



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΗΛΕΚΤΡΙΚΩΝ ΒΙΟΜΗΧΑΝΙΚΩΝ ΔΙΑΤΑΞΕΩΝ ΚΑΙ
ΣΥΣΤΗΜΑΤΩΝ ΑΠΟΦΑΣΕΩΝ

Μοντελοποίηση και Προσομοίωση Κίνησης Ανθρώπων σε Αστικό Περιβάλλον με χρήση Agent- Based Modeling

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΙΑΣΩΝ ΛΑΖΑΡΟΣ ΠΑΠΑΓΕΩΡΓΙΟΥ

Επιβλέπων : Δημήτριος Ασκούνης
Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2020



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΗΛΕΚΤΡΙΚΩΝ ΒΙΟΜΗΧΑΝΙΚΩΝ ΔΙΑΤΑΞΕΩΝ ΚΑΙ
ΣΥΣΤΗΜΑΤΩΝ ΑΠΟΦΑΣΕΩΝ

Μοντελοποίηση και Προσομοίωση Κίνησης Ανθρώπων σε Αστικό Περιβάλλον με χρήση Agent- Based Modeling

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΙΑΣΩΝ ΛΑΖΑΡΟΣ ΠΑΠΑΓΕΩΡΓΙΟΥ

Επιβλέπων : Δημήτριος Ασκούνης

Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 27^η Νοεμβρίου 2020

.....
Δημήτριος Ασκούνης
Καθηγητής Ε.Μ.Π.

.....
Ιωάννης Ψαρράς
Καθηγητής Ε.Μ.Π.

.....
Χάρης Δούκας
Αν. Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2020

.....
ΙΑΣΩΝ ΛΑΖΑΡΟΣ ΠΑΠΑΓΕΩΡΓΙΟΥ

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

Copyright © ΙΑΣΩΝ ΛΑΖΑΡΟΣ ΠΑΠΑΓΕΩΡΓΙΟΥ, 2020
Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

Περίληψη

Στην παρούσα διπλωματική εργασία αναλύεται και μελετάται η τεχνική μοντελοποίησης που βασίζεται σε αντιπροσωπευτικούς πράκτορες (Agent-Based Modeling), παρουσιάζεται η χρήση του Agent-Based Modeling σε συνδυασμό με πραγματικά γεωγραφικά δεδομένα μέσω της ανάπτυξης εφαρμογών στην πλατφόρμα GAMA, παρέχονται οδηγίες για την εγκατάσταση και χρήση της πλατφόρμας GAMA, και με χρήση των εργαλείων αυτών επιλύεται με τη δημιουργία μίας παραμετροποιήσιμης προσομοίωσης ένα πρακτικό πρόβλημα, η ανεύρεση ενός παιδιού που έχει χαθεί σε αστικό περιβάλλον.

Ειδικότερα, στο Agent Based Modeling προσδιορίζονται οι παράγοντες που εμπλέκονται σε ένα πρόβλημα και στη λύση του, και καθένας από τους παράγοντες μοντελοποιείται ως ξεχωριστός πράκτορας (agent) με δικά του χαρακτηριστικά, ιδιότητες και συμπεριφορά. Με αυτόν τον τρόπο, για την μοντελοποίηση ενός εξαιρετικά πολύπλοκου συστήματος δεν απαιτείται η μοντελοποίηση του ολικού συστήματος, αλλά αρκεί η κωδικοποίηση των επιμέρους απλών μερών του, που είναι ευκολότερη και απλούστερη. Με την μοντελοποίηση αυτή των επιμέρους μερών και τον προγραμματισμό των κανόνων που διέπουν τη συμπεριφορά τους, τα επιμέρους μέρη μέσα από την αλληλεπίδρασή τους δημιουργούν την ολική συμπεριφορά του πολύπλοκου συστήματος, και δίνουν την δυνατότητα μοντελοποίησής του. Έτσι μπορούμε να εξερευνήσουμε και να επιλύσουμε το πρόβλημα. Η δυνατότητα αυτή του Agent Based Modeling έχει οδηγήσει αρκετούς να πιστεύουν ότι το Agent-Based Modeling «είναι ένας τρίτος τρόπος για να κάνει κανείς επιστήμη» [1].

Προκειμένου να κάνουμε χρήση του Agent-Based Modeling στην ανθρώπινη κινητικότητα, απαιτείται να εισάγουμε ως συνιστώσες τον χώρο και τον χρόνο. Για να το πραγματοποιήσουμε αυτό χρειαζόμαστε μια πλατφόρμα που να συνδυάζει το Agent-Based Modeling με πραγματικά γεωγραφικά δεδομένα. Μία τέτοια πλατφόρμα είναι η πλατφόρμα GAMA (GIS & Agent-based Modeling Architecture). Και επειδή όπως πολλές πλατφόρμες που υποστηρίζουν νέες και ισχυρές τεχνολογίες μοντελοποίησης, έτσι και η GAMA είναι δύσκολη για κάποιον νέο χρήστη, παρέχονται οδηγίες για εγκατάσταση και χρήση της.

Επιδιώκοντας να αναδείξουμε στη πράξη τις δυνατότητες που προσφέρει η μοντελοποίηση με Agent Based Modeling και χρήση της πλατφόρμας GAMA στην επίλυση προβλημάτων ανθρώπινης κινητικότητας, επιλέξαμε να λύσουμε το πρόβλημα της εύρεσης ενός χαμένου παιδιού σε αστικό περιβάλλον με χρήση πραγματικών γεωγραφικών δεδομένων χαρτών και χρήση της πλατφόρμας GAMA. Το χαμένο παιδί κινείται στο χάρτη της προσομοίωσης, έχοντας μία αρχική θέση και κάποια σημεία ενδιαφέροντος (π.χ. παιδικές χαρές). Ταυτόχρονα, άνθρωποι μετακινούνται στο χάρτη κάνοντας τις καθημερινές δουλειές τους. Ένα ποσοστό αυτών βρίσκονται σε επιφυλακή για την εύρεση του παιδιού. Ένα πιθανοτικό μοντέλο προσομοιώνει την επιφυλακή των ανθρώπων, σε συνδυασμό με τον τρόπο με τον οποίο κινούνται οι άνθρωποι, και εξάγονται συμπεράσματα σχετικά με την εύρεση του παιδιού.

Τέλος, τίθενται με σαφήνεια οι προδιαγραφές για την υλοποίηση της προσομοίωσης προς επίλυση του προβλήματος, και αναλύεται το πώς οι προδιαγραφές ικανοποιήθηκαν σε επίπεδο κώδικα. Δίνονται επίσης οδηγίες χρήσης τόσο για την πλατφόρμα, όσο και για την προσομοίωση που κατασκευάστηκε.

Λέξεις Κλειδιά: Μοντελοποίηση κατά αντιπροσωπευτικούς πράκτορες, πλατφόρμα GAMA, Κίνηση ανθρώπων σε Αστικό Περιβάλλον, Προσομοίωση Ανθρώπινης Κίνησης.

Abstract

For the purposes of this diploma thesis, we have studied and analysed Agent-Based Modeling, we have presented the concept of using Agent-Based Modeling in combination with geographical data and systems (GIS) by developing an application using the GAMA platform development environment, we have provided instructions for installing and using the GAMA platform, and we have deployed all of the above tools to create a highly customisable simulation model in order to solve a practical problem; finding a missing child in an urban environment.

More specifically, in Agent-Based Modeling all of the factors that comprise a problem and its solution are defined, and each factor can be modeled into an independent agent with their own characteristics, qualities and behaviours. Thus, in order to model an exceptionally complicated system, we do not need to model the system in its undivided, complex form, but, instead, we can program the behaviours of the individual, simpler parts of the system, something that can be achieved with substantially smaller programming effort. By modeling the individual agents and programming the rules that govern their behaviour, through the simultaneous operations and interactions of the agents, we can re-create the overall behaviour of the complex initial system, which enables us to create its model. In this way we can explore the complex problem and find the solution. This feature of Agent Based Modeling has led many to believe that Agent-Based Modeling "is a third way to do science" [1].

In order for Agent-Based Modeling to be applied to human mobility, space and time are required to be introduced as components. For this to be realised a platform that combines Agent-Based Modeling with actual geographic data is needed. GAMA platform is such. And since, like many platforms that support new and robust modeling technologies, GAMA is challenging for the new user, installation and operation instructions are provided.

Seeking to highlight the possibilities of modeling with Agent Based Modeling and the opportunities entailed in developing models with the GAMA platform for human mobility problems in practice, we have elected to explore and attempt to solve the problem of finding a missing child in an urban environment, using real geographical and mapping data, and the use of GAMA platform for developing our model. The missing child has a starting position and moves between targets on the road network represented by the simulation's map. Some of the buildings contained in the map may also be possible points of interest for the missing child (e.g. playing grounds). At the same time a specific number of people are moving on the map, executing their daily routine. A certain percentage of the people moving around the map are on alert for locating the child. A probabilistic model simulates people's alertness, factoring in the means of transport that people are using when encountering the missing child and conclusions are drawn about the objective of finding the missing child.

Finally, the specifications for the implementation of the simulation that is developed in order to solve the problem are clearly established. Additionally, the way that the specifications are met at the code level is analysed. Moreover, clear instructions on how to operate the GAMA platform and on how to execute the simulation are provided.

Key Words: Agent-Based Modeling, Gama Platform, Human Movement in an Urban Environment, Simulation of Human Mobility.

Πρόλογος

Η παρούσα διπλωματική εργασία εκπονήθηκε υπό την επίβλεψη του Καθηγητή Δημητρίου Ασκούνη στο Εργαστήριο Συστημάτων Αποφάσεων και Διοίκησης της Σχολής Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Εθνικού Μετσόβιου Πολυτεχνείου.

Το αντικείμενο της εργασίας συνίσταται στην μελέτη του Agent-Based Modeling και της πλατφόρμας GAMA, και στην χρήση τους για τη δημιουργία προσομοίωσης κίνησης ατόμων σε αστικό περιβάλλον, με χρήση πραγματικών γεωγραφικών δεδομένων, με ειδικότερη εφαρμογή στην εξεύρεση ενός παιδιού που έχει χαθεί σε ένα τέτοιο περιβάλλον.

Θα ήθελα σε αυτό το σημείο να εκφράσω τις βαθύτατες ευχαριστίες μου στην επιβλέπουσα της εργασίας μου, υποψήφια διδάκτορα Αριάδνη Μιχαλίτση-Ψαρρού για την εξαιρετική συνεργασία, για την εμπιστοσύνη που μου επέδειξε και για την αμέριστη υποστήριξη και καθοδήγηση καθ' όλη τη διάρκεια εκπόνησης της παρούσας εργασίας.

Ιάσων Παπαγεωργίου

Οκτώβριος 2020

Πίνακας Περιεχομένων

Κεφάλαιο 1. Εισαγωγή	10
1.1. Σκοπός της διπλωματικής εργασίας	11
1.1.1. Συνεισφορά	12
1.2. Δομή της διπλωματικής εργασίας	12
Κεφάλαιο 2. Agent-Based Modeling.....	14
2.1. Εισαγωγή στο Agent-Based Modeling	15
2.2. Τι είναι το Agent-Based Modeling	16
2.3. Ετερογένεια, αλληλεπίδραση και το φαινόμενο emergence.	18
2.4. Δυνατότητες και Οφέλη	20
2.5. Εφαρμογές	22
2.5.1. Agent-Based Μοντέλο της πανδημίας Covid-19 για την προσομοίωση των οικονομικών και επιδημιολογικών επιπτώσεων των μέτρων κοινωνικής αποστασιοποίησης [11].....	22
2.5.2. Agent-Based μοντέλο για τη διαχείριση περιστατικών πλημμύρας με ανάλυση ρίσκου [12].....	25
2.6. Συστήματα Γεωγραφικών Πληροφοριών (GIS) και Agent-Based Modeling.....	27
2.7. Multi-Level Agent-Based Modeling	29
Κεφάλαιο 3. Η πλατφόρμα Gama.....	31
3.1. Εισαγωγή στην Πλατφόρμα GAMA	32
3.2. Η πλατφόρμα GAMA.....	32
3.3. Επιλογή της πλατφόρμας GAMA	34
3.4. Εγκατάσταση της GAMA-Platform.....	35
3.4.1. Απαιτήσεις Συστήματος.....	36
3.4.2. Διαδικασία Εγκατάστασης.....	37
3.4.3. Δημιουργία GAMA Models Workspace.....	39
3.5. Μειονεκτήματα και μελλοντική λειτουργία της GAMA	41
Κεφάλαιο 4. Ανάλυση του Μοντέλου Προσομοίωσης	42
4.1. Ορισμός του Προβλήματος.....	43
4.2. Προδιαγραφές και Επιλογές Υλοποίησης.....	44
4.3. Εισαγωγή Δεδομένων Χαρτών στο Μοντέλο	47
4.3.1. Η πλατφόρμα Open Street Map	48
4.3.2. Εξαγωγή Χαρτών .osm	49

4.3.2.1 Απευθείας Εξαγωγή Χαρτών από την ιστοσελίδα OSM	49
4.3.2.2 Εξαγωγή Χαρτών μέσω της Extract.BBBike.....	52
4.4. Κώδικας του Μοντέλου	58
4.4.1. Κωδικοποίηση Χαρτών.....	61
4.4.2. Προγραμματισμός People Agents.....	63
4.4.3. Προγραμματισμός Missing Person	66
4.4.4. Κωδικοποίηση Πιθανοτικού Μοντέλου Εύρεσης.....	70
4.4.5. Στοιχεία Κίνησης	72
4.4.6. Κωδικοποίηση Λειτουργιών Εκτέλεσης.....	75
Κεφάλαιο 5. Εκτέλεση και Εποπτεία της Προσομοίωσης	78
5.1. Φόρτωση Προσομοίωσης στην GAMA	79
5.2. Εκτέλεση της Προσομοίωσης	81
5.2.1. Πείραμα Κανονικής Λειτουργίας και Εποπτεία	81
5.2.2. Πείραμα Εκτέλεσης Πολλαπλών Προσομοιώσεων (Batch Experiment) ..	91
5.2.2.1 Χρόνος Πρώτης Εύρεσης Missing Person.....	91
5.2.2.2 Φορές Εύρεσης Missing Person σε Συγκεκριμένο Χρόνο.....	93
Κεφάλαιο 6. Συμπεράσματα και Προοπτικές.....	95
Βιβλιογραφία	101

Κεφάλαιο 1. Εισαγωγή

1.1. Σκοπός της διπλωματικής εργασίας

Από την αρχή της ανθρώπινης ιστορίας, ο άνθρωπος είχε από τη φύση του την τάση να δημιουργεί και να χρησιμοποιεί μοντέλα προσομοίωσης πραγματικότητας σε διάφορους τομείς της ζωής του, χρησιμοποιώντας τις δυνατότητες που του παρείχαν τα εργαλεία της κάθε εποχής [2]. Σήμερα, με την εκθετική αύξηση της επεξεργαστικής δύναμης των υπολογιστών και με την ανάπτυξη της μαθηματικής επιστήμης, η προσομοίωση έχει εξελιχθεί σε ένα εξαιρετικό εργαλείο ανάλυσης και μελέτης συστημάτων και συμπεριφορών.

Αναφερόμενοι στην κίνηση ανθρώπων στο αστικό περιβάλλον και σε προβλήματα που συνδέονται με την ανθρώπινη κινητικότητα, διαπιστώνουμε ότι όχι σε πολύ μακρινό χρόνο στο παρελθόν, σημαντικό μέρος των προβλημάτων αυτών μπορούσε να προσεγγιστεί μόνο με μαθηματικά εργαλεία, κάτι που αποτελούσε τον κυρίαρχο τρόπο επίλυσης προβλημάτων ανθρώπινης κίνησης μέχρι σήμερα.

Με την ραγδαία όμως ανάπτυξη των υπολογιστικών συστημάτων, εργαλείων λογισμικού και την εκθετική αύξηση σε υπολογιστική δύναμη που υπήρξε κυρίως από τις αρχές του 21^{ου} αιώνα, δημιουργήθηκαν πλέον νέες δυνατότητες επίλυσης των προβλημάτων κίνησης με τη χρήση μοντέλων προσομοίωσης.

Δημιουργώντας το μοντέλο ενός προβλήματος σε υπολογιστή, είτε με τη μακροσκοπική προσέγγιση της απευθείας μίμησης των αποτελεσμάτων του, είτε με τη μικροσκοπική προσέγγιση της αποδόμησης του, στα επιμέρους συστατικά του και την προσομοίωση της συμπεριφοράς αυτών, την οποία πρεσβεύει η μοντελοποίηση βασισμένη σε αντιπροσωπευτικούς πράκτορες (Agent-Based Modeling), μπορούμε μέσω της συμπεριφοράς του μοντέλου να μάθουμε πολλά για το πραγματικό πρόβλημα. Αξιοποιώντας και τη δυνατότητα που μας παρέχει το μοντέλο να πειραματιστούμε με χιλιάδες ή εκατομμύρια συνδυασμούς λύσεων (κάτι πρακτικά αδύνατο στην πραγματική ζωή), μπορούμε να οδηγηθούμε στη βέλτιστη λύση του πραγματικού προβλήματος.

