φέρει ενημερωμένα, υψηλού επιπέδου δεδομένα, ορίζεται η παράμετρος ακρίβειας $\zeta(t)$ ως

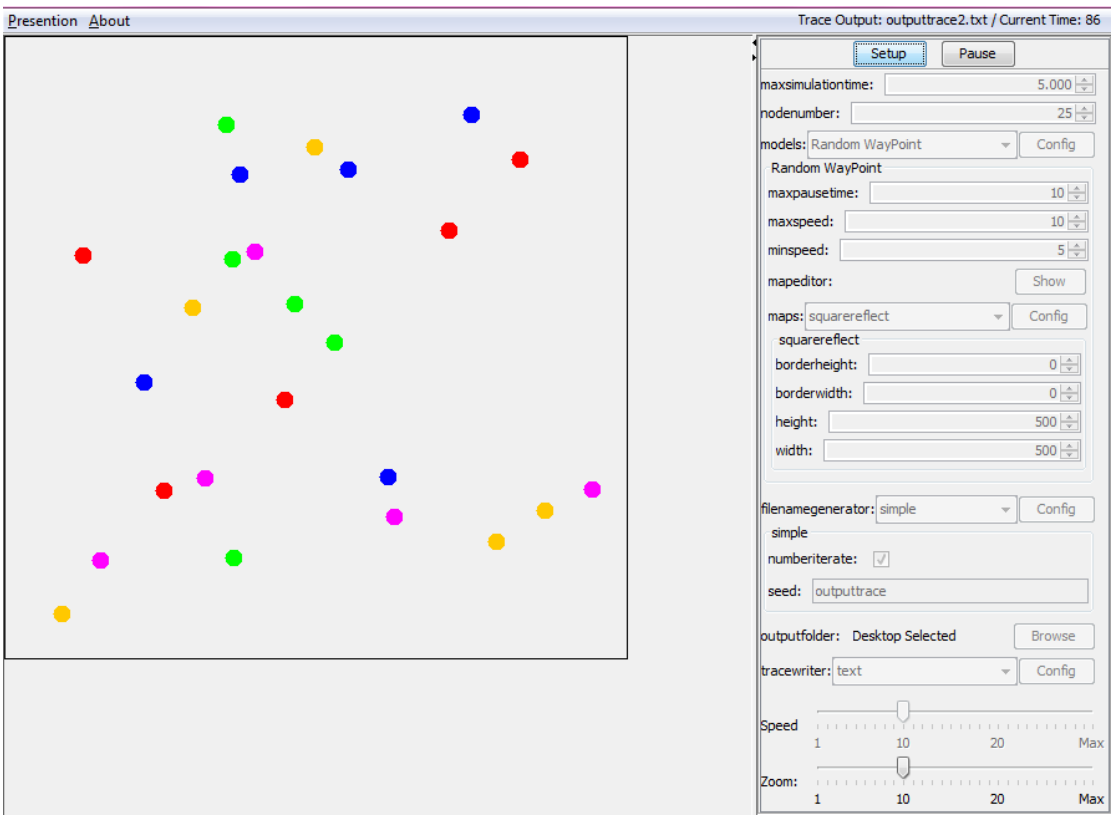
$$\zeta(t) = n(t) * (1 - e(t))$$

Υψηλές τιμές για την παράμετρο ακρίβειας υποδηλώνουν υψηλή αποτελεσματικότητα για το μοντέλο σε συνδυασμό με καλό συγχρονισμό των δεδομένων των κόμβων με την εξωτερική πληροφορία.

4.2.2 Μοντέλο Προσομοίωσης

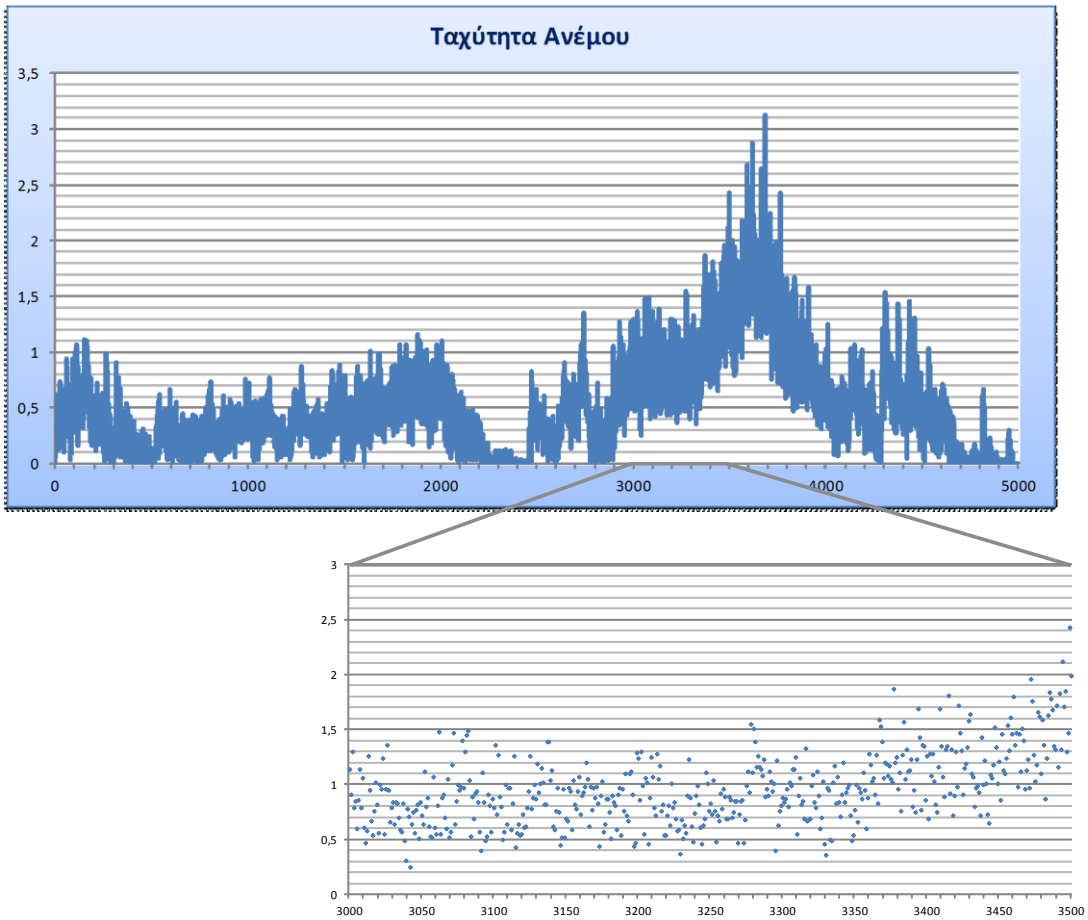
Η συγκριτική μελέτη του σχήματος μη ομοιόμορφης δειγματοληψίας που θα ακολουθήσει βασίζεται στην υλοποίηση του αλγόριθμου aSIS σε Java που παρουσιάζεται στο Παράρτημα Ι.

Πιο συγκεκριμένα, έστω δίκτυο διαστάσεων 250.000 τετραγωνικών μονάδων (500X500) πληθυσμού N=25 κόμβων, με τον κόμβο 0 να αποτελεί, χωρίς βλάβη της γενικότητας, το μοναδικό κόμβο – πηγή και τους υπόλοιπους i=1…24 να αποτελούν τους κόμβους – αναμεταδότες.



Εικόνα 1: Στιγμιότυπο της θέσης των κόμβων του δικτύου προσομοίωσης (αναπαράσταση μέσω Mobisim) [10]

Έστω, επίσης, αναλογικό σήμα πραγματικού χρόνου το οποίο, με χρήση πολύ υψηλής συχνότητας δειγματοληψίας, αναπαρίσταται από μια σειρά 5.000 διακριτών τιμών. Δεδομένου ότι δύο διαδοχικές τιμές της σειράς θεωρείται ότι απέχουν μεταξύ τους στοιχειώδες χρονικό διάστημα, η γραφική παράσταση του σήματος δειγματοληψίας προσομοιάζει οριακά το αρχικό, συνεχές σήμα ως εξής:



Σχήμα 4: Αρχικό, αναλογικό σήμα εισόδου στον κόμβο - πηγή

Έχοντας, λοιπόν, στη διάθεσή του ένα «*συνεχο-φανές*» (continuous-like) σήμα, ο κόμβος 0 επιλέγει, βάσει της τρέχουσας τιμής της παραμέτρου r_s, τη χρονική στιγμή λήψης επόμενου δείγματος. Για την περίπτωση της ομοιόμορφης δειγματοληψίας, το r_s ορίζεται σε 5 χρονικές μονάδες, που σημαίνει ότι ο κόμβος-πηγή αξιοποιεί κάθε πέμπτη τιμή από τη σειρά των 5.000 τιμών του σήματος προσομοίωσης, ενώ για την περίπτωση της μη ομοιόμορφης δειγματοληψίας, το r_s υπολογίζεται από τον τύπο

$$r_s(t) = 1 + 10 * 0{,}35^{1{,}5*\frac{\Delta x_0(t)}{\Delta t}}$$

ο οποίος προκύπτει ότι φράσσεται από τις τιμές r_{min} = 1 και r_{max} = 11 χρονικές μονάδες και επιλέχθηκε για λόγους απλότητας στην υλοποίηση.

Μόλις αποκτηθεί η προγραμματισμένη τιμή δείγματος, καθορίζονται βάσει αυτής οι αντίστοιχες τιμές στις παραμέτρους v, β_s και –για την περίπτωση του νέου αλγόριθμου- r_s και την επόμενη χρονική μονάδα το μήνυμα $(x_s(t), v_s(t), t)$ διαχέεται στο δίκτυο με πιθανότητα

$$\beta_s(t) = \frac{1}{1 + e^{-12*(\frac{\Delta x_0(t)}{\Delta t} - 0{,}3)}}$$

Εν συνεχεία, η πηγή αναμένει την πάροδο των r_s χρονικών μονάδων, οπότε και επαναλαμβάνει την ίδια διαδικασία.

Όταν οι λοιποί κόμβοι του δικτύου δεχθούν εισερχόμενο μήνυμα από τους γειτονικούς τους κόμβους, συγκρίνουν την ποιότητα της φέρουσας πληροφορίας με αυτήν της νέας τιμής και λαμβάνουν αποφάσεις για τα δεδομένα που θα επικρατήσουν. Την επόμενη χρονική στιγμή και εφόσον η φέρουσα πληροφορία είναι ακόμα εν ισχύ, δημιουργείται η

τριπλέτα $(x_r(t), v_r(t), t)$ και προωθείται με σταθερή πιθανότητα β_r = 0,5 στους κόμβους που βρίσκονται εντός της ακτίνας επικοινωνίας τους.

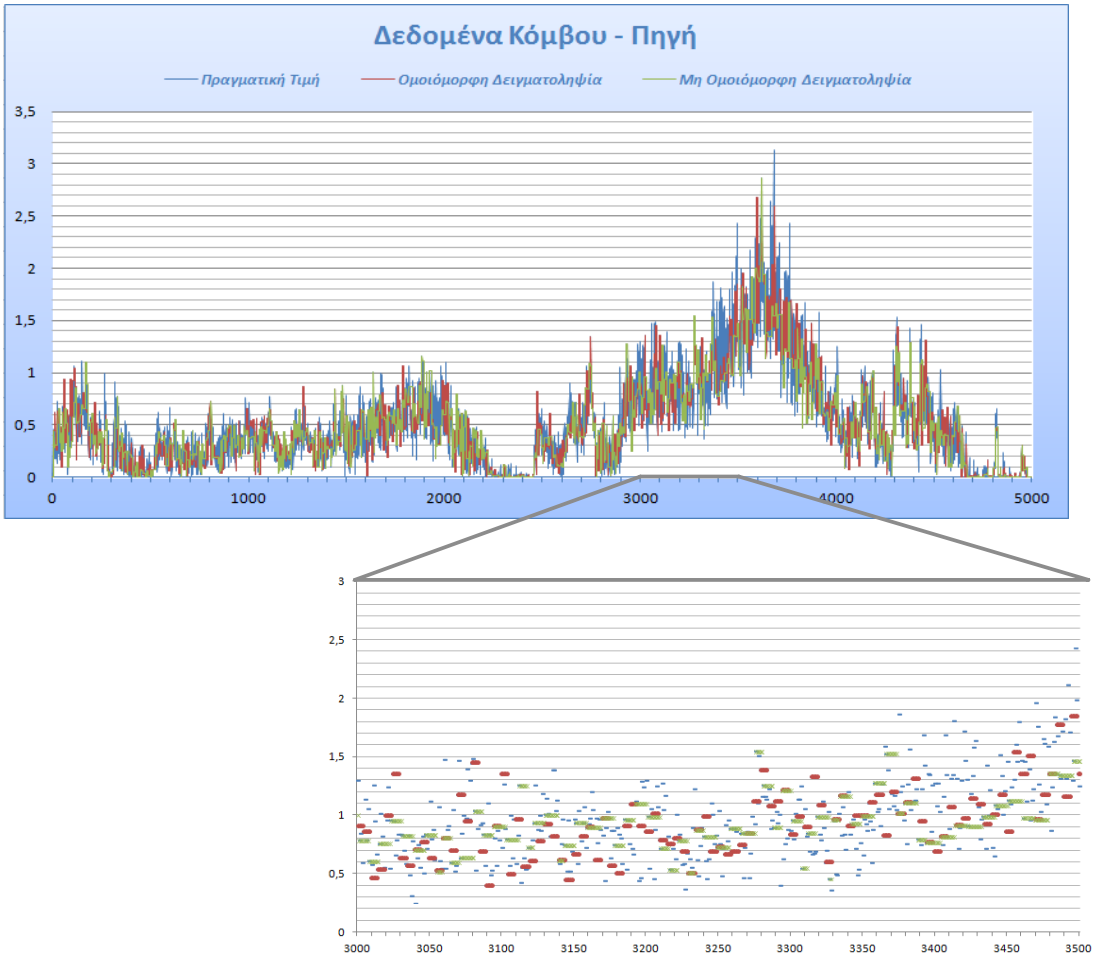
Με τον τρόπο αυτό, αλλά και με τη βοήθεια της διαρκούς πλοήγησης των κόμβων εντός του πλαισίου προσομοίωσης κάθε χρονική στιγμή (βλ. Παράρτημα ΙΙ), η επιδημία εξαπλώνεται σταδιακά από ζεύξη σε ζεύξη σε όλο το δίκτυο και οι κόμβοι γίνονται τελικά κοινωνοί της αντίληψης που φέρει η εκάστοτε γειτονιά τους για το υπό παρατήρηση ρεύμα δεδομένων.