Αυτός είναι και ο στόχος της παρούσας διπλωματικής. Συγκεκριμένα, αντί να προσεγγίσουμε την ανθρώπινη κινητικότητα με συμβατικούς όρους μελέτης, θα επικεντρωθούμε στη μελέτη των δυνατοτήτων μοντελοποίησης και συγκεκριμένα στο Agent-Based Modeling, καθώς και στην πλατφόρμα GAMA, μία ανερχόμενη πλατφόρμα μοντελοποίησης και ανάπτυξης Agent-Based Modeling εφαρμογών, αρμονικά συνδυασμένων με γεωγραφικά δεδομένα. Στη συνέχεια, θα ορίσουμε ένα σαφές πρόβλημα κίνησης ανθρώπων σε αστικό περιβάλλον και θα δείξουμε όλα τα βήματα, συλλογιστικά αλλά και ανάπτυξης κώδικα, για την επίλυσή του.

Φιλοδοξούμε αυτή η διπλωματική να αποτελέσει αφετηρία και αρωγό για την επίλυση προβλημάτων ανθρώπινης κινητικότητας αλλά και άλλων προβλημάτων της ανθρώπινης καθημερινότητας, μέσω προσομοίωσης με χρήση Agent-Based Modeling. Ταυτόχρονα, ευελπιστούμε να αποτελέσει εργαλείο βοήθειας και οδηγό για το πώς μπορεί να χρησιμοποιηθεί το Agent-Based Modeling σε συνδυασμό με γεωγραφικά δεδομένα και την πλατφόρμα GAMA, ώστε να οδηγηθούμε σε μία νέα οπτική του πεδίου των προβλημάτων αυτών.

1.1.1. Συνεισφορά

Η συνεισφορά της παρούσας διπλωματικής εργασίας συνοψίζεται ως εξής:

1. Μελέτη της φιλοσοφίας μοντελοποίησης του Agent-Based Modeling, των χαρακτηριστικών, των δυνατοτήτων και των εφαρμογών του, καθώς και μελέτη της σύνδεσης του Agent-Based Modeling με Συστήματα Γεωγραφικών Πληροφοριών.
2. Μελέτη της πλατφόρμας GAMA για ανάπτυξη εφαρμογών Agent-Based Modeling, και εφαρμογών Agent-Based Modeling βασισμένων σε πραγματικά γεωγραφικά δεδομένα. Ανάλυση των δυνατοτήτων και του τρόπου λειτουργίας της GAMA και τεκμηρίωση της επιλογής χρήσης της για την προσομοίωση που θα αναπτυχθεί στα πλαίσια της παρούσας διπλωματικής εργασίας.
3. Χρήση της πλατφόρμας GAMA για την δημιουργία προσομοίωσης κίνησης ατόμων σε αστικό περιβάλλον που μοντελοποιείται με πραγματικά χαρτογραφικά δεδομένα και διατύπωση οδηγιών για την χρήση και τροποποίησή της.
4. Υλοποίηση ειδικής εφαρμογής που προσομοιώνει την εύρεση ενός εξαφανισμένου ατόμου σε αστικό περιβάλλον και δημιουργία παραμετροποιήσιμων πειραμάτων εύρεσης.
5. Αξιολόγηση των συνολικών αποτελεσμάτων και παρουσίαση των τρόπων με τους οποίους μπορεί να εξελιχθεί περαιτέρω το μοντέλο και διατύπωση ιδεών για σχετικές μελλοντικές εφαρμογές.

1.2. Δομή της διπλωματικής εργασίας

Στο παρόν κεφάλαιο (Κεφάλαιο 1, Εισαγωγή) αναγράφεται ο σκοπός της διπλωματικής εργασίας και η δομή της. Στη συνέχεια (Κεφάλαιο 2, Agent-Based Modeling) περιγράφονται οι έννοιες που σχετίζονται με το Agent-Based Modeling, αναλύονται τα χαρακτηριστικά και οι περιπτώσεις στις οποίες ενδείκνυται η χρήση του για την επίλυση προβλημάτων. Παρουσιάζονται επίσης επίκαιρες και βοηθητικές στην κατανόηση του Agent-Based Modeling εφαρμογές. Στη συνέχεια του κεφαλαίου παρουσιάζεται η σύνδεση του Agent-Based Modeling με τα Συστήματα Γεωγραφικών Πληροφοριών, αναδεικνύεται η χρησιμότητα αυτής της σύνδεσης, και περιγράφεται η έννοια του Agent-Based Modeling πολλαπλών επιπέδων.

Στο τρίτο κεφάλαιο της εργασίας (Κεφάλαιο 3, Η πλατφόρμα GAMA) αναλύονται τα χαρακτηριστικά και ο τρόπος λειτουργίας μίας εκ των σημαντικότερων πλατφορμών για ανάπτυξη Agent Based Modeling εφαρμογών σε συνδυασμό με πραγματικά γεωγραφικά δεδομένα, της πλατφόρμας GAMA. Αναφέρονται οι λόγοι για τους οποίους η GAMA επιλέχθηκε για την ανάπτυξη της προσομοίωσης της εργασίας και παρέχεται ένας σύντομος οδηγός για τη διαδικασία εγκατάστασης.

Στο τέταρτο μέρος της εργασίας (Κεφάλαιο 4, Ανάλυση του Μοντέλου Προσομοίωσης) περιγράφεται συγκεκριμένα το πρόβλημα που επιχειρεί να λύσει η προσομοίωσή μας και παρουσιάζεται η προσέγγιση επίλυσης αυτού. Τίθενται οι

προδιαγραφές που θα διέπουν την προσομοίωση, και παρουσιάζεται η διαχείριση των δεδομένων εισόδου της, και συγκεκριμένα των πραγματικών, αξιόπιστων χαρτών και το πώς οι χάρτες εξάγονται από τις πλατφόρμες OSM και Extract.BBBike. Στη συνέχεια, για κάθε προδιαγραφή και περιορισμό που τέθηκε αναλύεται η αντίστοιχη υλοποίηση σε κώδικα.

Στο πέμπτο κεφάλαιο της εργασίας (Κεφάλαιο 5, Εκτέλεση και Εποπτεία της Προσομοίωσης) περιγράφεται το πώς φορτώνεται και εκτελείται η προσομοίωση που κατασκευάσαμε στην πλατφόρμα GAMA. Αναλύονται τα δύο είδη πειραμάτων, πειράματα κανονικής λειτουργίας και batch πειράματα, και παρατίθενται παραδείγματα εκτέλεσης για κάθε ένα.

Τέλος, στο τελευταίο κεφάλαιο (Κεφάλαιο 6, Συμπεράσματα και Προοπτικές), παρουσιάζονται τα συμπεράσματα της διπλωματικής εργασίας και εκφράζονται σκέψεις και ιδέες για τις δυνατότητες περαιτέρω αξιοποίησης και ανάπτυξης της συγκεκριμένης εργασίας καθώς και της προσομοίωσης που αναπτύχθηκε στο πλαίσió της.

Κεφάλαιο 2. Agent-Based Modeling

2.1. Εισαγωγή στο Agent-Based Modeling

Το Agent-Based Modeling είναι μία νέα προσέγγιση μοντελοποίησης που αποκτά όλο και μεγαλύτερη δυναμική και συγκεντρώνει την αυξανόμενη προσοχή της επιστημονικής κοινότητας τα τελευταία χρόνια. Το συνεχώς διεκρινόμενο ενδιαφέρον για το Agent-Based Modeling αντικατοπτρίζεται στην πληθώρα νέων εφαρμογών και χρήσεων, καθώς και στην έρευνα και τις δημοσιεύσεις σχετικά με το Agent-Based Modeling. Η πρόοδος αυτή στην ακολουθούμενη συλλογιστική, καθώς και η μεθοδολογική διαφοροποίηση που ενυπάρχει στο Agent-Based Modeling, έχει οδηγήσει αρκετούς να πιστεύουν ότι το Agent-Based Modeling «είναι ένας τρίτος τρόπος για να κάνει κανείς επιστήμη» και θα μπορούσε να εμπλουτίσει την παραδοσιακή συλλογική και επαγωγική συλλογιστική με νέες μεθόδους ανακάλυψης [1][3].

Η σύγχρονη δημοφιλία του Agent-Based Modeling οφείλεται στην ολοένα αυξανόμενη περιπλοκότητα του μοντέρνου κόσμου, και ειδικότερα στους ακόλουθους τέσσερις παράγοντες [3]:

1. Τα συστήματα που πρέπει να αναλύσουμε και να μοντελοποιήσουμε γίνονται πιο περίπλοκα όσον αφορά τις αλληλεξαρτήσεις τους και τα συμβατικά εργαλεία μοντελοποίησης μπορεί να μην είναι τόσο εφαρμόσιμα όσο στο παρελθόν.

Χαρακτηριστικό παράδειγμα αποτελεί η εκθετική αύξηση της πολυπλοκότητας του συστήματος παραγωγής και διαχείρισης ηλεκτρικής ενέργειας, όπου από ένα ανά χώρα κεντριοποιημένο ενιαίο σύστημα, οδηγούμαστε σε ένα παγκοσμιοποιημένο διασυνδεδεμένο σύστημα, όπου οι μονάδες παραγωγής, οι διανομείς, και οι καταναλωτές έχουν πλέον την ελευθερία και τη δυνατότητα να κάνουν τις επιλογές τους με βάση τα δικά τους ιδιαίτερα κριτήρια και συμφέροντα, σε ένα πολύπλοκο, αλληλεξαρτώμενο περιβάλλον.

2. Ορισμένα πολύπλοκα συστήματα ήταν αδύνατο να επιλυθούν επαρκώς με χρήση των παραδοσιακών εργαλείων και μεθόδων, γι' αυτό και η επίλυσή τους απαιτούσε την απλούστευση, με παράληψη σημαντικών πραγματικών παραμέτρων. Η δυσκολία αυτή οδηγούσε σε αποτελέσματα που απείχαν ουσιωδώς από την πραγματικότητα.

Η μοντελοποίηση των οικονομικών αγορών, για παράδειγμα, βασιζόταν παραδοσιακά στις έννοιες των τέλειων αγορών, των ομοιογενών παραγόντων και της μακροπρόθεσμης ισορροπίας, επειδή αυτές οι παραδοχές έκαναν τα προβλήματα αναλυτικά και υπολογιστικά ανιχνεύσιμα. Μπορούμε πλέον να χαλαρώσουμε αυτές τις παραδοχές, αποκτώντας έτσι μια πιο ρεαλιστική εικόνα αυτών των οικονομικών συστημάτων μέσω του Agent-Based Modeling.

3. Τα δεδομένα τα οποία συλλέγονται και αποθηκεύονται αντιπροσωπεύουν έναν εξαιρετικά λεπτομερέστερο βαθμό ανάλυσης, και μπορούν πλέον να υποστηρίξουν προσομοιώσεις σε σημαντικά μικρότερη κλίμακα (μικροκλίμακα), επιτρέποντάς μας να προσεγγίσουμε τη λειτουργία κάθε συστήματος πληρέστερα σε σχέση με την πραγματικότητα.

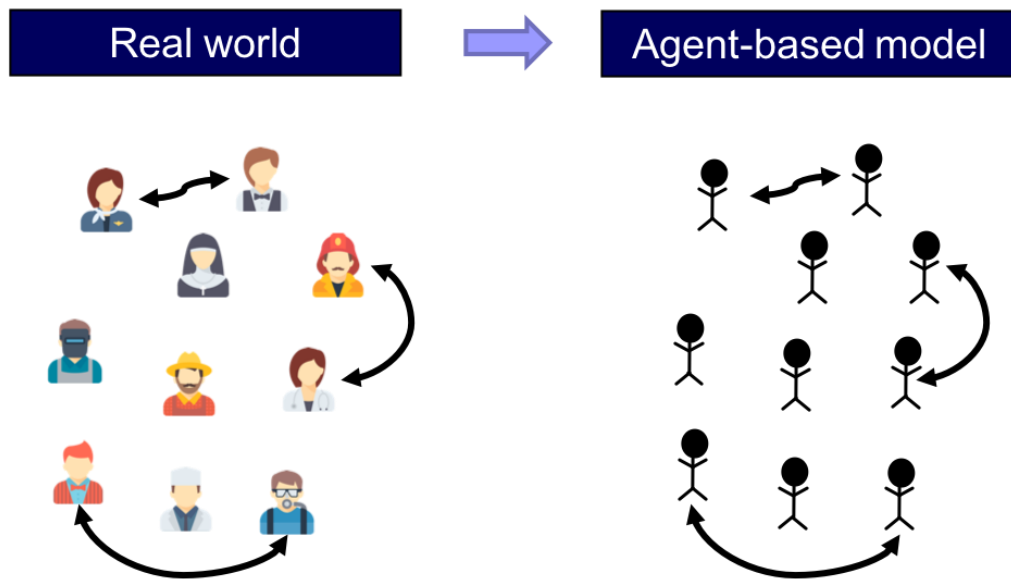
4. Και το τέταρτο, αλλά πιο σημαντικό, η υπολογιστική δύναμη αυξάνεται ταχύτατα. Μπορούμε τώρα να υπολογίσουμε μοντέλα μεγάλης κλίμακας με πολύ μεγαλύτερη ανάλυση, ώστε να έχουμε περισσότερα αξιόπιστα αποτελέσματα με μεγαλύτερη πιστότητα, κάτι που δεν ήταν εφικτό πριν από λίγα χρόνια.

2.2. Τι είναι το Agent-Based Modeling

Παρά την ευρεία χρήση του όρου και τις πολλαπλές εφαρμογές, δεν υπάρχει κάποιος σαφής και καθολικά αποδεκτός ορισμός του Agent-Based Modeling. Αυτό μάλιστα ίσως οφείλεται ακριβώς στην ευρύτητα χρήσης και στις διαφορετικές εφαρμογές. Παρόλα αυτά, η βιβλιογραφία γύρω από τον ορισμό των agents και του Agent-Based Modeling συγκλίνει σε περισσότερα πράγματα απ' ότι αποκλίνει [4]. Μπορούμε συνεπώς να καταλήξουμε σε μία άτυπη αλλά πολύ χρηστική περιγραφή.

Το Agent-Based Modeling είναι η προσέγγιση μοντελοποίησης περίπλοκων συστημάτων η οποία αποτελείται από αλληλεπιδρώντες, αυτόνομους agents (παράγοντες). Οι agents έχουν συμπεριφορές, συχνά περιγραφόμενες από απλούς κανόνες και αλληλεπιδρούν με άλλους agents, αλλάζοντας τη δική τους συμπεριφορά και επηρεάζοντας την συμπεριφορά των άλλων agents, σε μία συνεχή αλληλεπίδραση. Σχέσεις μεταξύ των οντοτήτων του συστήματος, μεταφέρονται και στις σχέσεις μεταξύ των agents στη μοντελοποίηση του συστήματος με Agent-Based Modeling (Εικόνα 2.1.).

Στα Agent-Based Modeling συστήματα ορίζονται τύποι από agents, αλλιώς κλάσεις, με βάση ομοειδή, βασικά χαρακτηριστικά (π.χ. κλάση άνθρωποι, κλάση κτήρια κτλ.). Με την ανά κλάση μοντελοποίηση των agents, μπορεί να παρατηρηθεί και καταγραφεί, η πλήρης επίδραση της ποικιλομορφίας που υπάρχει μεταξύ των agents στα χαρακτηριστικά και την συμπεριφορά τους, στην συνολική συμπεριφορά του συστήματος. Εμφανίζονται μοτίβα, δομές και καταστάσεις που δεν είχαν προγραμματιστεί ρητά στα μοντέλα, αλλά προκύπτουν μέσω των αλληλεπιδράσεων. Η έμφαση στη μοντελοποίηση της ετερογένειας των παραγόντων σε έναν πληθυσμό και στην εμφάνιση της αυτοοργάνωσης των αλληλεπιδρώντων agents, είναι δύο από τα διακριτά χαρακτηριστικά της προσομοίωσης που βασίζεται σε agents [5].



Εικόνα 2.1. Μία αντιστοίχιση ενός Social Agent-Based μοντέλου σε διαφορετικούς ανθρώπους του πραγματικού κόσμου. Απεικονίζονται με βέλη οι σχέσεις μεταξύ των ανθρώπων, οι οποίες παρατηρούμε ότι μεταφράζονται στα ίδια βέλη στο αντίστοιχο Agent-Based Model.

Πηγή: <https://www.lib.ncsu.edu/workshops/introduction-agents-based-modelling-talk>

Θα μπορούσαμε να κατανοήσουμε με απλό τρόπο τις αρχές λειτουργίας του Agent-Based Modeling μεταφερόμενοι σε ένα ποδοσφαιρικό αγώνα μέσα σε ένα κατάμεστο από φιλάθλους γήπεδο. Κάποιοι από τους φιλάθλους σε ένα κομμάτι στήλης του γηπέδου αρχίζουν να σηκώνονται από τη θέση τους, να εκτείνουν μπροστά τα χέρια τους και να κάθονται μετά από μερικά δευτερόλεπτα. Πριν καθίσουν οι προηγούμενοι, η αμέσως επόμενη στήλη φιλάθλων εκτελεί ακριβώς το ίδιο. Στη συνέχεια η επόμενη και η επόμενη, μέχρι που ολοκληρώνεται ο κύκλος σε ολόκληρο το γήπεδο. Ο κάθε ένας από τους φιλάθλους ατομικά, εκτελεί μία εξαιρετικά απλή κίνηση. Σηκώνεται, εκτείνει τα χέρια του και κάθεται ξανά. Όλοι μαζί όμως, ως σύνολο, σχηματίζουν ένα περίπλοκο ανθρώπινο κύμα (Εικόνα 2.2.). Στο παράδειγμα αυτό απεικονίζεται το πως απλές μεμονωμένες κινήσεις κάθε ενός εκ των φιλάθλων δημιουργούν ένα σύνθετο ολικό αποτέλεσμα. Το Agent-Based Modeling βασίζεται ακριβώς στην ίδια αρχή. Προσομοιώνει τις απλές μεμονωμένες συμπεριφορές των agents ενός συστήματος, οι οποίες αποδίδουν ένα σύνθετο ολικό αποτέλεσμα [6].



Εικόνα 2.2. Πλήθος φιλάθλων εκτελούν το «κύμα» στο κύπελλο συνομοσπονδιών, Φρανκφούρτη 2005.

Πηγή: [https://en.wikipedia.org/wiki/Wave_\(audience\)#/media/File:Confed-Cup_2005_-_Laolawelle.JPG](https://en.wikipedia.org/wiki/Wave_(audience)#/media/File:Confed-Cup_2005_-_Laolawelle.JPG)

2.3. Ετερογένεια, αλληλεπίδραση και το φαινόμενο emergence.

Η **ετερογένεια** και η **αλληλεπίδραση** που αναφέρθηκαν προηγουμένως, και το **φαινόμενο emergence** το οποίο θα αναλυθεί στη συνέχεια, είναι τρεις έννοιες που βρίσκονται στο επίκεντρο του Agent-Based Modeling. Η κατανόηση της σημασίας τους και του ρόλου τους στο Agent-Based Modeling είναι ιδιαίτερα σημαντική.

1. Ετερογένεια:

Η ετερογένεια στο Agent-Based Modeling έγκειται στη δυνατότητα των agents να διακατέχονται από διαφορετικά χαρακτηριστικά, ιδιότητες και συμπεριφορές. Τα χαρακτηριστικά και οι συμπεριφορές του agent μπορεί να ποικίλλουν ως προς την ποικιλομορφία και την πολυπλοκότητά τους, το πόσες και ποιες πληροφορίες λαμβάνονται υπόψη στις αποφάσεις του, τα αντανακλαστικά του agent που καθορίζουν το πώς λειτουργεί σε σχέση με τον εξωτερικό κόσμο, την άποψη του agent για τις πιθανές αντιδράσεις άλλων agents ως απόκριση στις ενέργειές του, και την έκταση της μνήμης των προηγούμενων γεγονότων που ο agent διατηρεί και χρησιμοποιεί στη λήψη των

αποφάσεών του. Οι agents μπορούν επίσης να διαθέτουν διαφορετικά ποσά πόρων ή να συσσωρεύουν διαφορετικά επίπεδα πόρων ως αποτέλεσμα αλληλεπιδράσεων με άλλους agents, κάτι που τους διαφοροποιεί ακόμα περισσότερο [4].

2. Αλληλεπίδραση:

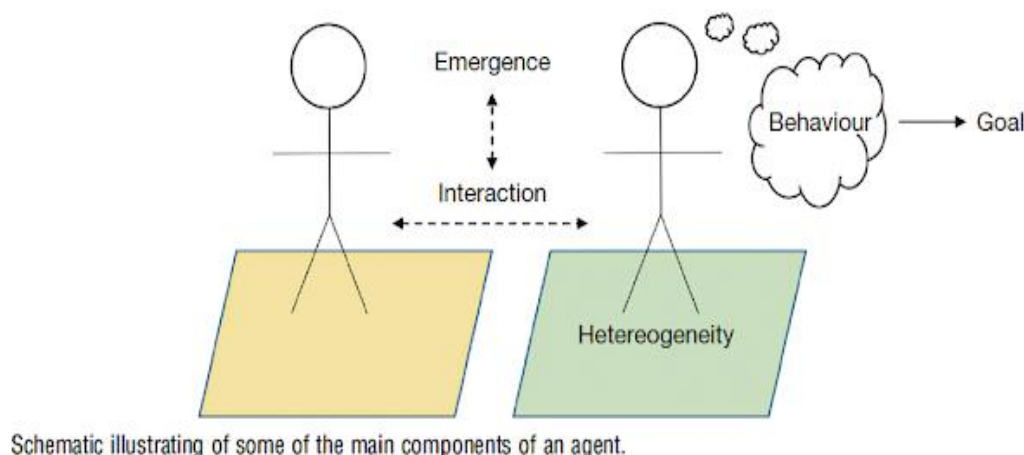
Η αλληλεπίδραση είναι ένα είδος διάδρασης που συμβαίνει καθώς δύο ή περισσότερα αντικείμενα επηρεάζουν το ένα το άλλο. Η ιδέα ενός αμφίδρομου αποτελέσματος είναι ουσιαστική στην έννοια της αλληλεπίδρασης, σε αντίθεση με το μονόδρομο αιτιώδες αποτέλεσμα [7].

Σε ένα περίπλοκο σύστημα που μοντελοποιείται με Agent-Based Modeling, η αλληλεπίδραση μεταξύ των agents αποτελεί την πεμπτούσια της συμπεριφοράς του συστήματος καθώς και καθοριστικό παράγοντα του αποτελέσματος. Οι διαφορετικές ιδιότητες, συμπεριφορές και αντανakλαστικά των agents καθώς και οι πολλαπλές επιλογές για τον χώρο και τον τρόπο που θα γίνει η αλληλεπίδραση, αποτελούν συστατικά στοιχεία της μοντελοποίησης με Agent-Based Modeling.

3. Φαινόμενο Emergence:

Το φαινόμενο δημιουργίας και ανάδειξης συμπεριφορών λόγω αλληλεπίδρασης (emergence) παρατηρείται όταν μία οντότητα έχει ιδιότητες ή συμπεριφορές οι οποίες δεν παρατηρούνται στα συστατικά της στοιχεία όταν αυτά εξετάζονται ανεξάρτητα, αλλά εμφανίζονται μόνο όταν τα μέρη αλληλεπιδρούν ως μέρη ενός ευρύτερου συνόλου. Στα συστήματα στα οποία παρατηρείται το φαινόμενο emergence, «το σύνολο είναι πλουσιότερο σε συμπεριφορές και ιδιότητες από το άθροισμα των μερών που το απαρτίζουν» [8].

Ένα από τα απλούστερα παράδειγμα του φαινομένου emergence στην καθημερινή ζωή είναι αυτό του συστήματος ενός ποδηλάτου και του αναβάτη του. Εξετάζοντας το σύστημα αυτό, παρατηρούμε μία συνεχόμενη ομαλή κίνηση προς τα εμπρός που δημιουργείται όταν αναβάτης και ποδήλατο αλληλεπιδρούν. Εξετάζοντας όμως το κάθε ένα συστατικό, αναβάτη και ποδήλατο, ξεχωριστά, καταλαβαίνουμε ότι κανένα από τα δύο μέρη δεν μπορεί να παράγει την συνεχόμενη ομαλή κίνηση μόνο του, χωρίς την αλληλεπίδραση με το άλλο.



Εικόνα 2.3. Οι βασικές έννοιες της ετερογένειας, της αλληλεπίδρασης και των φαινομένων emergence σχηματικά απεικονιζόμενες σε σχέση με τους Agents [8].

2.4. Δυνατότητες και Οφέλη

Τα οφέλη του Agent-Based Modeling σε σχέση με άλλες τεχνικές μοντελοποίησης είναι πολυάριθμα. Ανεπάρκειες φυσικά επίσης υπάρχουν, όμως τα οφέλη αυτής της μεθοδολογίας ξεπερνούν τα μειονεκτήματα εάν είναι κατανοητή η φιλοσοφία, τα δυνατά σημεία και οι δυνατότητες του Agent-Based Modeling.

Συγκεκριμένα, ο σκοπός μιας προσομοίωσης δεν είναι να δημιουργήσει ένα ακριβές αντίγραφο οποιουδήποτε συγκεκριμένου συστήματος ή περιβάλλοντος, αλλά να βοηθήσει στην εξερεύνηση των συνεπειών διάφορων συνδυασμών συμπεριφορών των agents. Με άλλα λόγια, η προσομοίωση θα πρέπει να χρησιμοποιείται ως εργαλείο για τη βελτίωση της θεωρίας [9]. Προσομοιώνοντας μια προσέγγιση της πραγματικής συμπεριφοράς που μπορεί να είναι δύσκολο να συλληφθεί σε στατικά μοντέλα, το Agent-Based Modeling εστιάζει στον τρόπο με τον οποίο οι διαδικασίες εξελίσσονται με την πάροδο του χρόνου και πώς οι συμπεριφορές μπορεί να αλλάξουν για να επηρεάσουν τα αποτελέσματα ενός εξελισσόμενου συστήματος. Agent-Based μοντέλα παρέχουν μια μεθοδολογία στην οποία αυτά τα μοτίβα μπορούν να αναπαραχθούν και στη συνέχεια να χρησιμοποιηθούν για τη μελέτη όλων των ενδεχομένων αποτελεσμάτων [10].

Κάποιος μπορεί εύκολα να μελετήσει συμπεριφορές μεμονωμένων Agents (μικροεπίπεδο), υποομάδων Agents, καθώς και συνδυασμένες συμπεριφορές Agents (μακροεπίπεδο), με διαφορετικά επίπεδα κανόνων να συνυπάρχουν σε ένα μόνο μοντέλο. Λόγω αυτής της δυνατότητας, μπορούν να δημιουργηθούν δυναμικά περιβάλλοντα όπου οι agents εισέρχονται και εξέρχονται από το σύστημα (για παράδειγμα σε μία προσομοίωση μίας ανταγωνιστικής αγοράς, οι επιχειρήσεις με χαμηλή απόδοση εγκαταλείπουν τον χώρο αγοράς ή/και νέοι ανταγωνιστές εισέρχονται στην αγορά) [10].

Αξίζει επίσης να αναλυθούν μερικές από τις καταστάσεις στις οποίες το Agent-Based Modeling μπορεί να προσφέρει ιδιαίτερα πλεονεκτήματα έναντι των συμβατικών τεχνικών προσομοίωσης όπως το discrete event simulation, το systems dynamics και άλλες ποσοτικές τεχνικές μοντελοποίησης. Είναι χρήσιμο να προσανατολιζόμαστε στη χρήση του Agent-Based Modeling όταν ένα ή περισσότερα από τα παρακάτω κριτήρια ικανοποιούνται [3] [10]:

- Όταν το πρόβλημα στη φυσική του ερμηνεία αποτελείται από agents.
- Όταν μας ενδιαφέρει τόσο το μακροσκοπικό όσο και το μικροσκοπικό επίπεδο ανάλυσης.
- Όταν μπορούν να παρατηρηθούν φαινόμενα emergence.
- Όταν οι δομικές αλλαγές της διαδικασίας πρέπει να είναι ενδογενή αποτελέσματα του μοντέλου, αντι για μία ακόμη είσοδός του.
- Όταν συνεμπλεκόμενα συστήματα αλληλεπιδρούν στο ίδιο περιβάλλον (π.χ. ανταγωνιστικές αγορές).
- Όταν δημιουργείται μάθηση ή προσαρμογή από το ίδιο το σύστημα (π.χ. έρευνα και ανάπτυξη, συνεργασία).
- Στις κοινωνικές επιστήμες: Όταν κοινωνικές δομές μπορούν ευκολότερα να περιγραφούν από σενάρια στα οποία κάθε μεταβολή μίας παραμέτρου διαφοροποιεί κάθε φορά το αποτέλεσμα, από ότι με διαφορεικές εξισώσεις (π.χ. οργανωτικές δομές).
- Όταν είναι σημαντικό να μοντελοποιηθούν οι διαδικασίες με τις οποίες οι πράκτορες σχηματίζουν οργανισμούς, και η προσαρμογή και η μάθηση είναι σημαντικές σε επίπεδο οργανισμού.
- Όταν μας ενδιαφέρει ο φυσικός χώρος και η χρονική διάσταση (π.χ. δίκτυα εφοδιαστικής αλυσίδας).
- Όταν ο πληθυσμός είναι ετερογενής ή η τοπολογία των αλληλεπιδράσεων είναι μεταβαλλόμενη και περίπλοκη (π.χ. κοινωνικά δίκτυα).

Τέλος, αξίζει να αναφερθεί ότι ένας από τους σημαντικότερους παράγοντες της δημοφιλίας των Agent-Based Modeling συστημάτων ιδιαίτερα στις κοινωνικές επιστήμες, εκτός από την ευκολία δημιουργίας απλών μοντέλων Agent-Based Modeling, είναι η δυνατότητα μοντελοποίησης σύνθετων λειτουργιών της ανθρώπινης κοινωνίας (agents στη θέση ανθρώπων αλληλεπιδρούν με άλλους agents-ανθρώπους και μέσα από τις επιμέρους αλληλεπιδράσεις-συμπεριφορές οδηγούμαστε σε ευρύτερα αποτελέσματα - προσομοίωση της κοινωνίας σε μακροσκοπικό επίπεδο).

Agent-Based Modeling συστήματα για απλές εφαρμογές μπορούν να δημιουργηθούν σε Excel ή Matlab με σχετικά μικρές προγραμματιστικές γνώσεις, ενώ υπάρχουν πολλαπλές Agent-Based Modeling πλατφόρμες ανοιχτού λογισμικού.

Σημαντικότερο όμως είναι το γεγονός, ότι τα πιο εξελιγμένα Agent-Based Modeling συστήματα μπορούν να χρησιμοποιηθούν σε πιο περίπλοκες εφαρμογές, καθώς μπορούν να ενσωματώσουν νευρωνικά δίκτυα, εξελικτικούς αλγορίθμους, γενετικούς αλγορίθμους και άλλες τεχνικές μάθησης που επιτρέπουν ρεαλιστική μάθηση και προσαρμογή, αλλά είναι φυσικά πιο σύνθετες υπολογιστικά [9].

2.5. Εφαρμογές

Στη συνέχεια θα περιγράψουμε μερικές χαρακτηριστικές, επίκαιρες ή καινοτόμες εφαρμογές του Agent-Based Modeling.

2.5.1. Agent-Based Μοντέλο της πανδημίας Covid-19 για την προσομοίωση των οικονομικών και επιδημιολογικών επιπτώσεων των μέτρων κοινωνικής αποστασιοποίησης [11]

Ο στόχος του μοντέλου είναι η προσομοίωση μία κλειστής κοινωνίας που ζει σε ένα κοινό πεπερασμένο περιβάλλον. Η κοινωνία αποτελείται από ανθρώπους, οι οποίοι είναι οργανωμένοι σε οικογένειες, επιχειρήσεις και κυβέρνηση, και αλληλεπιδρούν μεταξύ τους. Με αυτές τις κοινωνικές δομές η προσομοίωση προσπαθεί να καλύψει τα κύρια στοιχεία της κοινωνίας.

Η προσομοίωση εξερευνά επτά πιθανά σενάρια:

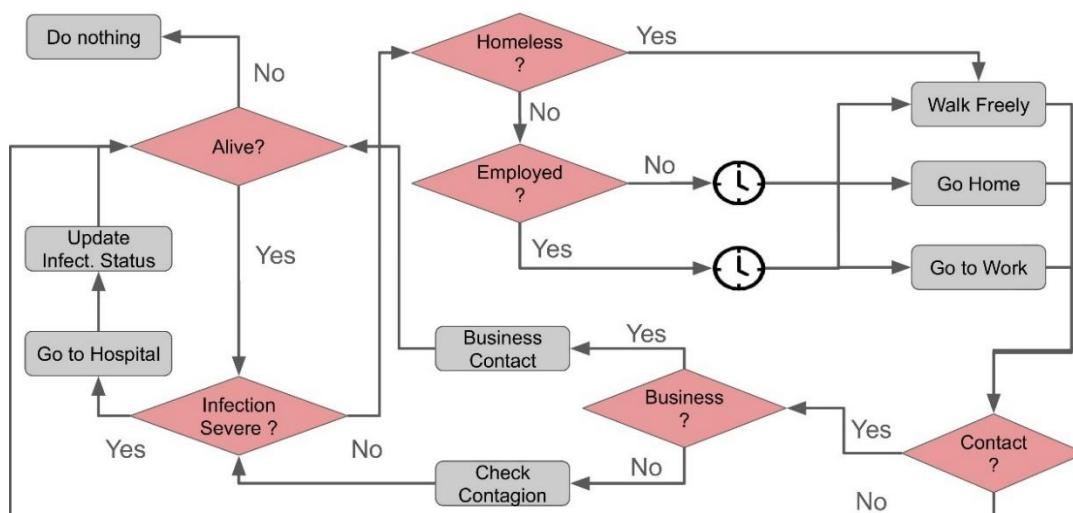
1. Κανένα μέτρο
2. Απαγόρευση κυκλοφορίας
3. Απαγόρευση κυκλοφορίας υπό όρους,
4. Κάθετη απομόνωση¹,
5. Μερική απομόνωση,
6. Χρήση μάσκας προσώπου και
7. Χρήση μάσκας προσώπου μαζί με το 50% της κοινωνικής απομόνωσης.