Πίνακας 1: Πίνακας Παραμέτρων Προσομοίωσης

Παράμετρος	***Συμβολισμός***	***Τιμή***
Πλήθος κόμβων	N	25 (1 πηγή – 24 αναμεταδότες)
Πιθανότητα προώθησης πηγής	β_s	$\beta_s(t) = \frac{1}{1+e^{-12*(\frac{\Delta x_0(t)}{\Delta t}-0,3)}} \in (0, 1]$
Πιθανότητα προώθησης αναμεταδοτών	β_r	0,5
Δείκτης εγκυρότητας	v_s	$v_s(t) = \frac{7}{5^{\beta_s * \beta_r}} \in [4, 7]$
Περίοδος δειγματοληψίας	r_s	➢ 5, για ομοιόμορφη δειγματοληψία ➢ $r_s(t) = 1 + 10 * 0,35^{1,5*\frac{\Delta x_0(t)}{\Delta t}} \in [1, 11]$, για μη ομοιόμορφη δειγματοληψία
Ακτίνα επικοινωνίας	r	100
Κύκλοι προσομοίωσης		5.000 χρονικές μονάδες

4.2.3 Αποτελέσματα Προσομοίωσης

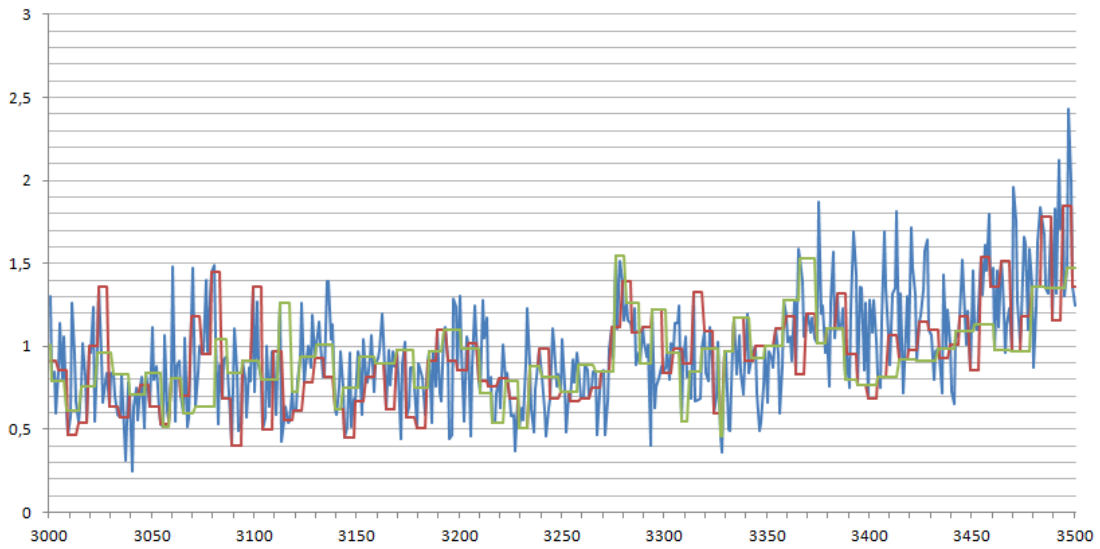
Το αποτέλεσμα εφαρμογής ομοιόμορφης και μη ομοιόμορφης δειγματοληψίας στο συνεχές σήμα που παρουσιάσθηκε στο Σχήμα 4 απεικονίζεται στο κάτωθι Σχήμα:



Σχήμα 5: Αποτέλεσμα εφαρμογής ομοιόμορφης (κόκκινη γραμμή) και μη ομοιόμορφης (πράσινη γραμμή) δειγματοληψίας στο αρχικό σήμα (μπλε γραμμή) από τον κόμβο - πηγή

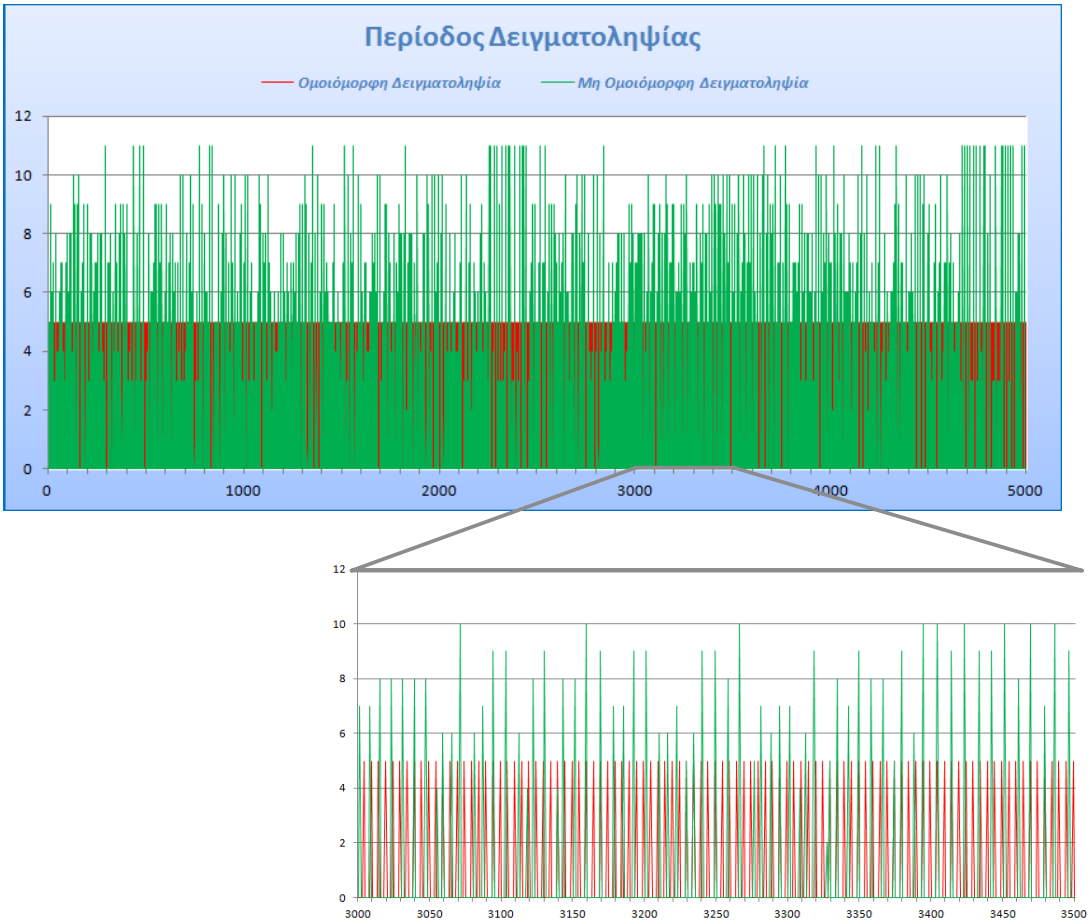
Εν πρώτοις, δεν εντοπίζονται ουσιαστικές διαφοροποιήσεις στον τρόπο με τον οποίο οι κόμβοι – πηγή αντιλαμβάνονται το περιβάλλον, τόσο μεταξύ των δύο προσεγγίσεων δειγματοληψίας, όσο και σε αντιπαραβολή με το πραγματικό, συνεχές σήμα (το οποίο ενσωματώθηκε με μπλε γραμμή χάριν αναγνωσιμότητας).

Η ίδια εικόνα επαληθεύεται και μικροσκοπικά. Στο Σχήμα που ακολουθεί αποτυπώνεται το ρεύμα δεδομένων που αισθάνεται ένας κόμβος ομοιόμορφης και ένας κόμβος μη ομοιόμορφης δειγματοληψίας (κόκκινη και πράσινη γραμμή, αντιστοίχως) εντός μιας περιόδου 500 χρονικών μονάδων.



Σχήμα 6: Μικροσκοπική ανάλυση του σήματος που αντιλαμβάνεται ο κόμβος – πηγή, ως αποτέλεσμα εφαρμογής ομοιόμορφης (κόκκινη γραμμή) και μη ομοιόμορφης (πράσινη γραμμή) δειγματοληψίας (στιγμιότυπο από 3.000 έως 3.500 χρονικές μονάδες)

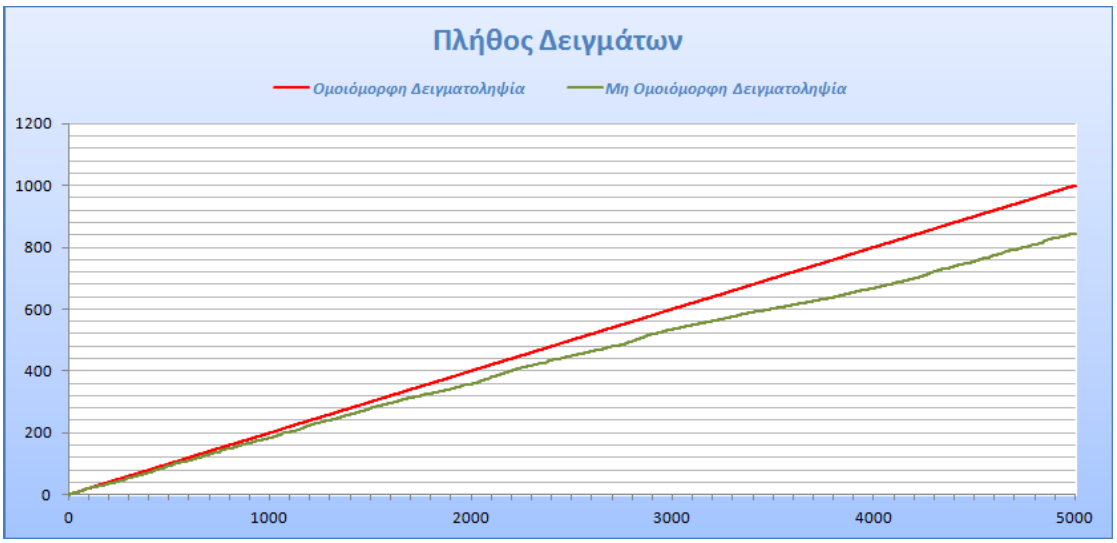
Πέραν της αναμενόμενης απώλειας πληροφορίας λόγω ψηφιοποίησης, τα δύο σήματα φαίνεται πως παρακολουθούν σε ικανοποιητικό βαθμό την τάση της αναλογικής μετάδοσης. Μάλιστα, στην περίπτωση της μη ομοιόμορφης δειγματοληψίας, αυτή η εναρμόνιση επιτυγχάνεται με «βέλτιστο» αριθμό δειγμάτων, καθώς βασική επιδίωξη του προτεινόμενου αλγόριθμου είναι η προσαρμογή της περιόδου δειγματοληψίας στην εντροπία του παρατηρούμενου μεγέθους.



Σχήμα 7: Χρονική διακύμανση της περιόδου δειγματοληψίας r_s, με $r_s(0) = 5$

Έτσι, σε περιόδους με έντονη μεταβλητότητα στο υπό παρακολούθηση ρεύμα δεδομένων (π.χ. κατά το διάστημα από 3.250 έως 3.300) η χρονική απόσταση μεταξύ δύο διαδοχικών παρατηρήσεων μειώνεται ραγδαία. Κατ' αναλογία, σε περιόδους στασιμότητας (π.χ. κατά το διάστημα από 3.150 έως 3.200) τα δείγματα επιλέγονται με μικρή συχνότητα, με αποτέλεσμα αφενός μεν τον περιορισμό της ενεργειακής δαπάνης που απαιτείται για τη λειτουργία του αισθητήρα, αφετέρου δε τη διακίνηση της πραγματικά ωφέλιμης πληροφορίας στο δίκτυο.

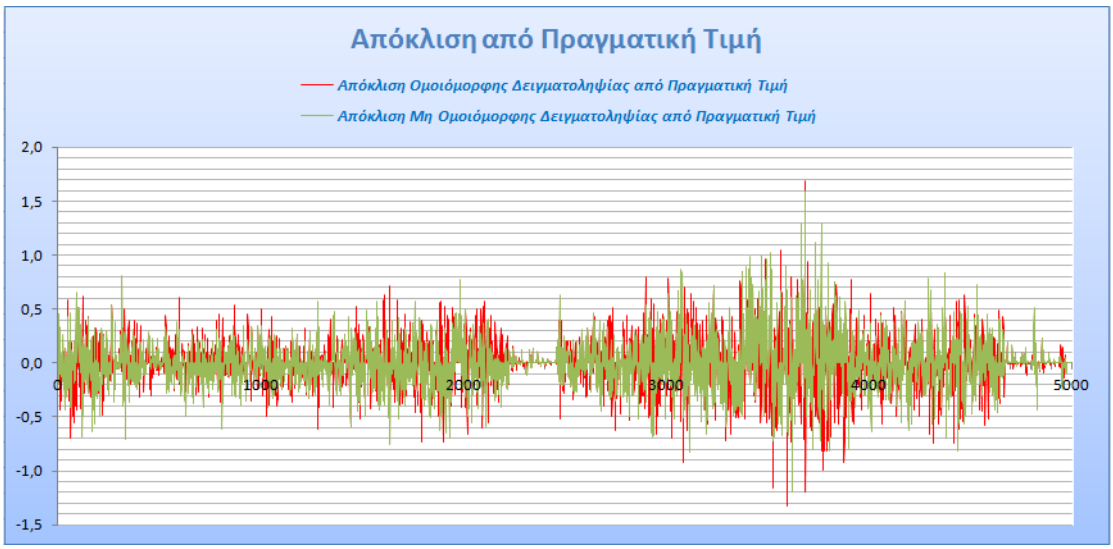
Είναι σαφές, επομένως, ότι για δεδομένες υλοποιήσεις ομοιόμορφης και μη ομοιόμορφης δειγματοληψίας, η ομαλότητα του ρεύματος δεδομένων επιδρά στη μέση τιμή συχνότητας δειγματοληψίας και αυτή με τη σειρά της καθορίζει το πλήθος των δειγμάτων που αποκτούν οι κόμβοι – πηγή σε ορισμένο χρονικό διάστημα. Για το σήμα του Σχήματος 4, η μέση περίοδος μη ομοιόμορφης δειγματοληψίας υπολογίζεται σε 5,92 χρονικές μονάδες οδηγώντας σε 844 επιλογές τιμών μέσα σε έναν χρονικό ορίζοντα 5.000 μονάδων, σε αντιπαραβολή με τα 1.000 δείγματα του αλγόριθμου ομοιόμορφης δειγματοληψίας, περιόδου 5 χρονικών μονάδων.

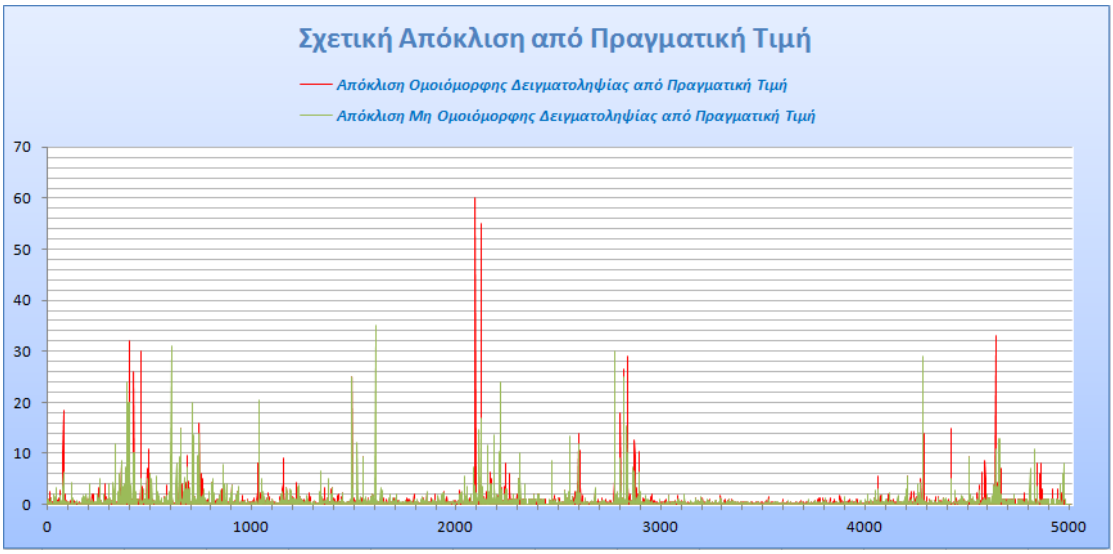


Σχήμα 8: Αθροιστικός δείκτης φορών δειγματοληψίας συναρτήσει του χρόνου

Στην περίπτωση που το πραγματικό, αναλογικό σήμα εμφανίζει εντονότερη διακύμανση, ο αλγόριθμος μη ομοιόμορφης δειγματοληψίας θα προσαρμόσει την μέση περίοδο σε χαμηλότερα επίπεδα για την καλύτερη παρακολούθηση του περιβάλλοντος, ενώ στην οριακή περίπτωση σταθερού αναλογικού σήματος, η μέση περίοδος δειγματοληψίας θα λάβει τη μέγιστη τιμή της (11 χρονικές μονάδες) αποδίδοντας $\frac{5000}{11} = 455$ δείγματα, εν αντιθέσει με τα 1000 δείγματα του αλγόριθμου ομοιόμορφης δειγματοληψίας.

Ωστόσο, η ως άνω βελτίωση της αποδοτικότητας δεν πρέπει να γίνει σε βάρος της ποιότητας της αντίληψης που σχηματίζει ο κόμβος – πηγή για το περιβάλλον. Η απόκλιση της φέρουσας πληροφορίας από την τρέχουσα τιμή του σήματος πρέπει να είναι, στη χειρότερη περίπτωση, συγκρίσιμη με τον αντίστοιχο θόρυβο που εισάγει στο δίκτυο ο αλγόριθμος ομοιόμορφης δειγματοληψίας.





Σχήμα 9: Απόκλιση (πάνω) και σχετική απόκλιση (κάτω) των δεδομένων του κόμβου – πηγή από την πραγματική τιμή, μετά την εφαρμογή δειγματοληψίας

Για την επαλήθευση της αποτελεσματικότητας του προτεινόμενου αλγόριθμου, επομένως, δημιουργήθηκε το γράφημα του Σχήματος 9α, το οποίο παριστάνει την απόκλιση των τιμών των κόμβων – πηγή από την πραγματική τιμή του φυσικού μεγέθους για τα σχήματα ομοιόμορφης και μη ομοιόμορφης δειγματοληψίας. Χονδρικά είναι δύσκολο να προκριθεί η μία προσέγγιση έναντι της άλλης, καθώς οι τιμές που λαμβάνει διαχρονικά το σφάλμα και στις δύο περιπτώσεις ακολουθούν παρόμοια κυμάτωση. Επιπροσθέτως, οι εξάρσεις των δύο καμπυλών ασφαλώς αποτελούν ένδειξη ότι οι κόμβοι – πηγή απώλεσαν τμήμα της περιρρέουσας πληροφορίας, ωστόσο η ποσοτικοποίηση του απόλυτου σφάλματος δεν οδηγεί σε αξιόπιστα συμπεράσματα. Και αυτό γιατί η έντονη διαφοροποίηση στις τιμές που αντιλαμβάνεται η πηγή σε σχέση με το συνεχές ρεύμα δεδομένων μπορεί να αποδοθεί είτε στον κακό συγχρονισμό της πηγής, είτε όμως να οφείλεται στην τάξη μεγέθους της παρατηρούμενης ποσότητας.

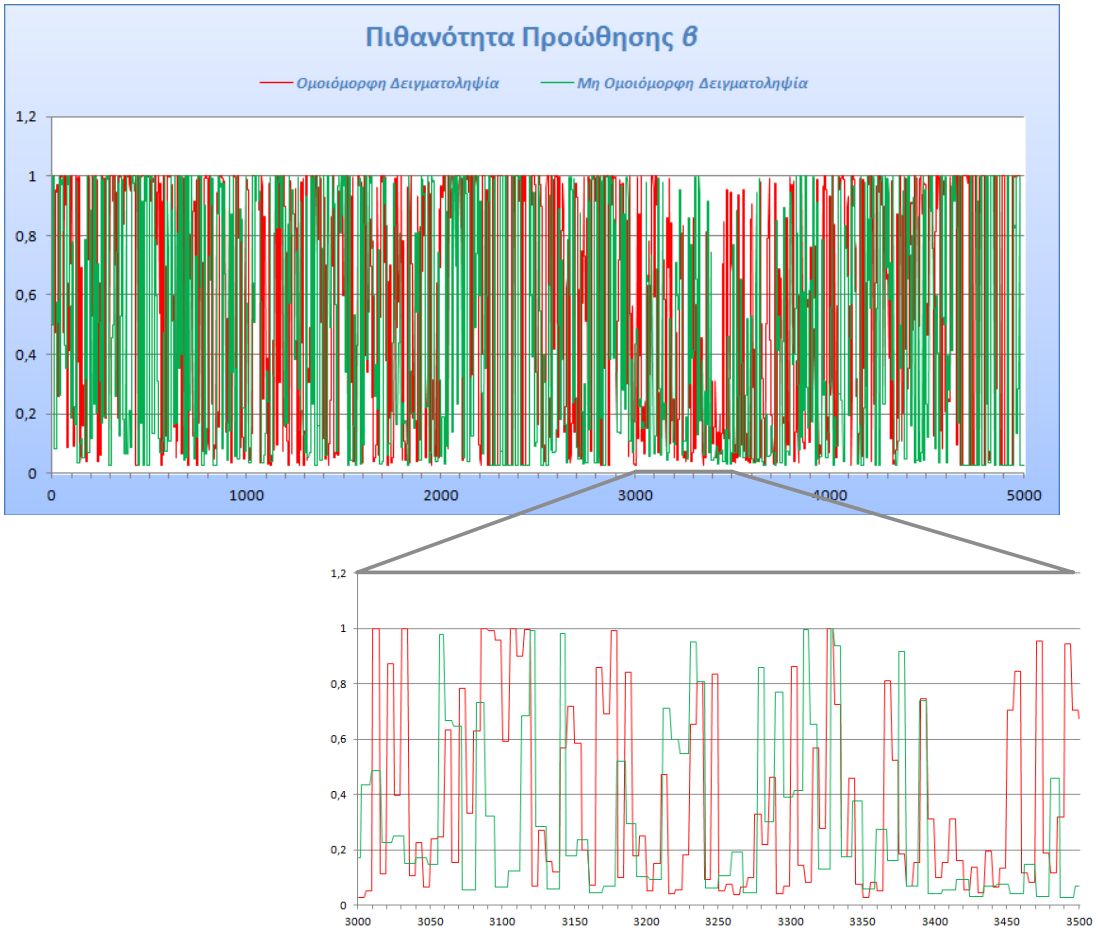
Με σκοπό τον περιορισμό της επιρροής της μονάδας μέτρησης του σήματος στη μελέτη του σφάλματος, χρησιμοποιείται συμπληρωματικά η μετρική του σχετικού σφάλματος, όπως αυτή υπολογίσθηκε και αποτυπώνεται στο διάγραμμα του Σχήματος 9β. Και πάλι τα δύο σχήματα παρουσιάζουν παραπλήσιες τιμές σχετικής απόκλισης, με τη μέθοδο μη ομοιόμορφης δειγματοληψίας να εμφανίζει εν γένει περισσότερα σημεία διαφορών αλλά ο αλγόριθμος ομοιόμορφης δειγματοληψίας να έχει μεγαλύτερα ανώτατα σημεία. Ωστόσο, η υπεροχή του αλγόριθμου μη ομοιόμορφης δειγματοληψίας ως προς το

σκέλος της ακρίβειας των δειγμάτων γίνεται εμφανέστερη, αν ληφθεί υπόψη και η δυνατότητα προσαρμογής της αναμενόμενης σχετικής απόκλισης μέσω φραγής της μέγιστης τιμής περιόδου δειγματοληψίας. Δηλαδή, σε περιπτώσεις ευαίσθητων σε σφάλματα εφαρμογών, η υιοθέτηση χαμηλότερης τιμής για την παράμετρο r_{max} συνεπάγεται (μέσω της σχέσης ορισμού του $r_s(t)$) αντίστοιχη μείωση της πιθανότητας εμφάνισης ακραίας τιμής σφάλματος.

Σε κάθε περίπτωση, πάντως, και όπως είναι αναμενόμενο, οι αποκλίσεις των τιμών των κόμβων – πηγή από το πραγματικό ρεύμα δεδομένων εκμηδενίζονται κατά τις στιγμές δειγματοληψίας, ενώ η έκταση του σφάλματος κατά τις λοιπές χρονικές στιγμές εξαρτάται από τη μεταβλητότητα του ρεύματος δεδομένων, αλλά και από το διάστημα που έχει παρέλθει από την πιο πρόσφατη δειγματοληψία.

Πέραν, όμως, της σημαντικότητας που έχει η ακολουθία των δειγμάτων του κόμβου – πηγή ως η καλύτερη δυνατή προσέγγιση του αρχικού σήματος που μπορεί να κυκλοφορήσει στο δίκτυο, διαδραματίζει πρόσθετα σπουδαίο ρόλο και στον καθορισμό των χαρακτηριστικών μόλυνσης του επιδημικού δικτύου, λόγω της συσχέτισης του κλάσματος μεταβολής δύο διαδοχικών δειγμάτων με τις παραμέτρους β_s και v. Εξάλλου φαίνεται λογικό όταν η τρέχουσα τιμή του κόμβου – πηγή διαφοροποιείται αισθητά από την αντίστοιχη προηγούμενη, το νέο δείγμα να προωθείται με μεγάλη πιθανότητα στο δίκτυο με γνώμονα την πιστή παρακολούθηση των μεταβολών του σήματος από όλους τους κόμβους. Αντίστοιχα, σε περιπτώσεις παραπλήσιων διαδοχικών δειγμάτων και υπό το πρίσμα ότι η πλειοψηφία των κόμβων – αναμεταδοτών έχει ήδη μολυνθεί κατά το παρελθόν με παρόμοια τιμή, το ενδεχόμενο νέας επιμόλυνσης περιορίζεται, χάριν αποδοτικότητας του αλγόριθμου.