Χρησιμοποιούνται δεδομένα από τη Βραζιλία για όλα τα σενάρια που εξετάζονται, αλλά το προτεινόμενο μοντέλο που βασίζεται σε πράκτορες είναι πλήρως παραμετροποιήσιμο και μπορεί εύκολα να μεταφερθεί σε άλλα περιβάλλοντα δεδομένου ότι παρέχονται αντίστοιχα δεδομένα.

Το μοντέλο δημιουργεί ένα σύνολο από πέντε διαφορετικά είδη agent (person, houses, business, government και healthcare system) καθένα από τα οποία έχει τα δικά του χαρακτηριστικά και δυνατές ενέργειες.

Η προσομοίωση είναι μία επαναληπτική διαδικασία με κάθε επανάληψη να αφορά χρόνο μίας ώρας. Κατά τη διάρκεια αυτής, οι πράκτορες καλούνται να εκτελέσουν ενέργειες που εξαρτώνται από τον τύπο και τις συμπεριφορές τους (Εικόνα 2.4.). Το μοντέλο αρχικοποιείται με τις κατάλληλες ανά περίπτωση παραμέτρους ως είσοδο και παράγει μία απόκριση (το σύνολο των τιμών απόκρισης των παρατηρήσιμων μεταβλητών του) που σχετίζεται με τις επιδημιολογικές ή οικονομικές συνέπειες της πανδημίας.

¹ Κάθετη απομόνωση είναι η πολιτική κοινωνικής παρέμβασης όπου τα γνωστά μολυσμένα άτομα και οι γνωστές ομάδες κινδύνου - ηλικιωμένοι και άτομα με προϋπάρχουσες ασθένειες - διατηρούνται σε κοινωνική απομόνωση, ενώ οι νέοι και οι ενήλικες επιτρέπεται να εργάζονται τακτικά [11]

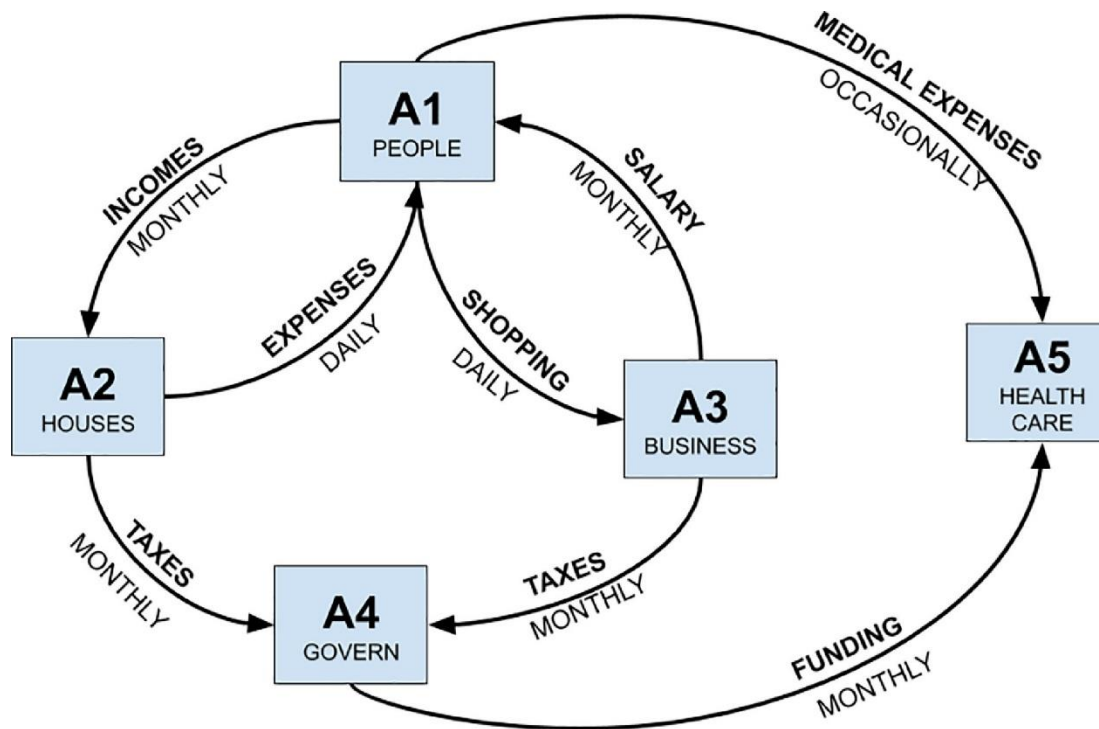


Εικόνα 2.4. Το διάγραμμα δραστηριοτήτων των People agents στην προσομοίωση [11]

Σε κάθε επανάληψη, ελέγχεται εάν υπήρχε επαφή μεταξύ οποιουδήποτε ζεύγους πρακτόρων. Μια επαφή συμβαίνει όταν η απόσταση μεταξύ οποιωνδήποτε δύο παραγόντων είναι μικρότερη ή ίση με ένα κατώφλι (θα δούμε αργότερα ότι έτσι χειριζόμαστε την απόσταση και στο δικό μας Agent-Based Modeling). Ανάλογα με τον τύπο ή τα χαρακτηριστικά των πρακτόρων η επαφή μπορεί να είναι οικονομική ή επιδημιολογική.

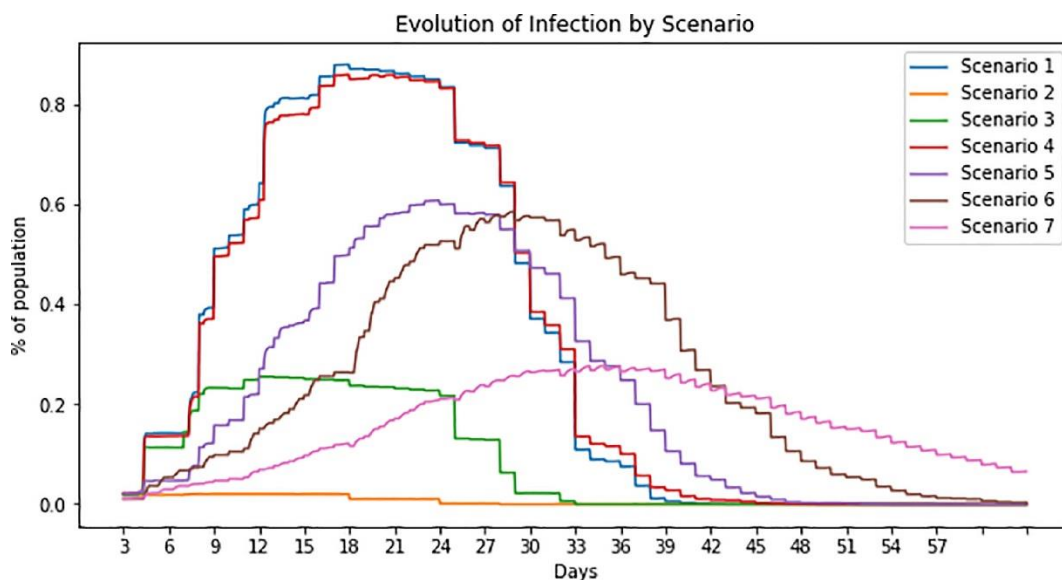
Η πιθανότητα μετάδοσης εξαρτάται από την αλληλεπίδραση των παραγόντων, την εγγύτητα και την επαφή. Ως εκ τούτου, όσο μεγαλύτερη είναι η κινητικότητα ενός ατόμου, τόσο μεγαλύτερη είναι η πιθανότητα να προσεγγίσει ένα μολυσμένο άτομο και να μολυνθεί. Μόλις μολυνθεί ένας agent, μπορεί να βρίσκεται σε μία από αυτές τις υποκαταστάσεις: α) ασυμπτωματική, η οποία περιλαμβάνει ήπια συμπτώματα χωρίς νοσηλεία, β) νοσηλεία και γ) σοβαρή, χρησιμοποιείται σε περιπτώσεις νοσηλείας σε μονάδα εντατικής θεραπείας (ΜΕΘ).

Ταυτόχρονα, για να προσομοιωθεί και ο αντίκτυπος στην οικονομία από τους διαφορετικούς τύπους περιορισμών κινητικότητας, ορίζονται και “business contacts”, οικονομικές συναλλαγές των agents οι οποίες εξαρτώνται από την κοινωνικοοικονομική τους θέση καθώς και από την κινητικότητα τους (Εικόνα 2.5.).

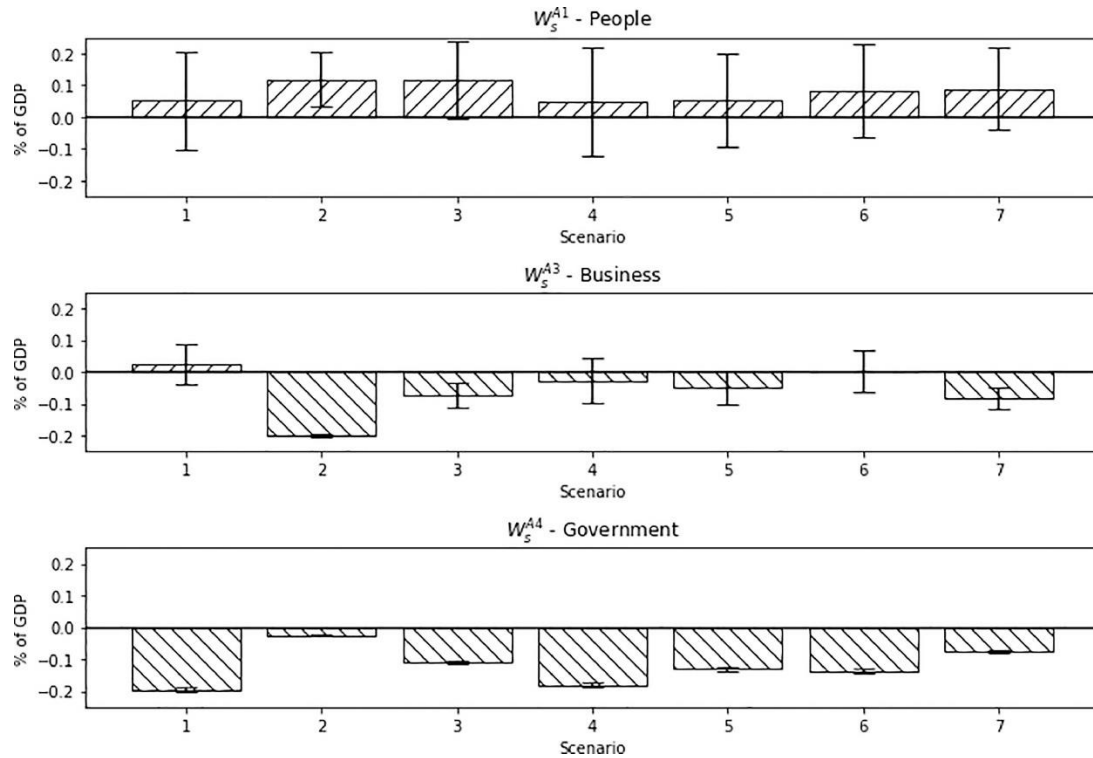


Εικόνα 2.5. Απεικόνιση των συναλλαγών μέσω των οποίων οι agents ανταλλάσσουν πλούτο στην προσομοίωση. Η οικονομική δυναμική ακολουθεί εποχιακές ρουτίνες που εξαρτώνται επίσης από τον τύπο του πράκτορα [11].

Παρουσιάζονται ενδεικτικά τα αποτελέσματα του μοντέλου για κάθε ένα από τα σενάρια που αριθμήσαμε παραπάνω (Εικόνα 2.6. και Εικόνα 2.7.).



Εικόνα 2.6. Απεικόνιση της επιδημιολογικής καμπύλης και οπτική σύγκριση της εξέλιξης της πανδημίας για κάθε ένα σενάριο [11].

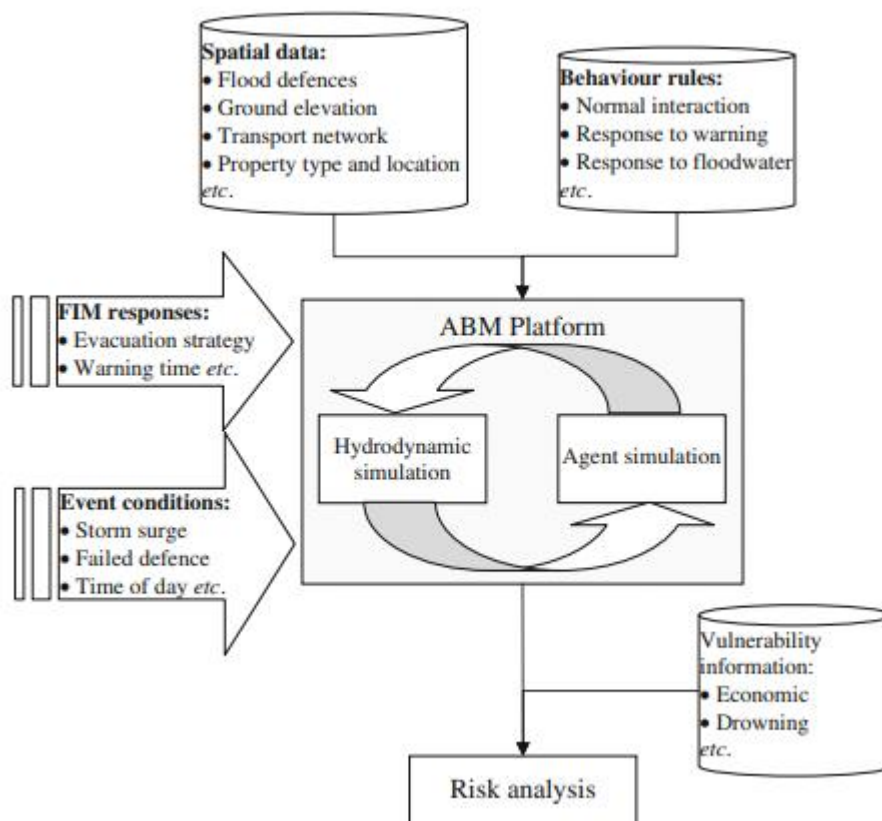


Εικόνα 2.7. Τα οικονομικά αποτελέσματα του κάθε σεναρίου συγκρινόμενα με τα αποτελέσματα χωρίς την πανδημία [11].

2.5.2. Agent-Based μοντέλο για τη διαχείριση περιστατικών πλημμύρας με ανάλυση ρίσκου [12].

Όπως αποδεικνύεται σε πολλά γεγονότα πλημμύρας, η κακή διαχείριση συμβάντων πλημμύρας (Flood Incident Management – FIM) μπορεί να προκαλέσει εκτεταμένη απώλεια ζωής και πολλές ακόμη σημαντικές ζημιές.

Ο στόχος του μοντέλου είναι η εκτίμηση της πιθανής έκθεσης των ανθρώπων σε πλημμύρες κατά τη διάρκεια ενός συμβάντος βροχόπτωσης και η δημιουργία μίας βασισμένης στην ανάλυση ρίσκου προσέγγισης για την αξιολόγηση των πιθανών μέτρων απόκρισης στην πλημμύρα, FIM.



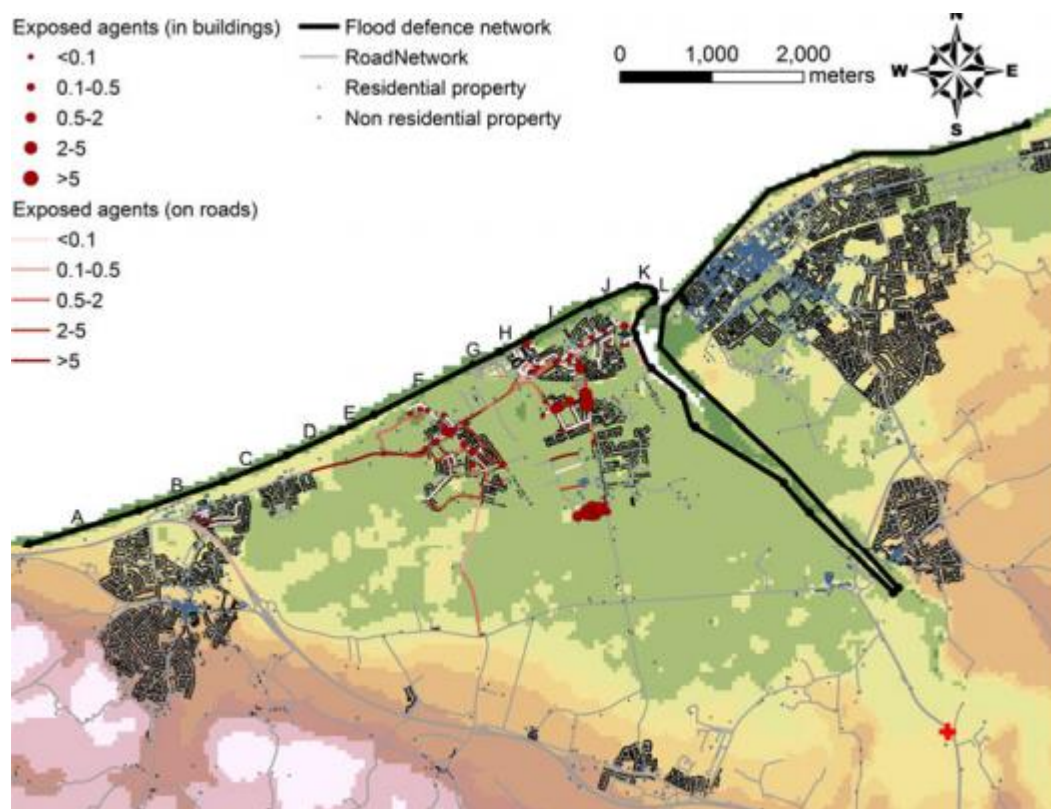
Εικόνα 2.8. Επισκόπηση του συστήματος ανάλυσης κινδύνου για τη διαχείριση περιστατικών πλημμυρών [12].