Η ανωτέρω αρχή εφαρμόζεται ανεξαρτήτως τρόπου απόκτησης των δειγμάτων και για τα σήματα των κόμβων ομοιόμορφης και μη ομοιόμορφης δειγματοληψίας του Σχήματος 5 παράγει τις κάτωθι τιμές πιθανότητας μόλυνσης β_s.

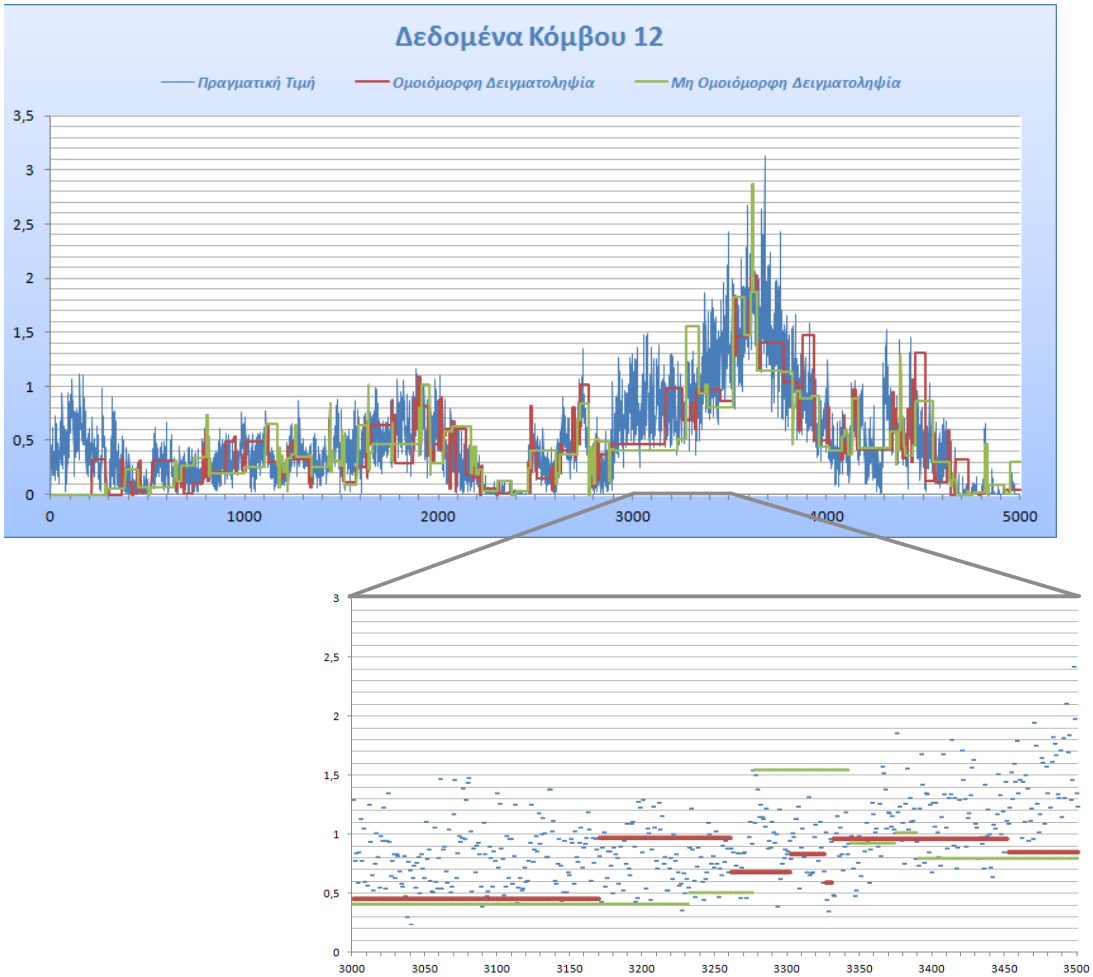


Σχήμα 10: Χρονική διακύμανση της πιθανότητας προώθησης β_s του κόμβου - πηγή

Σε γενικές γραμμές, οι δύο καμπύλες δείχνουν να εμφανίζουν παρόμοια χαρακτηριστικά, με τη μέση τιμή πιθανότητας προώθησης β_s για την ομοιόμορφη δειγματοληψία να ανέρχεται σε 0,5924757559 ενώ για τη μη ομοιόμορφη δειγματοληψία σε 0,5980260144. Ελάχιστα εντονότερη διαφοροποίηση μεταξύ των δύο σχημάτων παρατηρείται στην τιμή της διαμέσου, με τιμές 0,7071937274 και 0,7467594216 αντιστοίχως, ενώ πλήρης ταύτιση παρατηρείται στις επικρατούσες τιμές.

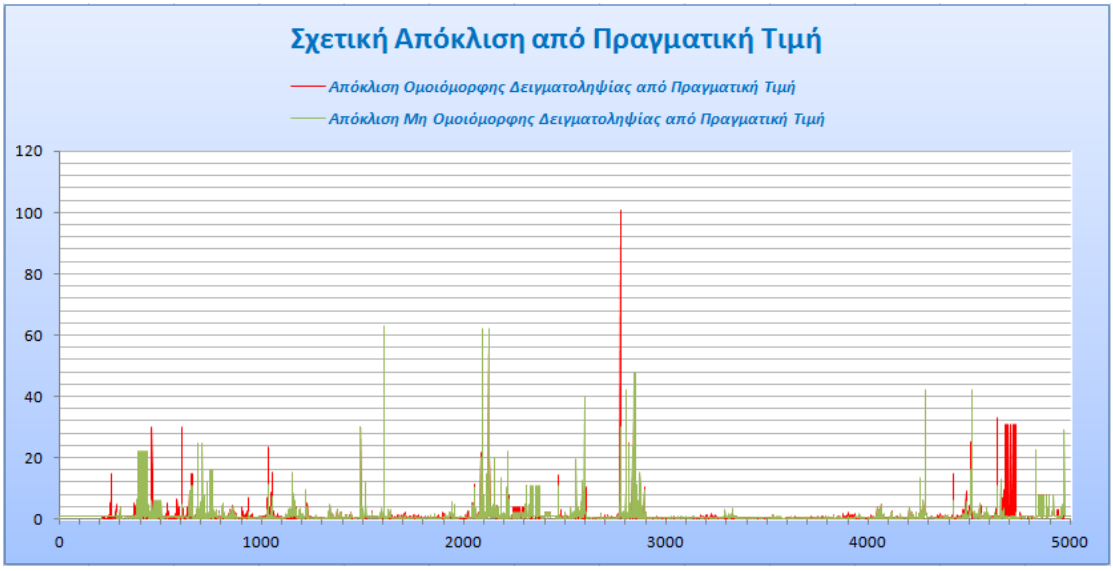
Η εφαρμογή των ανωτέρω τιμών πιθανοτήτων στον ίδιο αλγόριθμο επιδημικής μετάδοσης της πληροφορίας οδηγεί στα αποτελέσματα μόλυνσης που παρατίθενται στη συνέχεια.

Ειδικότερα, στο Σχήμα 11 αποτυπώνεται η αντίληψη που σχηματίζει ο τυχαίος κόμβος – αναμεταδότης για το ρεύμα δεδομένων που παρέχεται από τους κόμβους ομοιόμορφης (κόκκινη γραμμή) και μη ομοιόμορφης (πράσινη γραμμή) δειγματοληψίας στο δίκτυο. Με πρώτη ματιά, ο κόμβος 12 παρουσιάζει συντομότερο χρονικό σημείο αρχικής επιμόλυνσης για το σχήμα ομοιόμορφης δειγματοληψίας ενώ δίδεται και η αίσθηση συχνότερης λήψης νέων τιμών σε σχέση με τον αλγόριθμο μη ομοιόμορφης δειγματοληψίας. Από την άλλη, ο κόμβος 12 του σχήματος μη ομοιόμορφης δειγματοληψίας φαίνεται να είναι σε θέση να παρακολουθεί καλύτερα τις ακραίες τιμές των πρωτογενών δεδομένων, εξαιτίας της βελτιωμένης προσέγγισης του σήματος από τον αντίστοιχο κόμβο – πηγή. Παράλληλα, εντοπίζονται σημεία μακροσκελούς στασιμότητας εκατέρωθεν (π.χ. 2.900 έως 3.200 χρονικά σημεία), οι αιτίες των οποίων, όμως, ενδεχομένως να πρέπει να αναζητηθούν στην τροφοδοτική δυνατότητα της γειτονιάς (ακτίνα επικοινωνίας, πυκνότητα κόμβων κ.λπ.), καθώς ακόμα και στην περίπτωση της ομοιόμορφης δειγματοληψίας, όπου υπάρχει τακτικότερη ροή πληροφορίας, δεν επακολουθεί ενδιάμεση ενημέρωση των δεδομένων του κόμβου 12.



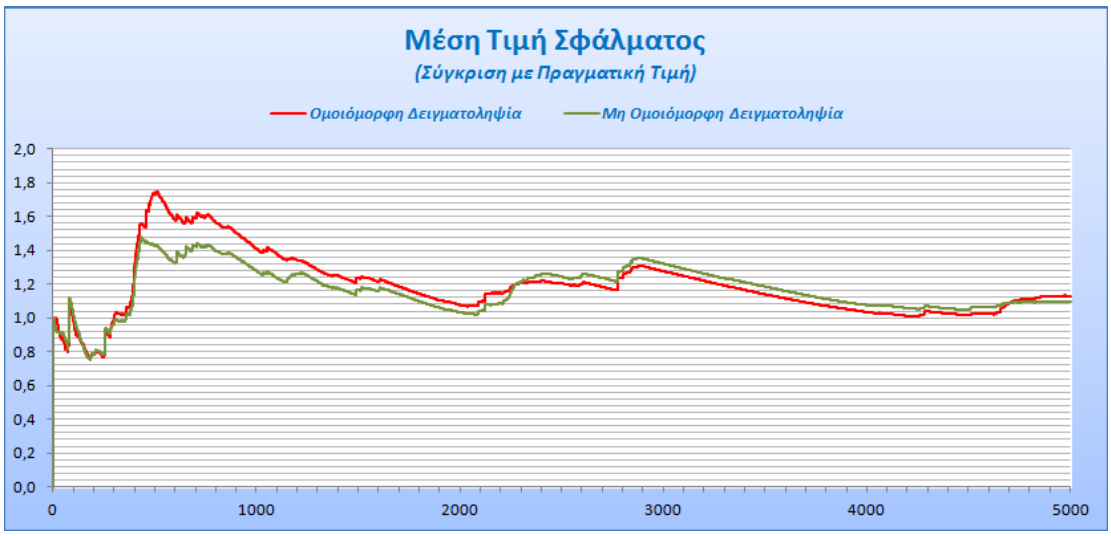
Σχήμα 11: Σήμα που αντιλαμβάνεται ο τυχαίος κόμβος – αναμεταδότης ως αποτέλεσμα εφαρμογής του ίδιου επιδημικού αλγόριθμου για πηγή ομοιόμορφης (κόκκινη γραμμή) και μη ομοιόμορφης (πράσινη γραμμή) δειγματοληψίας

Κατόπιν των ανωτέρω, σχηματίζεται συνολικά η εντύπωση ότι τα δύο γραφήματα παρακολουθούν τη γενική τάση του αρχικού σήματος με χονδρικά παρόμοια πιστότητα. Προς επαλήθευση αυτού δημιουργήθηκε το Σχήμα 12, στο οποίο αποτυπώνεται η σχετική απόκλιση της φέρουσας στον κόμβο 12 τιμής από την πραγματική τιμή του ρεύματος δεδομένων την ίδια χρονική στιγμή. Όπως διαπιστώθηκε και στο αντίστοιχο γράφημα του κόμβου – πηγή, τα δύο σχήματα εμφανίζουν εν γένει παρεμφερείς τιμές σχετικής απόκλισης, με τις εξαιρέσεις να είναι συχνότερες στη μέθοδο μη ομοιόμορφης δειγματοληψίας αλλά εντονότερες στην ομοιόμορφη δειγματοληψία.



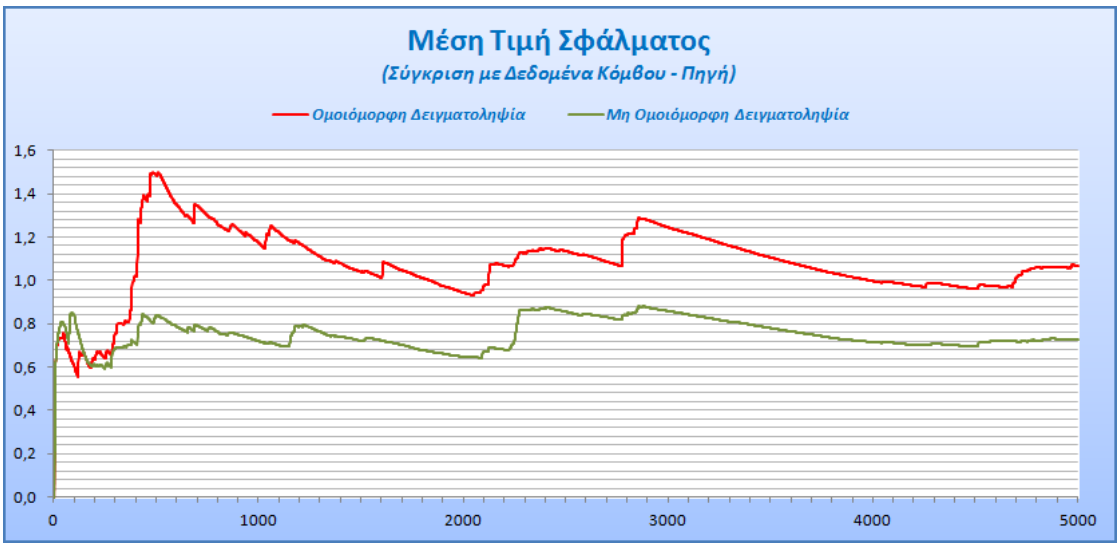
Σχήμα 12: Σχετική απόκλιση των δεδομένων του τυχαίου κόμβου – αναμεταδότη από την πραγματική τιμή, μετά την εφαρμογή του ίδιου επιδημικού αλγόριθμου

Θεωρώντας ότι αυτή η συμπεριφορά του τυχαίου κόμβου – αναμεταδότη αντιπροσωπεύει τον τρόπο αλληλεπίδρασης όλων των κόμβων καθ' όλη τη διάρκεια της προσομοίωσης, δεν προξενεί εντύπωση ότι το σωρευτικό σφάλμα του δικτύου φέρει τη μορφή του ακόλουθου Σχήματος.



Σχήμα 13: Μέση τιμή σφάλματος – σύγκριση με το πρωτογενές σήμα

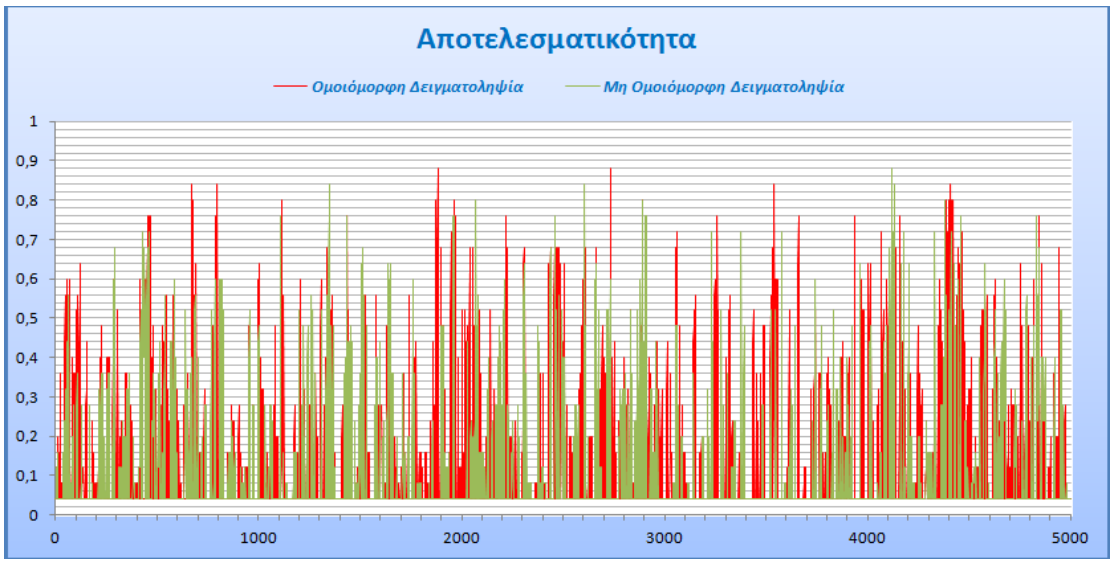
Επαληθεύεται για μια ακόμη φορά ότι οι κόμβοι των δύο σχημάτων επιτυγχάνουν παρόμοια επίπεδα σφάλματος, με σημαντικά περιορισμένο πλήθος δειγμάτων από πλευράς μη ομοιόμορφης δειγματοληψίας (βλ. Σχήμα 8). Ωστόσο, η απόκλιση γίνεται εντονότερη εάν στη θέση των τιμών του αρχέτυπου σήματος χρησιμοποιηθεί η τρέχουσα πληροφορία του κόμβου – πηγή ως μέτρο σύγκρισης για τον υπολογισμό του υπεισελθόντος σφάλματος (Σχήμα 14).



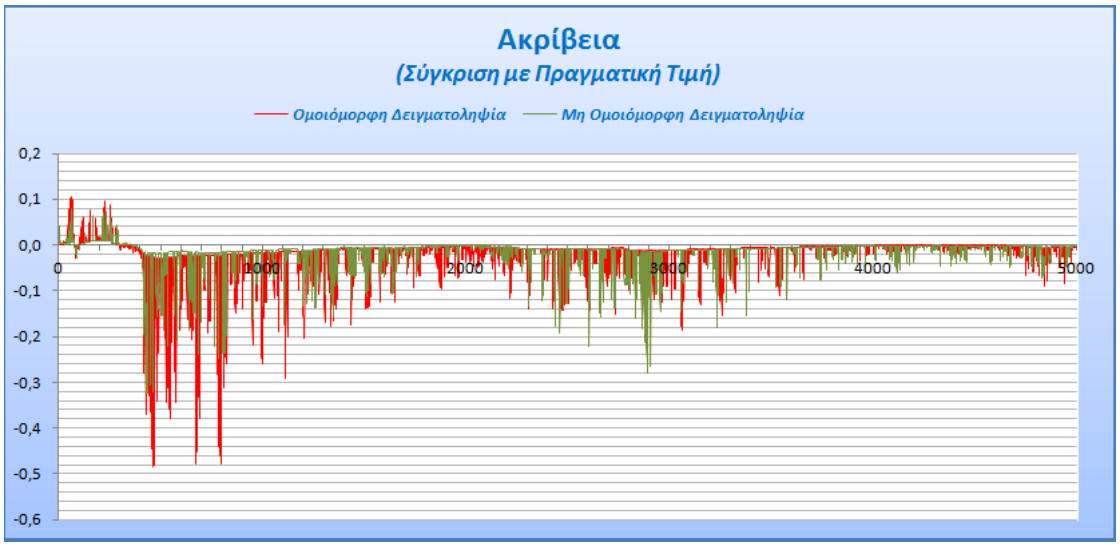
Σχήμα 14: Μέση τιμή σφάλματος – σύγκριση με το σήμα που αντιλαμβάνεται ο κόμβος – πηγή

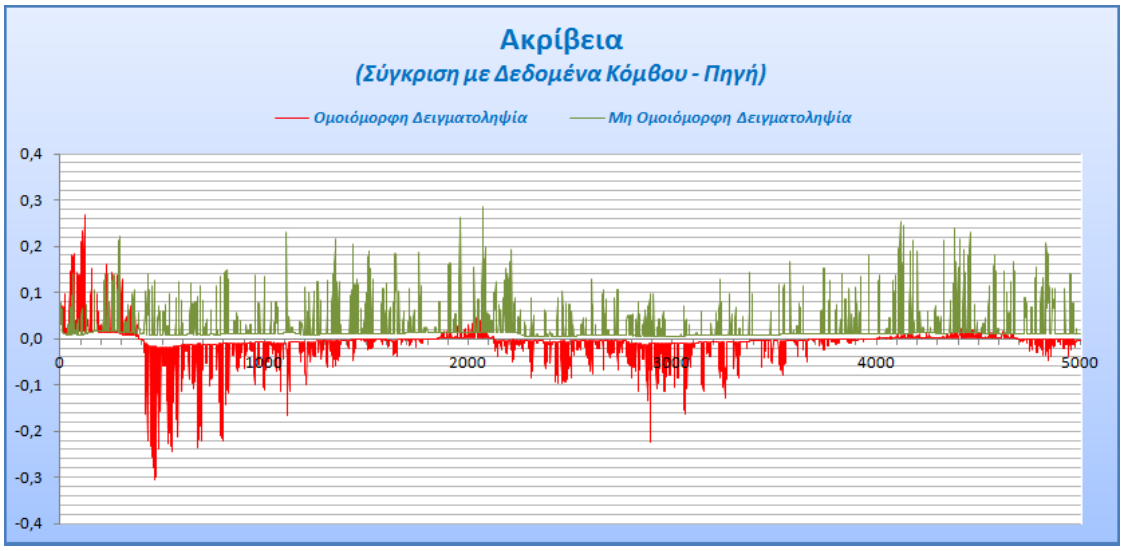
Όπως αποτυπώνεται και στην καμπύλη της μέσης απόκλισης από την τιμή του κόμβου – πηγή, η πληροφορία που επικρατεί στο δίκτυο του σχήματος μη ομοιόμορφης δειγματοληψίας είναι αισθητά πιο συγχρονισμένη στην αντίστοιχη τιμή του κόμβου – πηγή, καθώς η ανάλογη τιμή απόκλισης του αλγόριθμου ομοιόμορφης δειγματοληψίας διατηρείται σε συγκριτικά μεγαλύτερα επίπεδα. Λαμβάνοντας όμως υπόψη ότι οι κόμβοι – πηγή των δύο σχημάτων αποκτούν πρακτικά την ίδια θεώρηση για το αρχικό σήμα (Σχήμα 5) την οποία και τροφοδοτούν στο δίκτυο με την ίδια πιθανότητα (Σχήμα 10) ενώ αντιστοίχως οι κόμβοι – αναμεταδότες παρουσιάζουν παρεμφερή επίπεδα απόκλισης από το αρχικό σήμα (Σχήμα 12), η υπεροχή του Σχήματος 14 από τη σκοπιά της ποιότητας δεν υποστηρίζεται από τα μέχρι στιγμής αποτελέσματα της σύγκρισης. Για να μπορέσει να ερμηνευθεί το προβάδισμα που υπονοείται από το ανωτέρω σχήμα, θα πρέπει να αναλογισθεί κανείς το εκτενέστερο χρονικό περιθώριο που έχουν οι κόμβοι – αναμεταδότες του δικτύου μη ομοιόμορφης δειγματοληψίας να συντονισθούν με τα δεδομένα του κόμβου – πηγή λόγω σποραδικότερης δειγματοληψίας και ταυτόχρονα να συνυπολογισθεί η απαλοιφή τού δειγματοληπτικού σφάλματος, η οποία επηρεάζει σε διαφορετικό βαθμό τα περιεχόμενα των κόμβων – πηγή των δύο μοντέλων.

Ανάλογη εικόνα αναμένεται να εμφανίσει και η παράμετρος που συμβολίζει την ακρίβεια των δεδομένων των δύο σχημάτων. Καθόσον η καμπύλη της αποτελεσματικότητας (Σχήμα 15), που συμμετέχει μαζί με την παράμετρο $e(t)$ στον υπολογισμό της ακρίβειας, λαμβάνει παρεμφερή μορφή για τα δύο σχήματα, ο βαθμός συντονισμού των δεδομένων των κόμβων του δικτύου με το πρωτογενές σήμα και με τα δεδομένα του κόμβου – πηγή είναι φυσικό να εκδηλώσει για τους δύο αλγόριθμους το συσχετισμό του Σχήματος 16 (α και β, αντιστοίχως).



Σχήμα 15: Αποτελεσματικότητα των σχημάτων ομοιόμορφης (κόκκινη γραμμή) και μη ομοιόμορφης (πράσινη γραμμή) δειγματοληψίας συναρτήσει του χρόνου

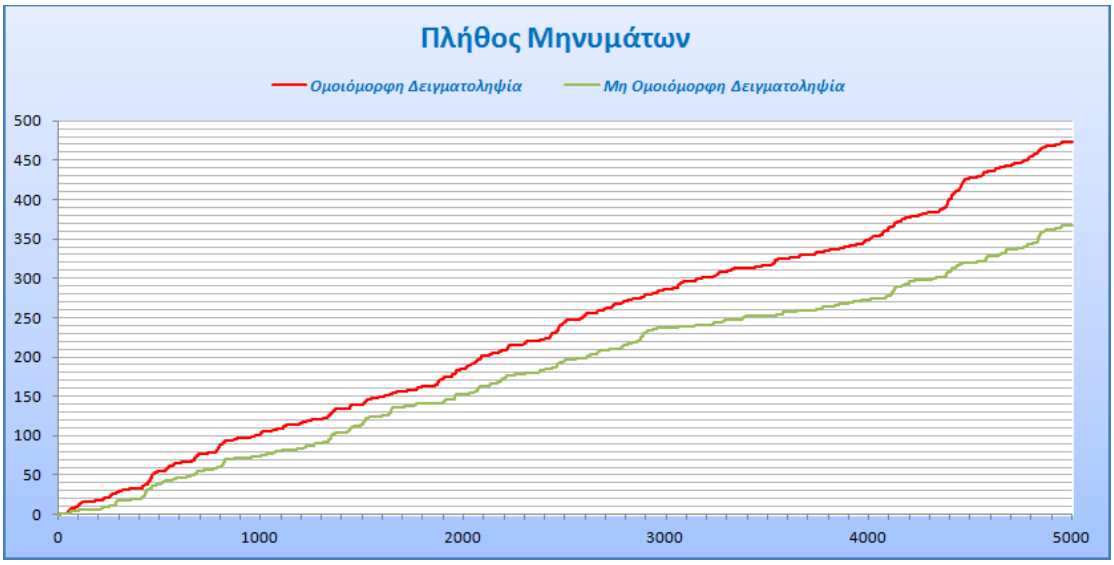




Σχήμα 16: Ακρίβεια των σχημάτων ομοιόμορφης (κόκκινη γραμμή) και μη ομοιόμορφης (πράσινη γραμμή) δειγματοληψίας – σύγκριση με το πρωτογενές σήμα (πάνω) και με το σήμα που αντιλαμβάνεται ο κόμβος – πηγή (κάτω)