Στην προσομοίωση οι πρωταρχικοί agents είναι ο ανθρώπινος πληθυσμός της περιοχής. Επίσης με βάση το Agent-Based Modeling αναπτύχθηκε ένα υδροδυναμικό μοντέλο που προσομοιώνει τη ροή της πλημμύρας, το οποίο αλληλεπιδρά άμεσα με τους agents και το δομημένο περιβάλλον. Χωρικά δεδομένα που περιγράφουν χαρακτηριστικά όπως οι δομές προστασίας από τις πλημμύρες, ο τύπος ιδιοκτησίας και η θέση της, το ύψος των αναχωμάτων για ανάσχεση πλημμυρών, και το δίκτυο μεταφορών, εισάγονται στην πλατφόρμα Agent-Based Modeling από εθνικά διαθέσιμα σύνολα δεδομένων.

Η συμπεριφορά των Agents καθορίζεται από ένα σύνολο σαφώς ορισμένων κανόνων συμπεριφοράς. Υπάρχουν δύο παράμετροι για τον προσδιορισμό της συμπεριφοράς του κάθε agent:

1. Πώς συμπεριφέρονται οι agents υπό κανονικές συνθήκες, οι οποίες καθορίζουν την αρχική τους θέση στην αρχή ενός συμβάντος πλημμύρας.
2. Πώς οι agents ανταποκρίνονται σε ένα συμβάν πλημμύρας ή σε FIM μέτρα.

Για τους σκοπούς της προσομοίωσης κάθε πράκτορας περιγράφεται χρησιμοποιώντας μια μηχανή πιθανοτήτων πεπερασμένης κατάστασης που περιέχει τις πιθανές καταστάσεις του, το σύνολο των δράσεων που μπορεί να πραγματοποιήσει, και τις μεταβάσεις μεταξύ καταστάσεων.



Εικόνα 2.9. Απεικόνιση της προσομοίωσης να εκτελείται. Με κόκκινα στίγματα οι άνθρωποι – agents οι οποίοι έχουν εκτεθεί σε νερό βάθους μεγαλύτερου των 25cm. Το συγκεκριμένο σενάριο αφορά άνοδο του επιπέδου της θάλασσας κατά 6 μέτρα ως αποτέλεσμα ενός storm surge² [12].

2.6. Συστήματα Γεωγραφικών Πληροφοριών (GIS) και Agent-Based Modeling

Μία σύντομη αναφορά αξίζει να γίνει στη σχέση του Agent-Based Modeling και των Γεωγραφικών Συστημάτων Πληροφοριών (Geographical Information Systems – GIS).

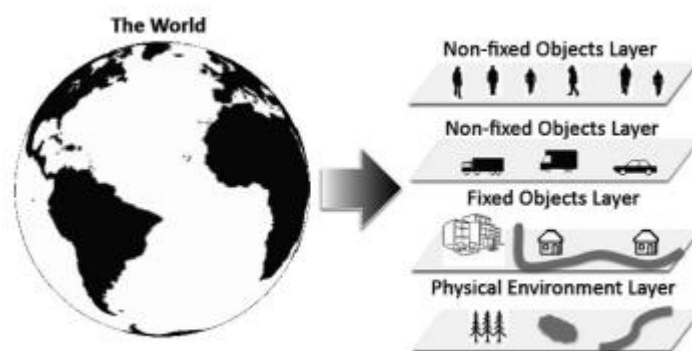
Το Agent-Based Modeling μπορεί να φανεί χρήσιμο σε πολλά γεωγραφικά προβλήματα, και υπάρχει αυξανόμενο ενδιαφέρον για την ενσωμάτωση των GIS και Agent-Based Modeling [13]. Για μοντέλα που βασίζονται σε agents, αυτή η ενσωμάτωση παρέχει τη δυνατότητα να διαθέτουμε agents που σχετίζονται με πραγματικές γεωγραφικές συντεταγμένες. Αυτό είναι ζωτικής σημασίας όσον αφορά τη μοντελοποίηση των γεωγραφικών συστημάτων, καθώς τα πάντα σε μια πόλη, περιοχή ή χώρα έχουν γεωγραφική αναφορά. Η γεωγραφική σύνδεση επιτρέπει στους δημιουργούς μοντέλων να σκεφτούν πώς τα αντικείμενα ή οι agents και οι

² Storm surge: Αύξηση του επιπέδου της θάλασσας λόγω αλλαγών του ανέμου και της ατμοσφαιρικής πίεσης, που σχετίζονται με καταιγίδα. Πηγή: https://www.lexico.com/definition/storm_surge

συγκεντρώσεις τους αλληλεπιδρούν και αλλάζουν στο χώρο και το χρόνο, ενώ παρέχει στους χρήστες GIS τη δυνατότητα μοντελοποίησης των αλληλεπιδράσεων εντός ενός GIS με την πάροδο του χρόνου και του χώρου.

Ο δεσμός αυτό είναι πολύ ελκυστικός διότι, ενώ το GIS μας παρέχει τη δυνατότητα να παρακολουθούμε τον κόσμο, δεν παρέχει κάποιο μηχανισμό για να ανακαλύψουμε νέα πλαίσια λήψης αποφάσεων, όπως για παράδειγμα γιατί οι άνθρωποι έχουν μετακινηθεί σε νέες περιοχές. Ο απλούστερος τρόπος οπτικοποίησης της ενσωμάτωσης των GIS και Agent-Based Modeling είναι με το να λάβουμε υπόψιν την οπτική των περισσότερων GIS για τον κόσμο (Εικόνα 2.10.)

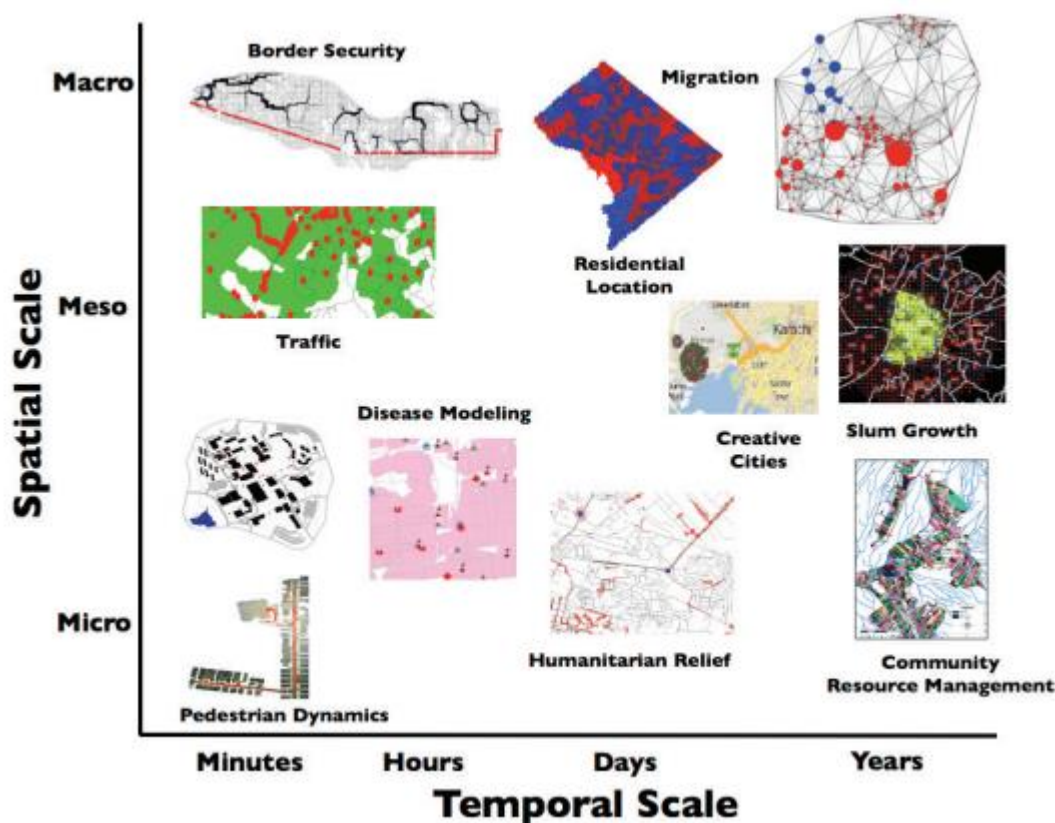
Σε ένα GIS ο κόσμος αναπαριστάται ως μία σειρά στρώματων (όπως το φυσικό περιβάλλον, το αστικό περιβάλλον κτλ.) και αντικειμένων διαφορετικών τύπων (όπως αυτοκίνητα ή άνθρωποι που μπορούν στη συνέχεια να χρησιμοποιηθούν ως βάση για τους agents μας). Αυτά τα στρώματα δημιουργούν τον τεχνητό κόσμο στον οποίο μπορούν να κατοικήσουν οι agents μας. Κάποια από τα στρώματα μπορούν να λειτουργήσουν ως όρια για τις προσομοιώσεις μας, ενώ κάποια σταθερά στρώματα μπορούν να παρέχουν μέσα για τους agents να μετακινούνται από το Α στο Β μέρος (όπως δρόμοι), ή κάποια μπορούν να παρέχουν μέρη για να ζήσουν (όπως σπίτια) [14].



Εικόνα 2.10. Αναπαράσταση του κόσμου ως στρώματα σταθερών και μη σταθερών αντικειμένων [14], [15].

Το εύρος των συνδυασμένων εφαρμογών Agent-Based Modeling και GIS είναι ενδεικτικό της χρησιμότητας της λειτουργικής διασύνδεσής τους. Στην Εικόνα 2.11. παρουσιάζονται ενδεικτικά κάποιες από τις πολυάριθμες εφαρμογές που συνδυάζουν Agent-Based Modeling και GIS.

Άλλωστε και η προσομοίωση που αναπτύχθηκε στην παρούσα διπλωματική εργασία, αποτελεί συνδυασμένη εφαρμογή των δύο συστημάτων.



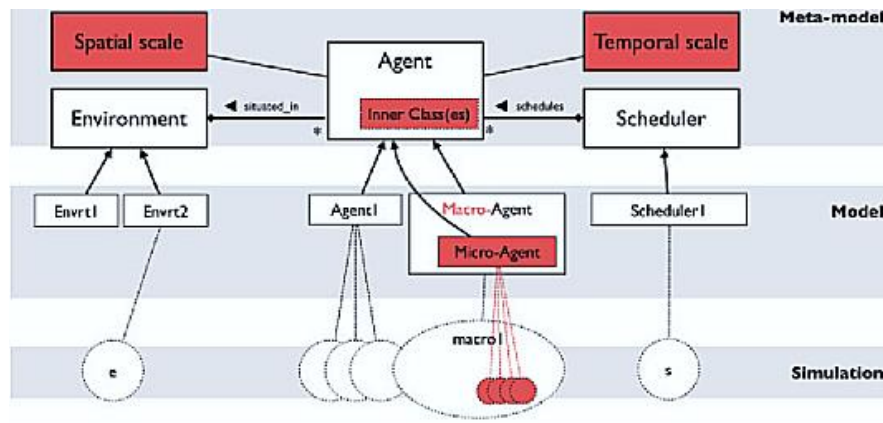
Εικόνα 2.11. Ένα δείγμα τομέων εφαρμογής για μοντέλα που βασίζονται σε χωρικούς και χρονικούς παράγοντες [14].

2.7. Multi-Level Agent-Based Modeling

Όπως μελετήσαμε στην προηγούμενη ενότητα, ο χώρος και ο χρόνος αποτελούν σημαντικές συνιστώσες ενός Agent-Based μοντέλου και βάση για πολλές από τις σύγχρονες εφαρμογές του Agent-Based Modeling.

Από την σύνδεση Agent-Based Modeling και GIS προέκυψε η ανάγκη για πιο γενικευμένη ενσωμάτωση των επιμέρους συνιστωσών αυτών. Οι απαιτήσεις αυτές οδήγησαν σε μοντέλα που συνδυάζουν τα επίπεδα αναπαράστασης. Τέτοια μοντέλα ονομάζονται μοντέλα Agent-Based Modeling πολλαπλών επιπέδων (Multi-Level Agent-Based Models). Τα Multi-Level Agent-Based Models ορίζουν agents που ανήκουν στο ίδιο μοντέλο αλλά σε διαφορετικά επίπεδα αναπαράστασης σε σχέση με το χρόνο, το χώρο και τη συμπεριφορά [16].

Λόγω του γενικευμένου ορισμού τους, τα Multi-Level Agent-Based Models συνδέονται με ένα ευρύ φάσμα εφαρμογών. Προσφέρουν ενοποιημένη αντιμετώπιση των επιμέρους επιπέδων, χωρίς απώλεια των ιδιαίτερων χαρακτηριστικών τους, και με ευελιξία στην αλληλεπίδραση των επιπέδων αυτών σε τέτοιο βαθμό ώστε να επιτρέπονται αναπροσαρμογές των ιδιοτήτων των agents σε σχέση με το επίπεδο στο οποίο ανήκουν.



Εικόνα 2.12. Επέκταση των παραδοσιακών εννοιών του Agent-Based Modeling στην κατεύθυνση του Multi-Level Agent-Based Modeling (οι νέες έννοιες με κόκκινο φόντο) [16]

Στο επόμενο κεφάλαιο, θα δούμε ότι τα Multi-Level Agent-Based Models αποτελούν βασικό κομμάτι της αρχιτεκτονικής της πλατφόρμας GAMA, πάνω στην οποία αναπτύχθηκε η προσομοίωση που παρουσιάζει η παρούσα διπλωματική εργασία.

Κεφάλαιο 3. Η πλατφόρμα Gama

3.1. Εισαγωγή στην Πλατφόρμα GAMA

Στο προηγούμενο κεφάλαιο αναλύθηκαν διεξοδικά η φιλοσοφία, η τεχνική και κάποιες σημαντικές πτυχές του Agent-Based Modeling. Στην τελευταία ενότητα, σταθήκαμε στη σχέση του Agent-Based Modeling με τα γεωγραφικά συστήματα πληροφοριών (GIS) και στην χρήση του Agent-Based Modeling σε συνδυασμό με χωρικές πληροφορίες. Στο κεφάλαιο αυτό θα επικεντρωθούμε σε μία από τις σημαντικότερες πλατφόρμες ανάπτυξης Agent-Based Modeling, η οποία δίνει την δυνατότητα ανάπτυξης και Agent-Based Modeling – GIS μοντέλων, την GAMA.

Θα ξεκινήσουμε με λίγες πληροφορίες για την πλατφόρμα GAMA, θα αναλύσουμε τους λόγους για τους οποίους επιλέχθηκε για την κατασκευή του μοντέλου της παρούσας διπλωματικής, και τέλος θα παρουσιάσουμε ένα σύντομο οδηγό για την λήψη και την λειτουργία της πλατφόρμας GAMA, έτσι ώστε φτάνοντας στο τέλος αυτής της διπλωματικής, ο αναγνώστης-ερευνητής, εκτός από το να κατανοεί τη λειτουργία της προσομοίωσης, να μπορεί να την χρησιμοποιεί, αλλάζοντας τις κατάλληλες παραμέτρους, στα δικά του πειράματα.

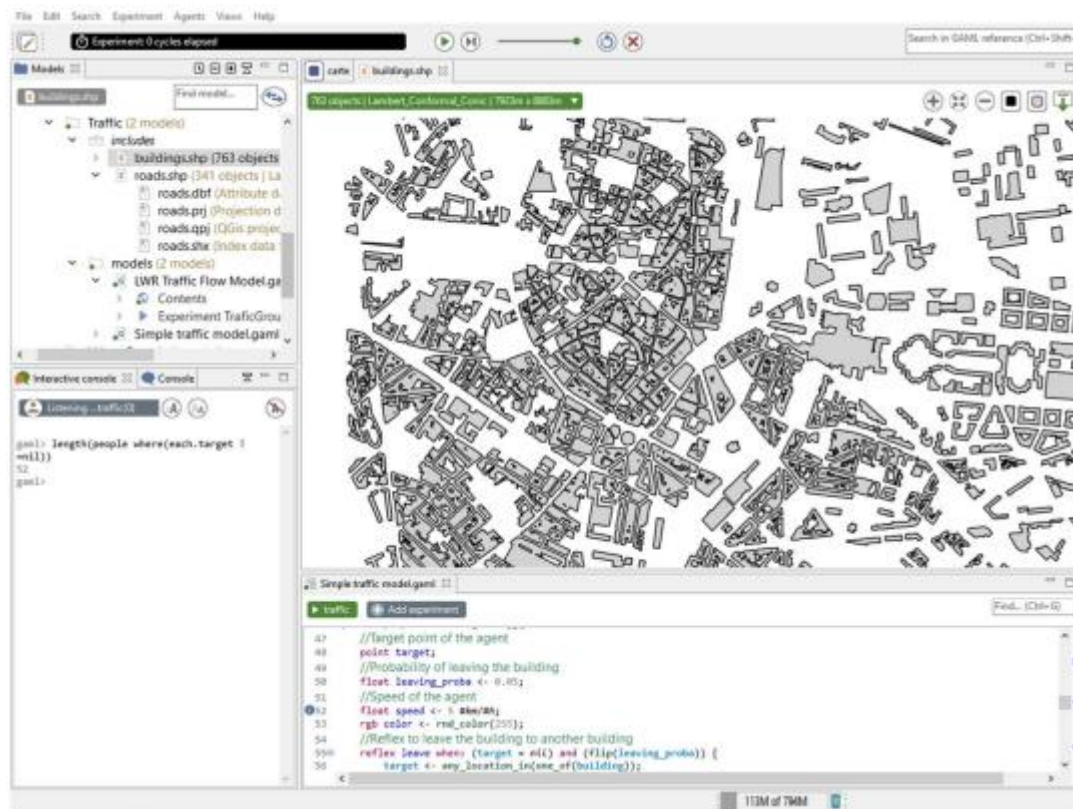
3.2. Η πλατφόρμα GAMA

Η πλατφόρμα GAMA (GIS & Agent-based Modeling Architecture) είναι ένα προηγμένο IDE³ για την ανάπτυξη προσομοιώσεων και μοντέλων με βάση το Agent-Based Modeling. Είναι λογισμικό ανοιχτού κώδικα (Open Source Software⁴) και χρησιμοποιείται συχνά για τη δημιουργία μοντέλων με πολλαπλούς τύπους agent, χωρικές και χρονικές συνιστώσες [17].

Η GAMA έχει χτιστεί έτσι ώστε να προσφέρει ευκολία στην χρήση και τον προγραμματισμό και το μικρότερο δυνατό «σκαλοπάτι» ξεκινήματος, χωρίς όμως να περιορίζει τις δυνατότητες μοντελοποίησης. Στην GAMA μπορούν να δημιουργηθούν τόσο απλά μοντέλα με περιορισμένους agents και παραμέτρους, όσο και πολυσύνθετες προσομοιώσεις που ενσωματώνουν πιθανοτικά μοντέλα, μηχανική μάθηση και σύνθετα δίκτυα.

³ IDE: Ένα ολοκληρωμένο περιβάλλον ανάπτυξης εφαρμογών, το οποίο παρέχει όλα τα απαραίτητα ανά περίπτωση εργαλεία για την διευκόλυνση του προγραμματιστή και της προγραμματιστικής διαδικασίας.

⁴ Open Source Software: Ένα λογισμικού ανοιχτού κώδικα είναι λογισμικό με πηγαίο κώδικα τον οποίο ο καθένας μπορεί να επιθεωρήσει, να τροποποιήσει και να βελτιώσει, εν αντιθέσει με το λογισμικό αποκλειστικής χρήσης ή λογισμικό κλειστού κώδικα (proprietary or closed source software), το οποίο μόνο το άτομο, η ομάδα ή ο οργανισμός που το δημιούργησε - και διατηρεί τον αποκλειστικό έλεγχο σε αυτό - μπορεί να τροποποιήσει. Πηγή: <https://opensource.com/resources/what-open-source>



Εικόνα 3.1. Το περιβάλλον μοντελοποίησης της GAMA [18]

Η GAMA βασίζεται:

1. Στην **Multi-Level modeling αρχιτεκτονική**. Μιλήσαμε στο τέλος της προηγούμενης ενότητας σύντομα για την χρησιμότητα του Multi-Level Modeling στο σύγχρονο Agent-Based Modeling.

Η προσέγγιση της GAMA στο Multi-Level Agent-Based Modeling βασίζεται σε τρεις αρχές:

- a. Ένας πράκτορας αντιστοιχεί σε ένα επίπεδο οργάνωσης το οποίο διαθέτει χωρική και χρονική αναπαράσταση.
- b. Τα επίπεδα οργανώνονται ιεραρχικά για να ορίσουν προνομιακές αλληλεπιδράσεις μεταξύ ενσωματωμένων επιπέδων.
- c. Οι οργανισμοί μπορούν να είναι δυναμικοί: ορισμένοι πράκτορες μπορούν να μετακινηθούν από έναν οργανισμό σε έναν άλλο για να προσαρμοστεί το επίπεδο αντιπροσώπευσης τους δυναμικά.

Το meta-model⁵ της GAMA είναι ορισμένο έτσι ώστε να διευκολύνει την πολύπλοκη αναπαράσταση του περιβάλλοντος και την δημιουργία μοντέλων πολλαπλών επιπέδων.

⁵ Meta-model: Ένα meta-model (ή, μεταμοντέλο σε ελληνική μετάφραση) είναι το μοντέλο μίας γλώσσας μοντελοποίησης. Το μεταμοντέλο ορίζει την δομή, τη σημασιολογία και τους περιορισμούς για μία οικογένεια μοντέλων [22].

2. **Στη γλώσσα μοντελοποίησης της, GAML (GAmA Modeling Language)** και τα στοιχεία που σχετίζονται με αυτή (parser και μεταγλωττιστής). Η GAML είναι μία πρακτοροκεντρική γλώσσα (agent-oriented language). Αυτό σημαίνει ότι οτιδήποτε «ενεργό» (οντότητες ενός μοντέλου, προσομοιώσεις, πειράματα κτλ.) αναπαρίσταται στην GAML ως agent και επομένως είναι ένα υπολογιστικό στοιχείο που διαθέτει τα δικά του δεδομένα και εκτελεί τη δική του συμπεριφορά, μόνο του ή σε αλληλεπίδραση με άλλους agents [18].
3. **Σε μία αποτελεσματική εικονική μηχανή** για την εκτέλεση του μοντέλου και την αλληλεπίδραση με την προσομοίωση. Συγκεκριμένα, η Gama χρησιμοποιεί JVM (Java Virtual Machine) αρχιτεκτονική, την συνηθέστερη προδιαγραφή παροχής περιβάλλοντος χρόνου εκτέλεσης στην οποία μπορεί να εκτελεστεί Java bytecode⁶.

3.3. Επιλογή της πλατφόρμας GAMA

Πριν τον προγραμματισμό της λύσης, κατά την ανάλυση του προβλήματος, τέθηκαν σαφείς προδιαγραφές για το μοντέλο προσομοίωσης που δημιουργήθηκε στα πλαίσια αυτής της διπλωματικής εργασίας.

Μερικές από τις βασικότερες προδιαγραφές ορίζουν ότι το μοντέλο πρέπει να είναι παραμετροποιήσιμο, να παρέχει μία οπτική αναπαράσταση των δεδομένων και των αλληλεπιδράσεων, να είναι εύκολο στην εγκατάσταση και τη χρήση ακόμη και από κάποιον χρήστη που δεν γνωρίζει προγραμματισμό, και να προσφέρει τη δυνατότητα για την εκτέλεση και κανονικών αλλά και συνεχόμενων πειραμάτων (όπως αυτά ορίζονται στο Κεφάλαιο 5) με διαφορετικές παραμέτρους, με την έξοδο πλήθους αποτελεσμάτων προς σύγκριση.

Ταυτόχρονα, κρίθηκε από την αρχή ως σημαντική η χωρική και γεωγραφική θεώρηση του προβλήματος. Συγκεκριμένα, η προσομοίωση θα πρέπει να μπορεί να χρησιμοποιηθεί για οποιοδήποτε μέρος του κόσμου για το οποίο υπάρχουν διαθέσιμα γεωγραφικά δεδομένα, και η οπτικοποίηση των αλληλεπιδράσεων που αναφέρθηκε να μπορεί να γίνεται σε πραγματικό χρόνο, με δυνατότητα παρακολούθησης της θέσης των agents ανά πάσα χρονική στιγμή.

Σημαντική επίσης κρίθηκε η open-source δυνατότητα χρήσης, ως ακαδημαϊκό εύχρηστο, επεκτάσιμο και δωρεάν εργαλείο διαθέσιμο για οποιονδήποτε ερευνητή και χρήστη.

⁶ Bytecode: Κώδικας προγράμματος που έχει μεταγλωττιστεί σε κώδικα χαμηλού επιπέδου κατάλληλα σχεδιασμένου για διερμηνέα λογισμικού (software interpreter). Μπορεί να εκτελεστεί από μία εικονική μηχανή ή να μεταγλωττιστεί περαιτέρω σε κώδικα μηχανής. Το Java bytecode συντάσσεται από πηγαίο κώδικα γραμμένο σε Java και εκτελείται από εικονική μηχανή Java, τη JVM. Πηγή: <https://techterms.com/definition/bytecode>

Τέλος, όπως αναλύθηκε στην προηγούμενη ενότητα, το Agent-Based Modeling εφαρμόζεται ολοένα και περισσότερο σε πραγματικές και θεωρητικές εφαρμογές. Ως αποτέλεσμα της ανερχόμενης αυτής ζήτησης, καινούργιες πλατφόρμες για την δημιουργία μοντέλων που βασίζονται στο Agent-Based Modeling εμφανίζονται με μεγάλη συχνότητα. Στη δική μας περίπτωση, οι προδιαγραφές της προσομοίωσής μας, η ανάγκη για σταθερότητα της εφαρμογής, η φιλοσοφία της GAMA ως open-source ακαδημαϊκής και συνεχώς εξελισσόμενης Multi-Level Agent-Based Modeling - GIS πλατφόρμας και το σύνολο των επιμέρους χαρακτηριστικών της που αναλύσαμε παραπάνω, κατέστησαν την επιλογή της ως τη βέλτιστη για την ανάπτυξη του μοντέλου.

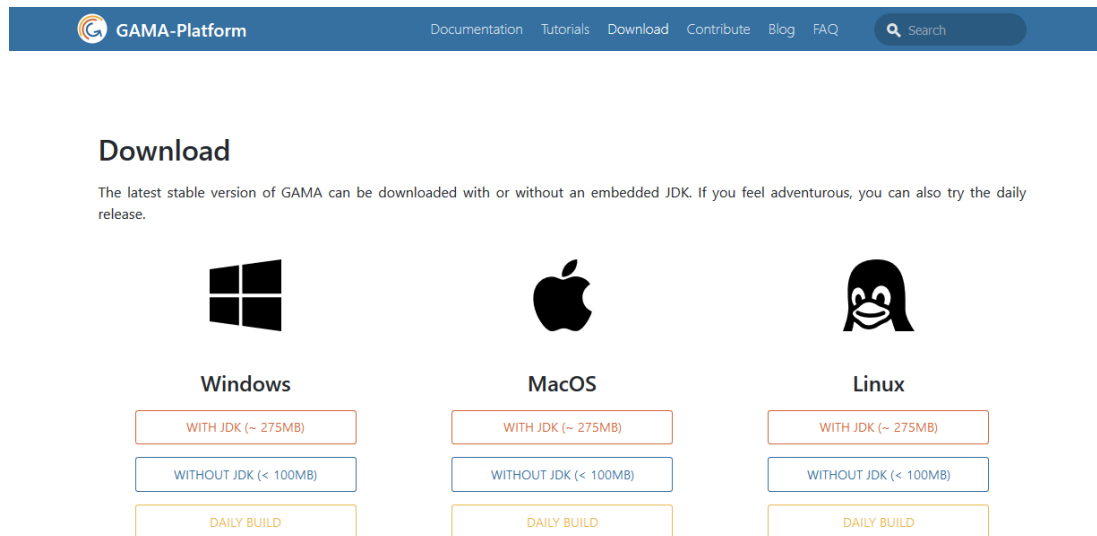
Αυτή την επιλογή ενίσχυσε σημαντικά και προηγούμενη διπλωματική εργασία που πραγματοποιήθηκε στο Εργαστήριο Συστημάτων και Αποφάσεων της Σχολής Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών, η οποία είχε ασχοληθεί διεξοδικά με την εύρεση της βέλτιστης πλατφόρμας υλοποίησης για ένα αντίστοιχο πρόβλημα, και είχε καταλήξει στην επιλογή της GAMA.

3.4. Εγκατάσταση της GAMA-Platform

Όπως προαναφέραμε, ένας από τους στόχους της παρούσας διπλωματικής είναι η εύκολη χρήση του μοντέλου που δημιουργείται. Προκειμένου να επιτευχθεί αυτός ο στόχος, λόγω του τρόπου που έχει αναπτυχθεί η GAMA, χρειάζεται πρώτα ο χρήστης να ολοκληρώσει τη διαδικασία εγκατάστασης για την οποία θα μιλήσουμε στις επόμενες παραγράφους.

Το πρώτο βήμα για την εγκατάσταση της GAMA είναι η μετάβαση στον ιστότοπο της πλατφόρμας, και συγκεκριμένα στην ενότητα [Download Gama](https://gama-platform.github.io/download)⁷ (Εικόνα 3.2).

⁷ <https://gama-platform.github.io/download>



Εικόνα 3.2. Σελίδα για τη μεταφόρτωση των αρχείων εγκατάστασης της GAMA. Η GAMA υποστηρίζει τις τρεις κυριότερες πλατφόρμες λογισμικού για προσωπικούς υπολογιστές. Πηγή: <https://gama-platform.github.io/download>

Αναλόγως με το λειτουργικό σύστημα του υπολογιστή που χρησιμοποιείται, Windows, MacOS ή Linux, επιλέγεται η αντίστοιχη στήλη.

Στη στήλη του κάθε λογισμικού, δίνονται **3 επιλογές** για μεταφόρτωση της πλατφόρμας:

1. Download with JDK (Java Development Kit)
2. Download without JDK
3. Daily Build

Οι επιλογές που θα αναλύσουμε θα είναι οι δύο πρώτες, καθώς το Daily Build αποτελεί την συνεχώς ανανεούμενη, πειραματική έκδοση της πλατφόρμας και όχι σταθερή έκδοση, είναι συνεπώς ακατάλληλη για την προσομοίωση μας.

3.4.1. Απαιτήσεις Συστήματος

Η έκδοση που περιλαμβάνει το JDK απαιτεί 540MB χώρου στο σκληρό δίσκο και η έκδοση χωρίς το JDK 250MB.

Τι είναι το Java Development Kit (JDK)

Το JDK αποτελεί ένα πακέτο εργαλείων για την ανάπτυξη Java-based λογισμικού. Περιλαμβάνει τον Java μεταγλωττιστή ο οποίος επιφορτίζεται με το έργο της μετατροπής του

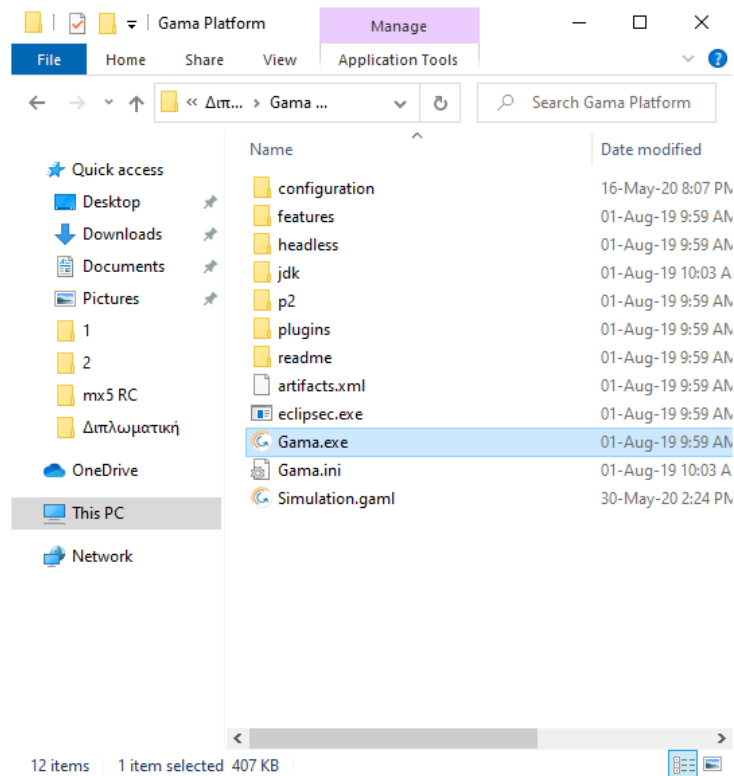
κειμένου-κώδικά μας σε γλώσσα μηχανής. Περισσότερες πληροφορίες για τον JDK και τα υπόλοιπα πακέτα της Java μπορείτε να βρείτε [εδώ](#)⁸.

1. Η έκδοση που περιλαμβάνει το JDK είναι ευκολότερη στην εγκατάσταση, καθώς για να εκτελεστεί απαιτεί μόνο την αποσυμπίεση του κατεβασμένου αρχείου (σημειώνεται ότι η έκδοση του JDK που περιλαμβάνεται είναι κατάλληλη μόνο για την χρήση με την Gama και όχι για άλλες εφαρμογές).
2. Η έκδοση που δεν περιλαμβάνει το JDK απαιτεί να είναι ήδη εγκατεστημένη στον υπολογιστή το Java 1.8 Oracle JDK (τουλάχιστον η ενημέρωση 161 της έκδοσης 1.8). Το πλεονέκτημα αυτής της έκδοσης είναι το μικρότερο μέγεθος και η ανεξάρτητη χρήση και ενημέρωση του JDK από τον χρήστη. Αξίζει να σημειωθεί ότι η Gama δεν θεωρείται συμβατή με τις εκδόσεις Java 1.9 και Java 1.10 του JDK καθώς δεν έχει δοκιμαστεί κάτω από αυτά τα περιβάλλοντα.