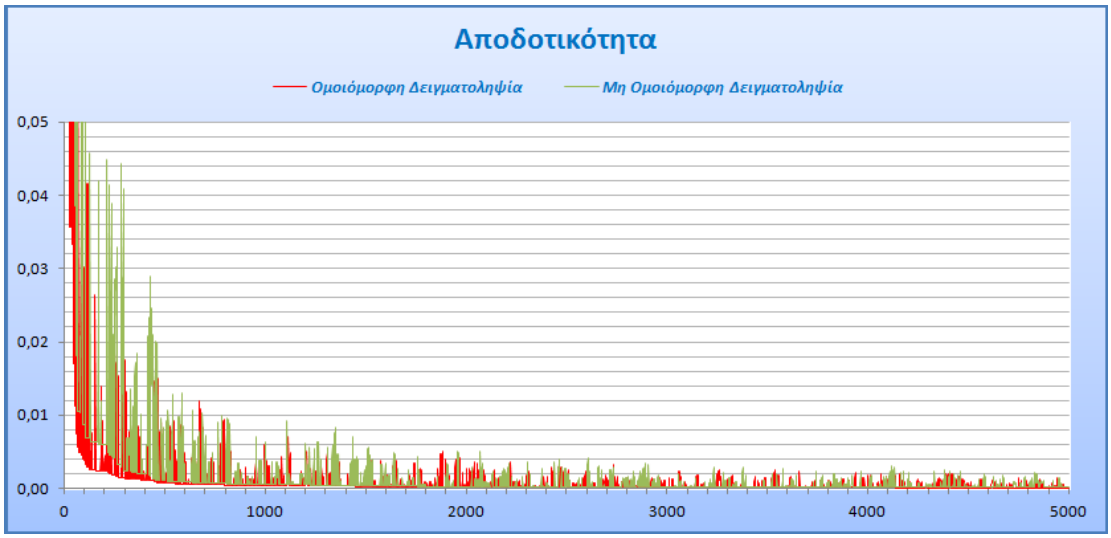
Φαίνεται λοιπόν ότι, κατ' αναλογία με τη μετρική μέσου σφάλματος, η μεταβλητή που εκφράζει την ακρίβεια των μοντέλων λαμβάνει παρόμοιες τιμές όταν τα δεδομένα των κόμβων του δικτύου αντιπαραβάλλονται με το πρωτογενές σήμα, ενώ η διαφοροποίηση γίνεται εντονότερη υπέρ του σχήματος μη ομοιόμορφης δειγματοληψίας, όταν η αποτίμηση πραγματοποιείται σε σχέση με τη φέρουσα στον κόμβο – πηγή πληροφορία.

Η υπεροχή αυτή, σε συνδυασμό με το σημαντικό περιορισμό του πλήθους των διακινούμενων μηνυμάτων (Σχήμα 17) οδηγεί στο συμπέρασμα ότι η επακόλουθη μείωση της απαιτούμενης ενεργειακής δαπάνης δεν επέφερε αρνητική επίδραση στον τρόπο λειτουργίας του δικτύου.



Σχήμα 17: Σωρευτικό πλήθος διακινηθέντων μηνυμάτων συναρτήσει του χρόνου

Βέβαια, η στάθμη αποδοτικότητας και των δύο σχημάτων παραμένει στα ίδια, χαμηλά ύψη (Σχήμα 18), ενδεχομένως λόγω των αυξημένων περιθωρίων ανάρρωσης που επιτρέπει στους κόμβους – αναμεταδότες η σπανιότερη δειγματοληψία.



Σχήμα 18: Αποδοτικότητα των σχημάτων ομοιόμορφης (κόκκινη γραμμή) και μη ομοιόμορφης (πράσινη γραμμή) δειγματοληψίας συναρτήσει του χρόνου

5. ΣΥΜΠΕΡΑΣΜΑΤΑ

Όταν το επίπεδο κρισιμότητας της εφαρμογής ενός κινητού ad–hoc δικτύου επιτρέπει το περιθώριο ατελούς κάλυψης, η χρήση μη ντετερμινιστικών αλγόριθμων μετάδοσης πληροφορίας αποτελεί μονόδρομο για την επίτευξη της ιδανικής ισορροπίας μεταξύ αποτελεσματικότητας και αποδοτικότητας στο δίκτυο. Κύριος αντιπρόσωπος αυτής της οικογένειας αλγορίθμων είναι η κλάση των επιδημικών αλγόριθμων, με εξέχουσα την παραλλαγή aSIS, η οποία διακρίνεται για τη βελτιστοποίηση των παραμέτρων μόλυνσης που επιτυγχάνει με βάση τα χαρακτηριστικά του ρεύματος δεδομένων που αντιλαμβάνεται ο κόμβος – πηγή.

Εξ αφορμής της σπουδαιότητας που αποκτά, σύμφωνα με τα ανωτέρω, η αντίληψη που σχηματίζει ο κόμβος – πηγή για το περιβάλλον, παρουσιάσθηκε μια δεδομενο-κεντρική παραλλαγή του αλγόριθμου επιδημικής διάχυσης aSIS, η οποία εκμεταλλεύεται τη φύση του πρωτογενούς σήματος, προκειμένου να προσαρμόσει αυτή τη φορά το ρυθμό δειγματοληψίας του αισθητήρα στις στατιστικές ιδιότητες της παθογένειας. Με αυτό τον τρόπο επιχειρείται μια πιο ποιοτική προσέγγιση των δεδομένων που θα αποτελέσουν πρώτη ύλη για την εφαρμογή του κλασικού aSIS σχήματος.

Η αξιολόγηση της αποτελεσματικότητας της προτεινόμενης μεθόδου στηρίχθηκε στη σύγκριση των επιδόσεων του ίδιου αλγόριθμου επιδημικής μετάδοσης (aSIS) για δύο διαφορετικές πηγές δεδομένων: μια πηγή που εμφανίζει σταθερό ρυθμό δειγματοληψίας και μία που προσαρμόζει την περίοδο δειγματοληψίας στη μεταβλητότητα του πρωτογενούς ρεύματος δεδομένων. Παρατηρείται, λοιπόν, ότι με περιορισμένη ενεργοποίηση του δειγματολήπτη – αισθητήρα και κατ' επέκταση λιγότερη διακινούμενη πληροφορία, ο αλγόριθμος μεταβλητής δειγματοληψίας εξασφαλίζει προσεγγιστικά την ίδια ποιότητα δεδομένων στο δίκτυο.

Βέβαια, τα παραπάνω αποτελέσματα θα πρέπει να εκτιμηθούν και πειραματικά, προκειμένου να επαληθευτεί το όφελος που επιτυγχάνεται από την εμφαινόμενη εξοικονόμηση ενέργειας. Καθώς η συχνή αλλαγή κατάστασης του αισθητήρα μπορεί να καταλήξει σε πρόσθετη σπατάλη ενεργοποίησης/ απενεργοποίησης, η οποία μάλιστα θα μπορούσε υπό συνθήκες να είναι συγκρίσιμη με την ωφέλιμη ενέργεια που έχει δαπανηθεί για την καθαρή μετάδοση της πληροφορίας στο δίκτυο, είναι χρήσιμο να καθορισθεί η επίδραση που έχει στο ενεργειακό ισοζύγιο η μικρή ή μεγαλύτερη παύση του αισθητήρα.

Περαιτέρω προοπτικές βελτίωσης θα πρέπει να αναζητηθούν στην προσαρμογή λοιπών συνιστωσών που επιδρούν στη μεταδοτικότητα των επιδημικών αλγόριθμων, όπως π.χ. στη ρύθμιση της ακτίνας επικοινωνίας των κόμβων. Και αυτό γιατί φαινομενικά μια μεγάλη ακτίνα επικοινωνίας θα συνεπικουρούσε στην ολοκληρωτική μετάδοση της πληροφορίας, ενδέχεται όμως να προκαλέσει μεγαλύτερη επιβάρυνση με δαπάνη ενέργειας για επαναποστολή πακέτων λόγω της αυξημένης πιθανότητας συγκρούσεων που συνεπάγεται η υπερβάλλουσα επέκταση της γειτονιάς (greedy forwarding). Σε κάθε περίπτωση, πάντως, κατά τη διερεύνηση των εναλλακτικών δεν πρέπει να υπονοείται γνώση της τοπολογίας του δικτύου (π.χ. καθορισμός συνιστωσών βάσει του μεγέθους της γειτονιάς ενός κόμβου), καθώς κάτι τέτοιο θα σήμαινε πρόσθετη απασχόληση πόρων για τη δημιουργία και συντήρηση δεδομένων, τα οποία ούτως ή άλλως μεταβάλλονται συχνά λόγω της ρευστής φύσης του δικτύου.

Τέλος, σφάλμα θα ήταν η παράλειψη της εξέτασης των κοινωνικών προεκτάσεων που συνοδεύουν ολόκληρη την οικογένεια συνεργατικών αλγόριθμων. Διότι μπορεί μεν σε ερευνητικό επίπεδο οι απαιτήσεις για τον τρόπο αναπαράστασης και διακίνησης της πληροφορίας να περιορίζονται στην υιοθέτηση κοινού πρωτοκόλλου επικοινωνίας, για τις εμπορικές εφαρμογές ωστόσο είναι αναγκαία η υιοθέτηση πιο περίπλοκων κανόνων

προσβασιμότητας στην πληροφορία, καθώς εγείρονται θέματα εμπιστευτικότητας και ανεξέλεγκτης δυνατότητας τροποποίησης των δεδομένων.

ΠΙΝΑΚΑΣ ΟΡΟΛΟΓΙΑΣ

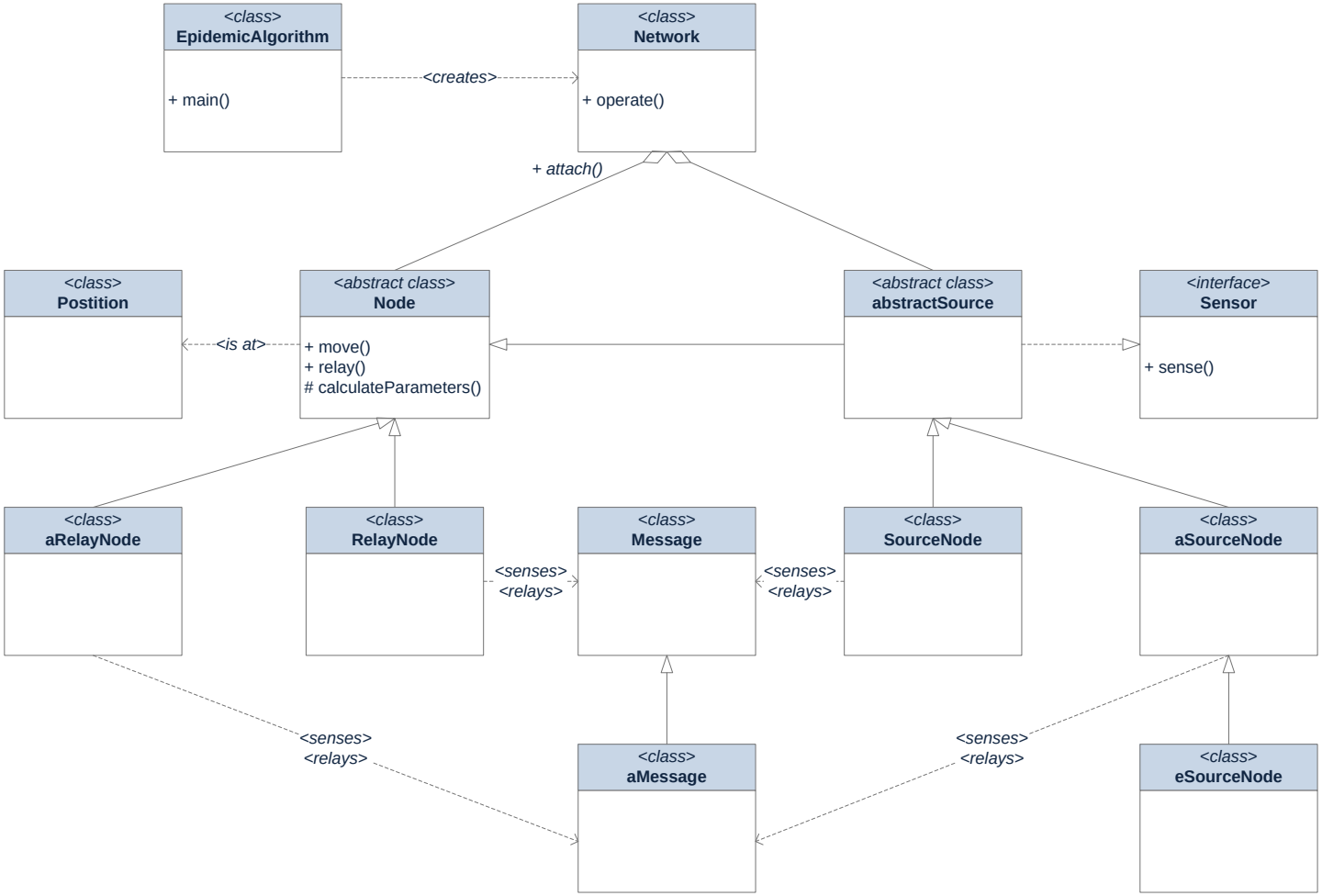
Ξενόγλωσσος όρος	**Ελληνικός Όρος**
Ad-Hoc Networks	Αυτό-Οργανούμενα Δίκτυα
Client - Server	Πελάτης - Εξυπηρέτης
Flooding Algorithm	Αλγόριθμος Πλημμύρας
Peer-to-Peer Networks	Ομότιμα Δίκτυα
Epidemical Spreading	Επιδημική Μετάδοση
Mobile Computing	Κινητός Υπολογισμός
Stateful Session	Ενακόλουθη Σύνοδος
Mobile Ad-Hoc Networks	Αυτόνομα Δίκτυα Ειδικού Σκοπού
Single Point of Failure	Κρίσιμο Στοιχείο Αστοχίας
Overhead	Πλεονάζον Κόστος
Collaborative Node	Συνεργατικός Κόμβος
Epidemic Algorithm	Επιδημικός Αλγόριθμος
Rumor Spreading	Διάδοση Φήμης
Data Stream	Ρεύμα Δεδομένων
Neighborhood	Γειτονιά
Susceptible Node	Υγιής Κόμβος
Infected Node	Μολυσμένος Κόμβος
Removed Node	Άνοσος Κόμβος
Stale	Απαρχαιωμένη
Source Node	Κόμβος – Πηγή
Relay Node	Κόμβος – Αναμεταδότης
Forwarding Probability	Πιθανότητα Μετάδοσης
Recovery Rate	Ρυθμός Ίασης
Removal Rate	Ρυθμός Ανοσοποίησης
Transmutation	Μετάλλαξη
Aggravation	Επιδείνωση
Inference	Συμπερασμός
Data Fusion	Συγχώνευση Δεδομένων
Collaborative Context Awareness	Συνεργατική Επίγνωση Πληροφορίας Πλαισίου
Partial Cure	Μερική Ανάρρωση
Hops	Άλματα
Expiration Time	Χρόνος Εκπνοής Ισχύος
Generation Time	Χρόνος Δημιουργίας
Continuous-Like Signal	Συνεχο-φανές Σήμα
Greedy Forwarding	Άπληστη Προώθηση Πληροφορίας

ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ – ΑΚΡΩΝΥΜΙΑ

MANET	Mobile Ad-hoc NETworks
SIR	Susceptible Infected Removed
SIS	Susceptible Infected Susceptible
SI	Susceptible Infected
SaIS	Susceptible aggravation Infected Susceptible
aSIS	Adaptive SIS
RWP	Random Waypoint

ΠΑΡΑΡΤΗΜΑ Ι

Στο Παράρτημα Ι παρατίθεται σε γλώσσα Java το πρόγραμμα προσομοίωσης που χρησιμοποιήθηκε για την παραγωγή των αποτελεσμάτων του παρόντος δοκιμίου. Η ανάπτυξη του λογισμικού πραγματοποιήθηκε στην πλατφόρμα eclipse MARS.1 [12] βάσει του μοντέλου που απεικονίζεται στο κάτωθι απλοποιημένο διάγραμμα κλάσεων.



Σχήμα 19: Απλοποιημένο διάγραμμα κλάσεων του προγράμματος προσομοίωσης

Αρχικά ο χρήστης καλείται να ορίσει το μέγεθος του δικτύου και να αποφασίσει για τον επιθυμητό αλγόριθμο προσομοίωσης. Η επιλογή αυτή γίνεται μέσα από ένα σύνολο υλοποιημένων σχημάτων επιδημίας, το οποίο, για λόγους πληρότητας και πέραν της προτεινόμενης παραλλαγής του αλγόριθμου aSIS και του αντίστοιχου κλασικού σχήματος, περιλαμβάνει και τους λοιπούς αλγόριθμους επιδημίας SIR, SIS, SI και Flooding. Εν συνεχεία και αναλόγως του επιλεγέντος σχήματος, προσδιορίζονται οι παράμετροι του αλγόριθμου (α, β ή/και δ) και ξεκινά η λειτουργία του δικτύου.

Πιο συγκεκριμένα, κάθε χρονική στιγμή έκαστος κόμβος μετακινείται εντός του πλέγματος του δικτύου. Στη νέα του θέση ανταλλάσσει πληροφορίες με τους γειτονικούς του κόμβους, στο πλαίσιο εφαρμογής του επιδημικού σχήματος. Παράλληλα, σε προκαθορισμένα χρονικά σημεία, ενεργοποιείται ο αισθητήρας του κόμβου – πηγή εισάγοντας νέα τιμή στο δίκτυο, η οποία με τη σειρά της ενδέχεται να επιμολύνει μέρος του πληθυσμού.

Τα δεδομένα δειγματοληψίας του κόμβου – πηγή προκύπτουν από τις γραμμές του αρχείου ./sample.txt, ενώ οι συντεταγμένες κίνησης φυλάσσονται στο αρχείο ./trace.txt, που φέρει γραμμογράφηση όπως αυτή προσδιορίζεται από τα plaintext αρχεία του προσομοιωτή κίνησης Mobisim (βλ. Παράρτημα ΙΙ).

Το αποτέλεσμα εκτέλεσης του προγράμματος καταγράφεται στο αρχείο ./output.txt, με εγγραφές της μορφής <Κόμβος, Τρέχουσα_Τιμή, Χρονικό_Σημείο_Τρέχουσας_Τιμής, Τιμή_Μετάδοσης, Χρονικό_Σημείο _Τιμής_Μετάδοσης, Πλήθος_Εξερχόμενων_Μηνυμάτων, Κατάσταση_Κόμβου, β>.

Πρόγραμμα

```
import java.io.*;
public class EpidemicAlgorithm {
	/**
	 * Dynamic Tuning of Epidemic Algorithm's Variables
	 */
	public static void main(String[] args) {
		int noNodes=0;
		double remRate=0;
		double recRate=0;
		double b=1;
		Network N;

		BufferedReader reader=new BufferedReader(new InputStreamReader(System.in));
		String commandPrompt;
		boolean validInput;
		int i=1, algo=0, j=1;
```

```
do{
    try{
        switch(j){
        case 1:
            do{
                validInput=true;
                System.out.print("Please enter relay population size (or press '@' to exit program): ");

                if((commandPrompt=reader.readLine()).equals("@"))
                    return;

                if((noNodes=Integer.parseInt(commandPrompt))<0){
                    System.out.println("Given network size is not acceptable.");
                    validInput=false;
                }
            }
            while(!validInput);
            j++;
        case 2:
            // Decide on execution mode
            System.out.println();
            System.out.println("List of implemented algorithms: ");
            System.out.println("1. SIR");
            System.out.println("2. SIS");
            System.out.println("3. SI");
            System.out.println("4. Flooding");
            System.out.println("5. aSIS");
            System.out.println("6. enhanced aSIS");
            System.out.println();
            do{
                validInput=true;
                System.out.print("Please choose an algorithm (or press '@' to exit program): ");

                if((commandPrompt=reader.readLine()).equals("@"))
```

```
                        return;

                    if((algo=Integer.parseInt(commandPrompt))<1 || algo>6){
                        System.out.println("Should enter any of the values given above!");
                        validInput=false;
                    }
                }
                while(!validInput);
                j++;
            case 3:
                do{
                    validInput=true;
                    switch(algo){
                        case 1:
                            System.out.print("Please enter removal rate (or press '@' to exit program): ");
                            if((commandPrompt=reader.readLine()).equals("@"))
                                return;
                            remRate=Double.parseDouble(commandPrompt);
                            if(remRate>1 || remRate<=0){
                                System.out.println("Removal rate values should be between 0 and 1.");
                                validInput=false;
                                break;
                            }
                        case 2:
                            System.out.print("Please enter recovery rate (or press '@' to exit program): ");
                            if((commandPrompt=reader.readLine()).equals("@"))
                                return;
                            recRate=Double.parseDouble(commandPrompt);
                            if(recRate>1 || recRate<=0){
                                System.out.println("Recovery rate values should be between 0 and 1.");
                                validInput=false;
                                break;
                            }
                            if(recRate+remRate>1){
                                System.out.println("Nodes can recover, immune or remain infected.");
```

```
                                        validInput=false;
                                        break;
                                    }
                                case 3:
                                case 5:
                                case 6:
                                    System.out.print("Please enter forwarding probability (or press '@' to exit program): ");
                                    if((commandPrompt=reader.readLine()).equals("@"))
                                        return;
                                    b=Double.parseDouble(commandPrompt);
                                    if(b>1 || b<=0){
                                        System.out.println("Forwarding probability values should be between 0 and 1.");
                                        validInput=false;
                                        break;
                                    }
                                case 4:
                                    break;
                                default:
                                    System.err.println("Should never occur!");
                                    return;
                                }
                            }
                            while(!validInput);
                            j++;
                            break;
                        default:
                            System.err.println("Should never occur!");
                            return;
                        }
                        reader.close();   //Close stream. No further input is required
                    }
                    catch(NumberFormatException nfe){
                        validInput=false;
                        System.out.println("Invalid number format. Please try your best!");
                    }
```

```
                catch(IOException e){
                        e.printStackTrace();
                        return;
                }
        }
        while(j<4);

        switch(algo){
        case 1:
        case 2:
        case 3:
        case 4:
                N= new Network(new SourceNode(0, b));
                for(i=1; i<=noNodes; i++)
                        N.attach(new RelayNode(i, b, remRate, recRate));
                break;
        case 5:
                N=new Network(new aSourceNode(0, b));
                for(i=1; i<=noNodes; i++)
                        N.attach(new aRelayNode(i, b));
                break;
        case 6:
                N=new Network(new eSourceNode(0, b));
                for(i=1; i<=noNodes; i++)
                        N.attach(new aRelayNode(i, b));
                break;
        default:
                System.err.println("Should never occur!");
                return;
        }
        N.operate();
        return;

    }
}
```

```
import java.util.ArrayList;
import java.util.Iterator;

public class Network {
	protected abstractSource<? extends Message> source;
	protected ArrayList<Node<? extends Message>> network;

	Network(abstractSource<? extends Message> s){
		source=s;
		network=new ArrayList<Node<? extends Message>>();
	}

	public boolean attach(Node<? extends Message> n){
		network.add(n);
		return true;
	}

	public void operate(){
		try{
			while(source.sense()){
				source.move();
				for(Node<?> relay:network)
					relay.move();

				source.relay(network);
				for(Node<?> relay:network)
					relay.relay(network);

				source.update();
				for(Iterator<Node<?>> relay=network.iterator(); relay.hasNext(); )
					if(relay.next().update()==Node.State.REMOVED)
						relay.remove();
			}
```

```
            }
            catch(Exception e){
                e.printStackTrace();
            }
            return;
        }
}
```

```
import java.io.*;
import java.util.ArrayList;
import java.util.Random;

public abstract class Node<T extends Message> {

    protected enum State {SUSCEPTIBLE, INFECTED, REMOVED}

    protected static final double R_MIN=20;         //Minimum infection radius
    protected static final double R_MAX=180;        //Maximum infection radius

    protected int name;   //Node's name
    protected double r;   //Infection radius
    protected double b;   //Infection rate
    protected double a;   //Removal rate
    protected double d;   //Recovery rate

    protected State e;

    protected Random decide;

    protected Position cur_pos;         //Node's position
    protected static StringBuffer line; //Buffer to host lines of coordinates' file
    protected static BufferedReader trace;      //Moving coordinates
    protected static BufferedWriter output;     //Results' file
```

```
protected T cur_msg;
protected T pre_msg;

protected boolean sensed;
protected boolean relay;
protected int noMsg;

Node(int i, double inf, double rem, double rec){  //constructor
        name=i;

        b=inf;
        a=rem;
        d=rec;

        r=(R_MAX+R_MIN)/2;

        noMsg=0;

        sensed=false;
        relay=false;

        cur_pos= new Position();
        decide=new Random(i);
}

public boolean move() throws Exception{
// Coordinates' file format:
// <time> <nodeId> <x> <y> <velocity> <movement_angle>
        try{
                line.replace(0, 30, trace.readLine());     //Get a line from trace file
                int start=0, stop;
                double x, y;

                for(int i=0; i<2; i++){
```

```
                stop= line.indexOf("\t", start);
                start= stop+1;
            }

            stop= line.indexOf("\t", start);       //Reached x coordinate
            x= Double.parseDouble(line.substring(start, stop));

            start= stop+1;
            stop= line.indexOf("\t", start);       //Reached y coordinate
            y= Double.parseDouble(line.substring(start, stop));

            cur_pos.move(x, y);    //Update node's coordinates

            return true;
        }
        catch(NullPointerException e){    //EOF has been reached
            return false;      //Epidemic algorithm on static nodes
        }
        catch(Exception e){
            e.printStackTrace();
            throw(e);   //Something went wrong
        }
    }

    public void relay(ArrayList<Node<? extends Message>> network){
        if(relay){
            boolean decision=(decide.nextDouble()<=b?true:false);
            if(decision)
                for(Node<? extends Message> node:network){
                    if(node.getName()==name || !inScope(node))
                        continue;
                    noMsg++;
                    node.sense(pre_msg);
                }
            relay=false;
```

```
            }
            return;
        }

        public abstract void sense(Message in);

        public final State update(){
            updateStatus();
            try{
output.write(name+","+cur_msg.getValue()+","+cur_msg.getValueTime()+","+pre_msg.getValue()+","+pre_msg.getValueTime()+","+noMsg+","+e+","+b);
                output.newLine();
                noMsg=0;
            }
            catch(Exception e){
                e.printStackTrace();
            }
            if(sensed){
                relay=true;
                sensed=false;
                calculateParameters();
                updateMessage();
            }
            return e;
        }

        protected abstract void updateMessage();

        public Position getPosition(){
            return cur_pos;
        }

        public int getName(){
            return name;
        }
```

```
	public boolean inScope(Node<?> n){
		return cur_pos.distance(n.getPosition())<=r?true:false;
	}

	protected void updateStatus(){
		if(e==State.INFECTED){
			double random=decide.nextDouble();
			if(random<d){
				e= State.SUSCEPTIBLE;	//Recover
				sensed=false;
			}
			else if(d<=random && random<d+a)
				e= State.REMOVED;		//Remove
		}
		return;
	}

	protected void calculateParameters(){
		return;
	}
}
```

```
public interface Sensor {
	public boolean sense();
}
```

```
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;
```

```
public abstract class abstractSource<T extends Message> extends Node<T> implements Sensor {

	private BufferedReader sensor;
	private int t;			//Network's clock
	protected int leaveout;		//Sampling period

	private static final int SAMPLING_RATE_MID=5;		//Mid sampling period

	abstractSource(int i, double inf) {		//constructor
		super(i, inf, 0, 0);

		e= State.INFECTED;		//Source nodes are always infected

		line= new StringBuffer(20);
		t=0;
		leaveout=SAMPLING_RATE_MID;

		try{
			sensor= new BufferedReader(new FileReader("sample.txt"));		//Operating sensor
			trace= new BufferedReader(new FileReader("trace.txt"));		//Get ready to move
			output= new BufferedWriter(new FileWriter("output.txt"));		//Noting results down

			output.write("Node Name,Current Value,Current Value Time,Relay Value,Relay Value Time,#Sent Messages,Node Status,b Value");
			output.newLine();
		}
		catch(IOException e){
			e.printStackTrace();
		}
	}

	public boolean sense(){
		t++;
		try{
			if(t==(cur_msg.getValueTime()+leaveout)){	//If it is time to sense
				cur_msg.setValue(Double.parseDouble(sensor.readLine()));
```

```
                        cur_msg.setValueTime(t);
                        sensed=true;
                }
                else
                        sensor.readLine();      //Skip value of file
                return true;
        }
        catch(NullPointerException e){  //EOF has been reached
                try{
                        sensor.close();
                        trace.close();
                        output.close();
                }
                catch(NullPointerException innerE){
                        innerE.printStackTrace();
                }
                catch(IOException innerE){
                        innerE.printStackTrace();
                }
        }
        catch(IOException e){
                e.printStackTrace();
        }
        return false;
    }

    public void sense(Message in){
        return;
    }
}
```

```
public class SourceNode extends abstractSource<Message> {
```

```
        SourceNode(int i, double inf) {    //constructor
                super(i, inf);

                cur_msg= new Message();
                pre_msg= new Message();
        }

        protected void updateMessage(){
                pre_msg.update(cur_msg);
        }
}
```

```
public class aSourceNode extends abstractSource<aMessage> {

        double b_rel;

        aSourceNode(int i, double inf) {  //constructor
                super(i, inf);

                cur_msg= new aMessage();
                pre_msg= new aMessage();

                b_rel= inf;
        }

        protected void updateMessage(){
                pre_msg.update(cur_msg);
        }

        protected void calculateParameters(){
                //Algorithm implementation
                //Depends on the processing capabilities of the nodes
```

```
            double proportion;
            if((cur_msg.getValue()-pre_msg.getValue())!=0)
                proportion= cur_msg.getValue()==0? Math.abs(cur_msg.getValue()-pre_msg.getValue())/Math.abs(pre_msg.getValue()): Math.abs(cur_msg.getValue()-
pre_msg.getValue())/Math.abs(cur_msg.getValue());
            else
                proportion= 0;
            b=1/(1+Math.exp(-12*(proportion-0.3)));

            cur_msg.setValidity((int) Math.ceil(7/Math.pow(r/R_MIN, b*b_rel)));

            return;
        }
}
```

```
public class eSourceNode extends aSourceNode {

        eSourceNode(int i, double inf) { //constructor
            super(i, inf);
        }

        protected void calculateParameters(){
            //Algorithm implementation
            //Depends on the processing capabilities of the nodes

            double proportion;
            if((cur_msg.getValue()-pre_msg.getValue())!=0)
                proportion= cur_msg.getValue()==0? Math.abs(cur_msg.getValue()-pre_msg.getValue())/Math.abs(pre_msg.getValue()): Math.abs(cur_msg.getValue()-
pre_msg.getValue())/Math.abs(cur_msg.getValue());
            else
                proportion= 0;
            b=1/(1+Math.exp(-12*(proportion-0.3)));

            cur_msg.setValidity((int) Math.ceil(7/Math.pow(r/R_MIN, b*b_rel)));
            leaveout=(int)(1+10*Math.pow(0.35, 1.5*proportion));
```

```
            return;
        }
}
```

```
public class RelayNode extends Node<Message> {

    RelayNode(int i, double inf, double rem, double rec) { //constructor
        super(i, inf, rem, rec);

        cur_msg= new Message();
        pre_msg= new Message();

        e= State.SUSCEPTIBLE;
    }

    protected void updateMessage(){
        pre_msg.update(cur_msg);
    }

    public void sense(Message in){
        if(cur_msg.getValueTime()<in.getValueTime()){    //context renewal
            e= State.INFECTED;
            cur_msg.update(in);
            sensed=true;
        }
        return;
    }
}
```

```
public class aRelayNode extends Node<aMessage> {
```

```
public aRelayNode(int i, double inf) { //constructor
	super(i, inf, 0, 0);

	cur_msg= new aMessage();
	pre_msg= new aMessage();

	e= State.SUSCEPTIBLE;
}

protected void updateMessage(){
	pre_msg.update(cur_msg);
}

protected void updateStatus(){
	if(e==State.INFECTED){
		int validity= cur_msg.getValidity()-1;
		if(validity<=0){
			e= State.SUSCEPTIBLE;	//Recover
			sensed=false;
		}
		else
			cur_msg.setValidity(validity);
	}
	return;
}

public void sense(Message in){
	if(cur_msg.getValueTime()<in.getValueTime()){	//context renewal
		e= State.INFECTED;
		sensed=true;
		if(cur_msg.getClass().isAssignableFrom(in.getClass()))
			cur_msg.update((aMessage)in);
		else{
			cur_msg.setValue(in.getValue());
			cur_msg.setValueTime(in.getValueTime());
```

```
                        cur_msg.setValidity(1);
                    }
                }
                return;
        }
}
```

```
public class Message {
        double value;
        int value_time;

        Message(){
                value=0;
                value_time=0;
        }

        public void update(Message in){
                value=in.getValue();
                value_time=in.getValueTime();
        }

        public void setValue(double in){
                value=in;
                return;
        }

        public void setValueTime(int in){
                value_time=in;
                return;
        }

        public double getValue(){
                return value;
```

```
        }

        public int getValueTime(){
                return value_time;
        }
}
```

```
public class aMessage extends Message {
        int validity;

        aMessage(){
                validity=0;
        }

        public void update(aMessage in){
                super.update(in);
                validity=in.getValidity();
                return;
        }

        public void setValidity(int in){
                validity=in;
                return;
        }

        public int getValidity(){
                return validity;
        }
}
```

```
public class Position {
```

```
        private double pos_x;
        private double pos_y;

        public void move(double x, double y){
                pos_x=x;
                pos_y=y;
                return;
        }

        public double distance(Position p){
                return Math.sqrt(Math.pow(p.getX()-pos_x, 2)+Math.pow(p.getY()-pos_y, 2));
        }

        public double getX(){
                return pos_x;
        }

        public double getY(){
                return pos_y;
        }
}
```