3.4.2. Διαδικασία Εγκατάστασης

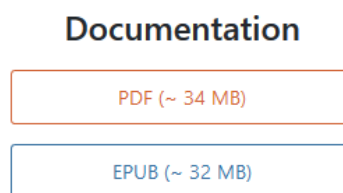
Μετά την επιλογή ενός εκ των δύο εκδόσεων (με ή χωρίς JDK) αποσυμπιέζουμε το .zip αρχείο που μεταφορτώθηκε και επιλέγεται ως προορισμός οποιοδήποτε φάκελος στον τοπικό δίσκο. Μέσα στα αρχεία που αποσυμπίστηκαν βρίσκουμε το εκτελέσιμο αρχείο (Gama.exe on Windows όπως φαίνεται στην Εικόνα 3.3. , Gama.app σε MacOS X, Gama on Linux) και επιλέγοντάς το εκκινούμε την πλατφόρμα Gama

⁸ <https://www.javaworld.com/article/3296360/what-is-the-jdk-introduction-to-the-java-development-kit.html>



Εικόνα 3.3. Τα περιεχόμενα του φακέλου εγκατάστασης της GAMA και το εκτελέσιμο αρχείο Gama.exe σε Windows 10.

Για οποιοδήποτε πρόβλημα προκύπτει κατά την εγκατάσταση της πλατφόρμας συστήνεται στον χρήστη να συμβουλευθεί το εκτενές documentation που υπάρχει στην ιστοσελίδα της πλατφόρμας κάτω από τους συνδέσμους μεταφόρτωσης, όπως φαίνεται στην Εικόνα 3.4.



Εικόνα 0.1

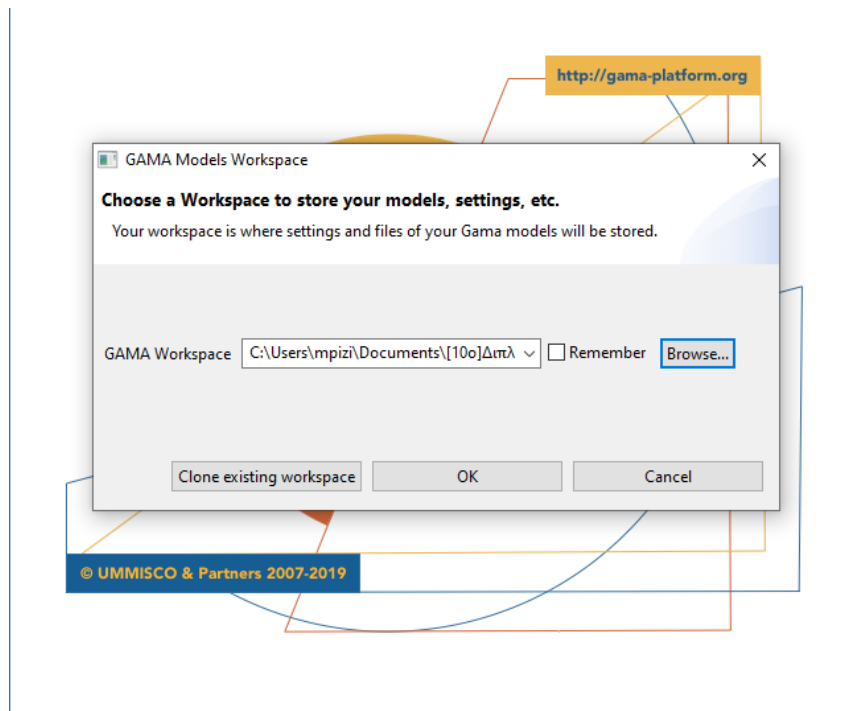
Εικόνα 3.4. Ο σύνδεσμος για την μεταφόρτωση του Documentation (τεκμηρίων λογισμικού στα ελληνικά) της πλατφόρμας GAMA.

Πηγή: <https://gama-platform.github.io/download>

3.4.3. Δημιουργία GAMA Models Workspace

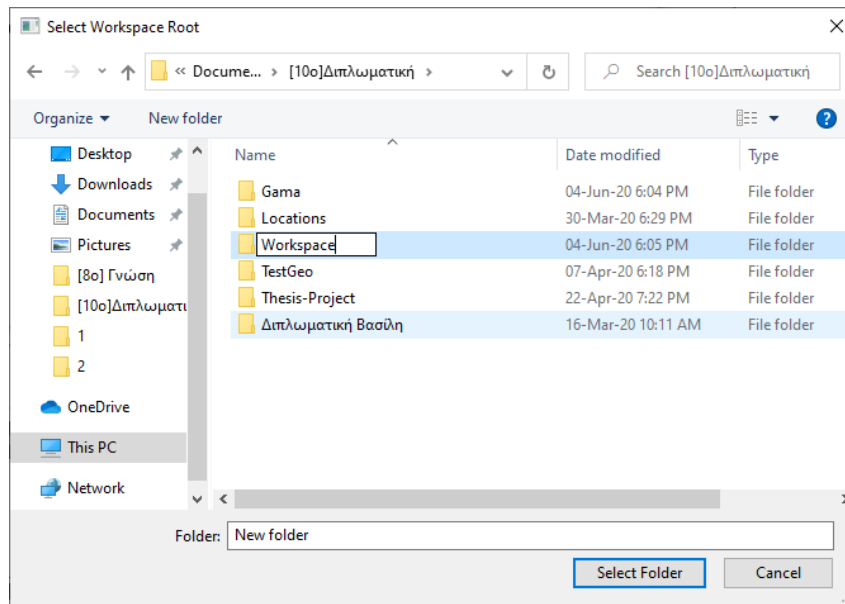
Μετά την εγκατάσταση της GAMA, και την εκκίνηση της πλατφόρμας από το εκτελέσιμο του φακέλου εγκατάστασης, μένουν λίγα βήματα για την ρύθμιση της πλατφόρμας και την λειτουργία της προσομοίωσης.

Με διπλό κλικ στο εκτελέσιμο αρχείο, η πλατφόρμα εκκινεί, και μετά την αρχική εικόνα φόρτωσης αναδύεται το παράθυρο Επιλογής GAMA Models Workspace (Εικόνα 3.5.).



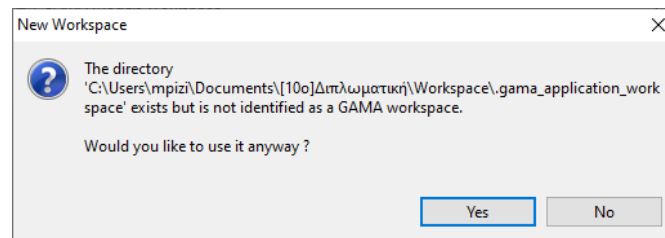
Εικόνα 3.5. Αναδυόμενο παράθυρο για επιλογή Workspace

Το Workspace είναι ο χώρος εργασίας στον οποίο αποθηκεύονται τα αρχεία, μοντέλα και οι ρυθμίσεις της πλατφόρμας GAMA. Μπορεί να είναι οποιοσδήποτε φάκελος. Εφόσον είναι η πρώτη φορά που χρησιμοποιείται η GAMA δίνεται η συμβουλή να δημιουργηθεί ένα νέος φάκελος που θα μετατραπεί σε GAMA Workspace, όπως φαίνεται στην Εικόνα 3.6.



Εικόνα 3.6. Δημιουργία Νέου Φακέλου, μετονομασία του σε Workspace και επιλογή του ως GAMA Workspace.

Μόλις επιλεγεί ο νέος φάκελος, η πλατφόρμα θα βγάλει ειδοποίηση ότι ο φάκελος που επιλέχθηκε υπάρχει, αλλά δεν αποτελεί GAMA Models Workspace, ρωτώντας εάν επιθυμούμε να τον μετατρέψουμε σε Workspace (Εικόνα 3.7.). Επιλέγουμε «Ναι».



Εικόνα 3.7. Μετατροπή του φακέλου που επιλέξαμε σε GAMA Models Workspace.

Μετά από αυτό το τελικό βήμα, η πλατφόρμα GAMA ανοίγει στην σελίδα υποδοχής, η εγκατάστασή της έχει ολοκληρωθεί επιτυχώς, και η πλατφόρμα είναι σε πλήρη λειτουργία. Στην επόμενη ενότητα, εκτός από την περιγραφή του στόχου, των σταδίων και του κώδικα της τελικής προσομοίωσης, θα περιγράψουμε όλα τα βήματα για την εκτέλεση της προσομοίωσης στην εγκατεστημένη πλατφόρμα GAMA τοπικά στον υπολογιστή μας.

3.5. Μειονεκτήματα και μελλοντική λειτουργία της GAMA

Όπως είδαμε στις τελευταίες παραγράφους η διαδικασία εγκατάστασης της GAMA, χωρίς να είναι ιδιαίτερα πολύπλοκη, αποτελείται από έναν όχι αμελητέο αριθμό βημάτων και διαδικασιών.

Ακόμη και για την εκτέλεση ενός απλού μοντέλου Agent-Based Modeling στην GAMA, ένας χρήστης θα πρέπει να κατεβάσει μια ολοκληρωμένη έκδοση της GAMA, προσαρμοσμένη στο περιβάλλον εργασίας του, να την εγκαταστήσει παράλληλα με μία λειτουργική και συμβατή εικονική μηχανή Java, ακολουθώντας όλα τα προηγούμενα βήματα. Στη συνέχεια θα πρέπει να μεριμνήσει για την συνεχή ενημέρωση των παραπάνω, μεταφορτώνοντας κάθε φορά τις καινούργιες ή ενημερωμένες εκδόσεις τόσο της GAMA όσο και της εικονικής μηχανής Java. Είναι φανερό ότι ακόμη και για έμπειρους χρήστες, αυτό μπορεί να αποδειχθεί πηγή δυσκολίας.

Πρόσθετη πολυπλοκότητα δημιουργεί και η δυνατότητα της GAMA να επεκτείνεται μέσω ανεπτυγμένων από τρίτους πρόσθετων εργαλείων (plug-ins) τα οποία είναι ανεξάρτητα από τον κώδικα της βασικής πλατφόρμας, και τα οποία να μην ενισχύουν ουσιωδώς τον εκφραστικό πλούτο της πλατφόρμας και των δυνατοτήτων της, αλλά απαιτούν πρόσθετα βήματα εγκατάστασης και μπορούν να αποτελέσουν τροχοπέδη ως προς την συνεχή ενημέρωση για αρχαίους χρήστες και μη [18].

Τέλος, ανεξάρτητα από τη διαδικασία εγκατάστασης ή/και ενημέρωσης, η λειτουργία της GAMA, ειδικά όταν πρόκειται για την εκτέλεση πολυσύνθετων μοντέλων είναι απαιτητική σε πόρους.

Προκειμένου να επιλυθούν τα παραπάνω προβλήματα, αναπτύσσεται μία νέα, διαδικτυακή έκδοση της GAMA. Με την έκδοση αυτή, όσοι επιθυμούν, θα μπορούν να γράφουν και να προσομοιώνουν μοντέλα σε ένα πρόγραμμα περιήγησης ιστού, χωρίς να χρειάζεται να εγκαταστήσουν στον υπολογιστή τους την πλήρη έκδοση της GAMA. Αυτή η εξέλιξη, όταν ολοκληρωθεί, θα διευκολύνει σημαντικά την κοινή χρήση, τη δημιουργία αλλά και τον πειραματισμό με Agent-Based μοντέλα, αυξάνοντας την ευχρηστία της πλατφόρμας, καθιστώντας τη χρήση της πιο προσιτή για έναν αρκετά μεγαλύτερο αριθμό νέων χρηστών, και δημιουργώντας μια ενδεχόμενη εκθετική ανάπτυξη στο χώρο των προσομοιώσεων και των μοντέλων βασισμένων σε agents [19][20].

Κεφάλαιο 4. Ανάλυση του Μοντέλου Προσομοίωσης

4.1. Ορισμός του Προβλήματος

Ο στόχος της παρούσας διπλωματικής εργασίας συνοψίζεται στη μοντελοποίηση και προσομοίωση της κίνησης των ανθρώπων σε αστικό περιβάλλον με χρήση Agent-Based Modeling. Ο στόχος αυτός θα επιτευχθεί με την ανάπτυξη της προσομοίωσης χρησιμοποιώντας την πλατφόρμα GAMA.

Όπως αναφέρεται στο Κεφάλαιο 1, επιδίωξή μας είναι η παρούσα εργασία να αποτελέσει βοήθημα και σημείο αφετηρίας για την δημιουργία μοντέλων επίλυσης προβλημάτων ανθρώπινης κινητικότητας με τη χρήση Agent-Based Modeling. Με αυτό το στόχο επιλέξαμε ένα συγκεκριμένο τύπο προβλήματος γύρω από τον οποίο μπορεί να αναπτυχθεί τόσο ο οδηγός, όσο και το ίδιο το μοντέλο, ενώ ταυτόχρονα να επιλυθεί και ένα πραγματικό πρόβλημα.

Πριν προχωρήσουμε, όμως, στις προδιαγραφές της τελικής εφαρμογής και στις λεπτομέρειες της ανάπτυξης, κρίνεται σκόπιμο να αναλυθεί συγκεκριμένα το πρόβλημα που επιχειρεί να λύσει η προσομοίωσή μας, καθώς και η προσέγγιση επίλυσης αυτού.

Πιο συγκεκριμένα το πρόβλημα με το οποίο θα ασχοληθούμε στην προσομοίωσή μας, είναι αυτό της εύρεσης ενός χαμένου παιδιού σε ένα ρεαλιστικό περιβάλλον πόλης.

Το σενάριο του προβλήματος έχει ως εξής:

1. Υπάρχει ένας ορισμένος γεωγραφικός χώρος. Ο γεωγραφικός αυτός χώρος μοντελοποιείται μέσω κτηρίων (building agents), δρόμων (road agents) και ενός συνόλου γεωγραφικών πληροφοριών οι οποίες εντάσσονται στον γράφο κίνησης⁹ της προσομοίωσης. Επάνω στον γεωγραφικό χώρο (ο οποίος όπως θα δούμε στη συνέχεια είναι προσαρμόσιμος και εισάγεται από τον χρήστη σύμφωνα με την πραγματική περιοχή στην οποία χάθηκε το παιδί) ζουν και εργάζονται οι άνθρωποι της περιοχής αυτής, ενώ ταυτόχρονα, επάνω στον χάρτη του γεωγραφικού χώρου μετακινείται το παιδί που αγνοείται.
2. Οι άνθρωποι της προσομοίωσης ζουν την καθημερινότητά τους στον γεωγραφικό χώρο που ορίσαμε, ταξιδεύουν μέσω του οδικού συστήματος, πεζοδρομίων και δρόμων και μετακινούνται από τις κατοικίες τους στον χώρο στον οποίο εργάζονται. Οι άνθρωποι αυτοί μοντελοποιούνται ως **People agents**.
3. Ταυτόχρονα, γνωρίζουμε την αρχική θέση καθώς και την ώρα κατά την οποία χάθηκε το αγνοούμενο παιδί. Γνωρίζουμε, επίσης, κάποια σημεία ενδιαφέροντος μέσα στην πόλη, παιδικές χαρές, σχολεία, μαγαζιά με παιχνίδια, στα οποία το παιδί έχει περισσότερες πιθανότητες να επισκεφθεί και να παραμείνει για κάποιο χρονικό διάστημα. Το παιδί θα μοντελοποιηθεί ως ο agent **missing person**.

⁹ Ο γράφος κίνησης της προσομοίωσης είναι ένας γράφος που δημιουργείται από τα γεωγραφικά στοιχεία τα οποία εισάγουμε στο σύστημα και αποτελεί μία κωδικοποιημένη αναπαράσταση της πραγματικής γεωγραφίας, των δρόμων και της κατεύθυνσής τους.

Για την επίλυση του προβλήματος θα δημιουργήσουμε μία προσομοίωση μέσω της οποίας ο ερευνητής-χρήστης θα μπορεί να πειραματιστεί κατά τη διαδικασία επίλυσης και να ελέγχει τα συμπεράσματα και αποτελέσματά του με εύχρηστο, αναλυτικό οπτικά και προσαρμόσιμο τρόπο, με στόχο τη βελτιστοποίηση της διαδικασίας εύρεσης του χαμένου παιδιού.

4.2. Προδιαγραφές και Επιλογές Υλοποίησης

Έχοντας περιγράψει το πρόβλημα το οποίο καλούμαστε να επιλύσουμε, δημιουργούμε τις προδιαγραφές που θα έχει η υλοποίησή της προσομοίωσης για την επίλυσή του συγκεκριμένου προβλήματος. Οι προδιαγραφές αυτές θα ορίσουν το πλαίσιο δημιουργίας του κώδικα και θα καθορίσουν τη λειτουργία της προσομοίωσης.

Μην έχοντας, όμως, τη δυνατότητα προσομοίωσης κάθε λειτουργίας ή λεπτομέρειας της πραγματικής ζωής, ή στην δική μας περίπτωση της ανθρώπινης κίνησης, κάθε διαδικασία μοντελοποίησης κρίνεται στην επιτυχία της πρώτα από τους σωστούς συμβιβασμούς περιορισμών που ο κάθε μοντελιστής θα κριθεί να κάνει.