Δεδομένου ότι παρέχεται η δυνατότητα προσομοίωσης ενός μόνον αλγόριθμου κάθε φορά, ο κώδικας που προηγήθηκε εκτελέσθηκε δύο φορές (μια για κάθε παραλλαγή) στο πλαίσιο της συγκριτικής μελέτης της προτεινόμενης εκδοχής του αλγόριθμου aSIS. Και στις δύο εκτελέσεις, όμως, χρησιμοποιήθηκε το ίδιο δίκτυο, ήτοι τα ίδια αρχεία δεδομένων και συντεταγμένων (sample.txt και trace.txt) και επιλέχθηκαν οι ίδιες παράμετροι προσομοίωσης (μέγεθος δικτύου, ρυθμός μόλυνσης), για λόγους συγκρισιμότητας των αποτελεσμάτων.

ΠΑΡΑΡΤΗΜΑ ΙΙ

Οι συντεταγμένες που χρησιμοποιήθηκαν από τα προγράμματα προσομοίωσης για την πλοήγηση των κινητών κόμβων στο δίκτυο αισθητήρων παρήχθησαν από την πλατφόρμα Mobisim, μια εφαρμογή διαχείρισης δεδομένων κίνησης που προσφέρει τη δυνατότητα αναπαραγωγής της συμπεριφοράς μιας πληθώρας αλγόριθμων κινητικότητας σε ad–hoc δίκτυα κινητών κόμβων. Εξαιτίας της απλότητας και της ευρείας αποδοχής του, ως μοντέλο κίνησης στην παρούσα εργασία επιλέχθηκε, μεταξύ των λοιπών διαθέσιμων υλοποιήσεων, το Random Waypoint (RWP).

Στο μοντέλο αυτό, αρχικά κάθε κόμβος επιλέγει τυχαία μια θέση εντός του πλαισίου προσομοίωσης και κατευθύνεται σε αυτή με επίσης τυχαία, σταθερή ταχύτητα $v \in [v_{min}, v_{max}]$. Όταν καταλήξει στον προορισμό του, ο κόμβος αναπαύεται για $\mathrm{t_{pause}}$ χρονικό διάστημα, με την πάροδο του οποίου επιλέγει νέα κατεύθυνση. Η διαδικασία επαναλαμβάνεται από την αρχή μέχρι την ολοκλήρωση της προσομοίωσης.

Πίνακας 2: Χαρακτηριστικά Μοντέλου Κινητικότητας

Παράμετρος	***Τιμή***
Μοντέλο κινητικότητας	RWP
Διαστάσεις grid	500 x 500
Πλήθος κόμβων	25
Ελάχιστη ταχύτητα (v_{min})	5
Μέγιστη ταχύτητα (v_{max})	10
Μέγιστος χρόνος ανάπαυσης (t_{pause})	10

Στο κάτωθι γράφημα απεικονίζεται μέρος της διαδρομής των κόμβων 0 (κόκκινο άνυσμα) και 12 (μπλε άνυσμα) μετά την εφαρμογή του μοντέλου RWP σε δίκτυο με χαρακτηριστικά όπως καταγράφονται στον πίνακα.

Εικόνα 2: Μέρος της διαδρομής του κόμβου – πηγή και ενός τυχαίου κόμβου - αναμεταδότη (αναπαράσταση μέσω Mobisim) [10]

ΑΝΑΦΟΡΕΣ

[1] C. Anagnostopoulos, O. Sekkas and S. Hadjiefthymiades, An Adaptive Epidemic Information Dissemination Model for Wireless Sensor Networks, *Elsevier, Pervasive and Mobile Computing Journal*, vol. 8, no. 5, 2012, pp. 751-763

[2] Ο. Σέκκας, «Διαχείριση Πληροφορίας Πλαισίου για το Διάχυτο Υπολογισμό», Διδακτορική Διατριβή, Τμήμα Πληροφορικής & Τηλεπικοινωνιών, Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών, 2010

[3] C. Anagnostopoulos and S. Hadjiefthymiades, On the Application of the Epidemical Spreading in Collaborative Context-Aware Computing, *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 12, no. 4, 2008, [doi>10.1145/1508285.1508291]

[4] Κ. Λυμπέρης, «Μελέτη Επιδημικών Αλγόριθμων σε Περιβάλλοντα Διάχυτου Υπολογισμού», Διπλωματική Εργασία, Σχολή Θετικών Επιστημών & Τεχνολογίας, Ελληνικό Ανοικτό Πανεπιστήμιο, 2008

[5] Pervasive Computing Research Group; p-comp.di.uoa.gr

[6] C. Anagnostopoulos, "Modeling and architecture of Mobile Computing Systems", Department of Informatics & Telecommunications, National and Kapodistrian University of Athens; cgi.di.uoa.gr/~phdsbook/files/anagnostopoulos.pdf

[7] T. Tassier, "Introduction to the Modeling of Epidemics – SIS Models"; www.facweb.iitkgp.ernet.in/~niloy/COURSE/Spring2006/CNT/Resource/SI-SIS.pdf

[8] P. Eugster et al., "Epidemic Information Dissemination in Distributed Systems", *ACM Journal*, vol. 37, no. 5, 2004, pp. 60-67 [doi>10.1109/MC.2004.1297243]

[9] M. Akdere et al., "A Comparison of Epidemic Algorithms in Wireless Sensor Networks", *ACM Journal*, vol. 29, no. 13-14, 2006, pp. 2450-2457 [doi>10.1016/j.comcom.2006.02.010]

[10] M. Moshref, "Mobility Simulator (MobiSim)", blog, 2009; masoudmoshref.com/old/myworks/documentpages/mobility_simulator.htm

[11] Oracle, "Overview (JavaPlatform SE7)", 2009; docs.oracle.com/javase/7/docs/api/

[12] The Eclipse Foundation, "Eclipse Mars1 Packages", 2009; www.eclipse.org/downloads/packages/release/Mars/1