Σημειώνεται ότι κάθε εκτέλεση της προσομοίωσης θα αναφέρεται στην εργασία ως πείραμα.

Στην ενότητα αυτή θα αναλυθούν οι αρχικές προδιαγραφές λειτουργίας καθώς και οι αποφάσεις περιορισμών που επιλέχθηκαν ανά κατηγορία. Θα αριθμήσουμε τις επιμέρους κατηγορίες και αποφάσεις έτσι ώστε να μπορούμε να αναφερθούμε σε αυτές στις επόμενες ενότητες.

Συγκεκριμένα για να γίνει κατανοητή η **αντιστοίχιση** μεταξύ των οριζόμενων προδιαγραφών και της υλοποίησής τους σε κώδικα, σημειώνεται ότι οι **μεν προδιαγραφές** ορίζονται παρακάτω με **αριθμητική σήμανση 1.a., 1.b., 2.a** κτλ. και η **υλοποίηση σε κώδικα** που περιγράφεται στο **Κεφάλαιο 4.4.** έχει αντιστοιχη αριθμητική σήμανση, **4.4.1.a., 4.4.1.b., 4.4.2.a.** κτλ.

1. Χάρτης:

- a. Ο χάρτης της προσομοίωσης πρέπει να ανταποκρίνεται σε πραγματικά δεδομένα. Στο Κεφάλαιο 4.3. θα περιγραφούν εύχρηστοι και δωρεάν τρόποι να εξαχθούν δεδομένα πραγματικών χαρτών από το χρήστη και να εισαχθούν στην προσομοίωση.
- b. Ο χάρτης της προσομοίωσης πρέπει να μπορεί να αλλάζει εύκολα από τον χρήστη.

2. People Agents:

- a. Ο αριθμός των People Agents είναι προσαρμόσιμος. Μπορεί να αλλάξει από το χρήστη για κάθε πείραμα σε αριθμό αντιπροσωπευτικό του εκάστοτε χάρτη, ή των φαινομένων που θέλει να παρατηρήσει.
- b. Ο κάθε People Agent έχει ένα κτήριο ως σπίτι και ένα κτήριο ως εργασιακό χώρο. Ο agent διαθέτει ένα συγκεκριμένο χώρο ή διαμέρισμα στα κτήρια αυτά. Για τους σκοπούς της προσομοίωσης και τα δύο κτήρια βρίσκονται εντός του χάρτη και κατανέμονται τυχαία.

- c. Οι μετακινήσεις των People Agents λαμβάνουν χώρα συγκεκριμένες ώρες της μέρας, ξεχωριστές για τον καθένα αλλά στο πλαίσιο ενός κανονικού ωραρίου εργασίας (προσαρμόσιμο κατά τη λειτουργία πειράματος). Μετακινούνται είτε με τα πόδια, είτε χρησιμοποιώντας μέσα μεταφοράς, κάτι που προσομοιώνεται προσδίδοντας διαφορετικές ταχύτητες στην κίνηση τους.
- d. Οι People Agents κατά τη μετακίνησή τους έχουν μία αφετηρία και έναν προορισμό. Όπως και στην πραγματικότητα, η κίνηση τους για να φτάσουν στον προορισμό τους ακολουθεί το οδικό σύστημα (που προσομοιώνεται μέσω του γράφου κίνησης) και η διαδρομή που ακολουθούν είναι η βέλτιστη δυνατή.

3. Missing Person:

- a. Κάνουμε την παραδοχή ότι ο missing person μετακινείται με τα πόδια, με προσαρμόσιμη από το χρήστη ταχύτητα ανάλογα με την ηλικία του παιδιού (ένα μεγαλύτερο σε ηλικία παιδί κινείται γρηγορότερα κατά κανόνα σε αστικό περιβάλλον από ένα μικρότερο).
- b. Πρέπει να μπορούμε να ορίσουμε την αρχική θέση (starting position) στην οποία βρισκόταν το παιδί, καθώς και την αρχική ημερομηνία/ώρα (start time) μετά από την οποία ξεκίνησε να αγνοείται. Αυτές οι πληροφορίες εισάγονται στην προσομοίωση από τον χρήστη και μπορούν να τροποποιούνται για κάθε πείραμα. Ταυτόχρονα, θα μπορούν να οριστούν πειράματα με τυχαίες τοποθεσίες εκκίνησης και χωρίς αρχική ημερομηνία και ώρα εξαφάνισης.
- c. Ο missing person δεν μετακινείται τυχαία στους δρόμους του χάρτη, αντίθετα κινείται πάντα προς κάποιον προορισμό.

Ταυτόχρονα, όπως αναφέραμε και στο 4.1., στο σύστημα πρέπει να μπορούν να εισαχθούν (εφόσον υπάρχουν και είναι διαθέσιμα) πιθανά σημεία ενδιαφέροντος για το αγνοούμενο παιδί (possible Points of Interest). Όπως αναφέρθηκε τέτοια σημεία είναι παιδικές χαρές, σχολεία, μαγαζιά με παιχνίδια κτλ.

Συνεπώς, αναλόγως με το εάν έχουν εισαχθεί σημεία ενδιαφέροντος από τον χρήστη, ο προορισμός (target) του missing person θα είναι είτε κάποιο τυχαίο κτήριο στο χάρτη, είτε κάποιο από τα σημεία ενδιαφέροντος. Αυτό θα προσομοιωθεί με τη χρήση μίας πιθανότητας, η τιμή της οποίας θα μπορεί να τροποποιηθεί από τον χρήστη για κάθε πείραμα. Στην υλοποίηση μας θα ενσωματωθεί η δυνατότητα για εισαγωγή ενός σημείου ενδιαφέροντος, για απλότητα, αλλά εύκολα μπορεί να επεκταθεί για την εισαγωγή μίας λίστας σημείων.

- d. Κάνουμε την παραδοχή ότι το παιδί θα αναμένει-ξεκουράζεται για ένα τυχαίο διάστημα κάποιων ωρών σε ένα κτήριο-καταφύγιο, το οποίο θα αλλάζει κάθε φορά σύμφωνα με την εύρεση προορισμού που αναλύθηκε. Το διάστημα αυτό θα είναι παραμετροποιήσιμο από το χρήστη.

4. Πιθανοτικό Μοντέλο Εύρεσης:

- a. Όπως συμβαίνει και στην πραγματική ζωή, κάποιοι από τους ανθρώπους της εκάστοτε περιοχής έχουν ειδοποιηθεί μέσω σχετικών

ενημερώσεων, μέσω κάποιας εφαρμογής για την εύρεση χαμένων παιδιών, ή με οποιοδήποτε άλλο μέσο, για την εξαφάνιση του παιδιού, και ως εκ τούτου κατά την καθημερινή μετακίνησή τους βρίσκονται σε επιφυλακή για την εύρεση του. Η επιφυλακή αυτή ενός ποσοστού του πληθυσμού προσομοιώνεται με ένα πιθανοτικό μοντέλο εύρεσης κατά το οποίο, όταν το παιδί προσεγγίσει κατά την κίνησή του κάποιον εκ των People Agents, ο agent θα συνειδητοποιήσει την ιδιότητα του παιδιού ως αγνοούμενου μόνο σε έναν αριθμό περιπτώσεων, τον αριθμό εκείνο που ορίζει η αντίστοιχη πιθανότητα.

- b. Οι πιθανότητες ένας εκ των People Agents να συνειδητοποιήσει ότι συνάντησε το χαμένο παιδί, missing person, εξαρτώνται επίσης από τρία σενάρια:
 - Ο People Agent μετακινείται πεζός.
 - Ο People Agent μετακινείται με κάποιο όχημα
 - Ο People Agent βρίσκεται στο σπίτι του κοιτώντας έξω (στα περισσότερα πειράματα μικρότερες πιθανότητες για αυτό το σενάριο).
- c. Οι πιθανότητες κάθε ενός εκ των ανωτέρω σεναρίων είναι προσαρμόσιμες από το χρήστη για κάθε πείραμα.

5. Στοιχεία κίνησης μέσα στην πόλη:

Όπως αναφέραμε ήδη, οι People agents μετακινούνται είτε με τα πόδια είτε με μεταφορικά μέσα. Η προσομοίωση θα υποστηρίζει δύο λειτουργίες σε σχέση με την κίνηση των People Agents στον χάρτη.

- a. Η πρώτη και προεπιλεγμένη είναι η κίνηση με τα πόδια (walking) σε περίπτωση που η απόσταση που πρέπει να καλυφθεί από τον agent είναι μικρότερη του ενός (1) χλμ και η χρήση μηχανοκίνητων μέσων στην περίπτωση που η απόσταση είναι μεγαλύτερη (driving). Η λειτουργία αυτή προσομοιώνει μία πραγματική κατάσταση στην οποία οι άνθρωποι μετακινούνται πεζοί για μικρές αποστάσεις.
- b. Η δεύτερη λειτουργία επιτρέπει στον χρήστη να τρέξει πειράματα δίνοντας ποσοστά των People Agents που μετακινούνται σε λειτουργία “walking” και εκείνων σε λειτουργία “driving” (π.χ. 30% “walkers” στον χάρτη και συνεπώς 70% “drivers”). Στην λειτουργία αυτή τα ποσοστά κατανέμονται τυχαία στους agents ανεξαρτήτως απόστασης προορισμού υπό τον όρο να ανταποκρίνονται συνολικά με τα ποσοστά που έχουν δοθεί ως είσοδος του χρήστη (στο παράδειγμά μας, 30% “walkers” και 70% “drivers”).

6. Λειτουργίες εκτέλεσης και πειράματα:

Η προσομοίωση θα υποστηρίζει δύο λειτουργίες εκτέλεσης:

- a. Την κανονική λειτουργία προσομοίωσης κατά την οποία βλέπουμε σε πραγματικό χρόνο την κίνηση των People Agents και του missing person.

Στην κανονική λειτουργία ο χρήστης έχει εποπτεία του πειράματος. Βλέπει, τόσο, το χάρτη, όσο και τους ορατούς agents πάνω σε αυτόν (building agents, road agents, People Agents και missing person).

Βλέπει την κίνησή τους πάνω στον χάρτη και την εξέλιξη της προσομοίωσης σε σχέση με την πάροδο του χρόνου.

Μπορεί επίσης να ελέγχει την ροή, να σταματά την προσομοίωση, ή να αλλάζει την ταχύτητα με την οποία περνάει ο χρόνος στην προσομοίωση, να προσαρμόζει τις παραμέτρους πριν την έναρξή του κάθε πειράματος και να βλέπει τις αλλαγές που συντελούνται στον χάρτη της προσομοίωσης.

Στην κανονική λειτουργία ο χρήστης εκτελεί ένα πείραμα κάθε φορά.

- b. Την λειτουργία πολλαπλών συνεχόμενων πειραμάτων (batch experiments).

Τα batch πειράματα αποτελούν συνεχόμενα διαδοχικά πειράματα στον ίδιο χάρτη τα οποία εκτελούμε όταν επιθυμούμε να μάθουμε περισσότερα για το πώς μία παράμετρος επηρεάζει τα αποτελέσματά μας.

Ένα batch πείραμα εκτελείται για πολλαπλές, ορισμένες από τον χρήστη, επαναλήψεις. Σε κάθε μία επανάληψη διατηρούμε όλες τις παραμέτρους εκτός από μία, σταθερές, και αλλάζουμε την τιμή εκείνης που θέλουμε να εξετάσουμε, έτσι ώστε να εξερευνήσουμε τα αποτελέσματα για διαφορετικές τιμές της.

Τα batch πειράματα εκτελούν κάθε επανάληψή τους, μέχρι την ικανοποίηση, μίας, ορισμένης από τον χρήστη, συνθήκης (π.χ. να βρεθεί το παιδί, να περάσει συγκεκριμένο χρονικό διάστημα κτλ.).

Στη περίπτωση της δικής μας προσομοίωσης θα ασχοληθούμε με δύο στόχους batch πειραμάτων:

- Χρόνος πρώτης εύρεσης του missing person υπό συνδυασμούς παραμέτρων.
- Φορές που βρέθηκε ο missing person μέσα σε ένα συγκεκριμένο χρονικό πλαίσιο.

Τα αποτελέσματα των batch πειραμάτων θα πρέπει να εξάγονται σε ένα αρχείο κατάλληλης μορφοποίησης, έτσι ώστε να διευκολύνεται η περαιτέρω ανάλυση από τον χρήστη.

4.3. Εισαγωγή Δεδομένων Χαρτών στο Μοντέλο

Η εισαγωγή δεδομένων στο μοντέλο είναι κρίσιμης σημασίας για την εξαγωγή ουσιαστικών αποτελεσμάτων από τη λειτουργία του.

Τα δεδομένα που εισάγονται από το χρήστη χωρίζονται σε τρεις κατηγορίες:

1. Δεδομένα Χάρτη
2. Αρχική θέση και ώρα κατά τις οποίες ξεκίνησε το παιδί να αγνοείται
3. Εισαγωγή Παραμέτρων

Στην παρούσα ενότητα θα επικεντρωθούμε στην εισαγωγή δεδομένων χαρτών στην προσομοίωση καθώς οι άλλες δύο κατηγορίες θα αναλυθούν διεξοδικά στη συνέχεια.

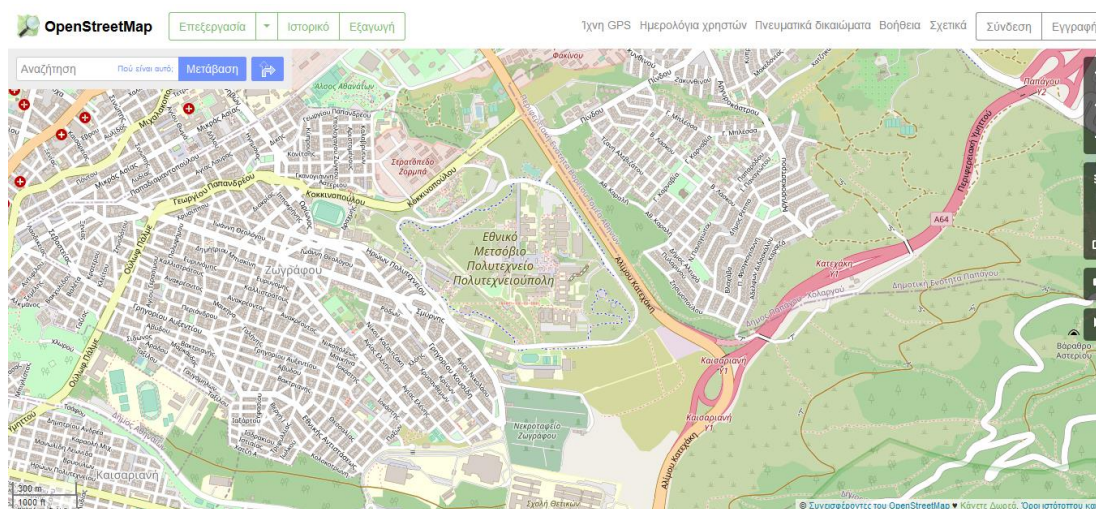
Αναλυτικότερα, η επιλογή, η εισαγωγή και η διαχείριση χαρτών αποτελούν ένα από τα σημαντικότερα δομικά στοιχεία της προσομοίωσης. Ιδιαίτερη σημασία έχει η εξαγωγή χαρτών να γίνεται με ευκολία, χωρίς κόστος, και οι χάρτες να περιέχουν αξιόπιστα δεδομένα. Εξίσου σημαντική είναι η εύκολη εισαγωγή των δεδομένων των χαρτών στην προσομοίωση. Και οι δύο αυτοί παράγοντες είναι καθοριστικοί για την ευχρηστία του εγχειρίματος της προσομοίωσης.

Στόχος συνεπώς αυτής της ενότητας είναι μία πρακτική εξαγωγή χαρτών των περιοχών που μας ενδιαφέρουν, και η εισαγωγή τους στην προσομοίωση.

4.3.1. Η πλατφόρμα Open Street Map

Η πλατφόρμα που θα χρησιμοποιήσουμε ως πηγή χαρτών για την προσομοίωση είναι η Open Street Map (OSM).

Το OSM project είναι ένα συνεργατικό, open-source project, το οποίο στοχεύει στην δημιουργία ενός δωρεάν στην χρήση, ολοκληρωμένου, παγκοσμίου συστήματος χαρτογράφησης [21]. Στην Εικόνα 4.1. βλέπουμε την αρχική σελίδα OSM project. Η ανάγκη για την ανάπτυξη του δημιουργήθηκε από την έλλειψη χαρτογραφικών δεδομένων σε πολλές περιοχές ανά τον κόσμο, καθώς και την ύπαρξη περιορισμών στη χρήση δεδομένων από άλλες πηγές.



Εικόνα 4.1. Η αρχική σελίδα της πλατφόρμας OSM

Πηγή: www.openstreetmap.org

Η πλατφόρμα OSM επιλέχθηκε για χρήση στην προσομοίωσή μας λόγω της ανοιχτής, open source φύσης της όπως προαναφέρθηκε, καθώς και της συμβατότητάς της με την Gama.

Οι χάρτες που θα χρησιμοποιηθούν θα είναι αρχεία τύπου **.osm**

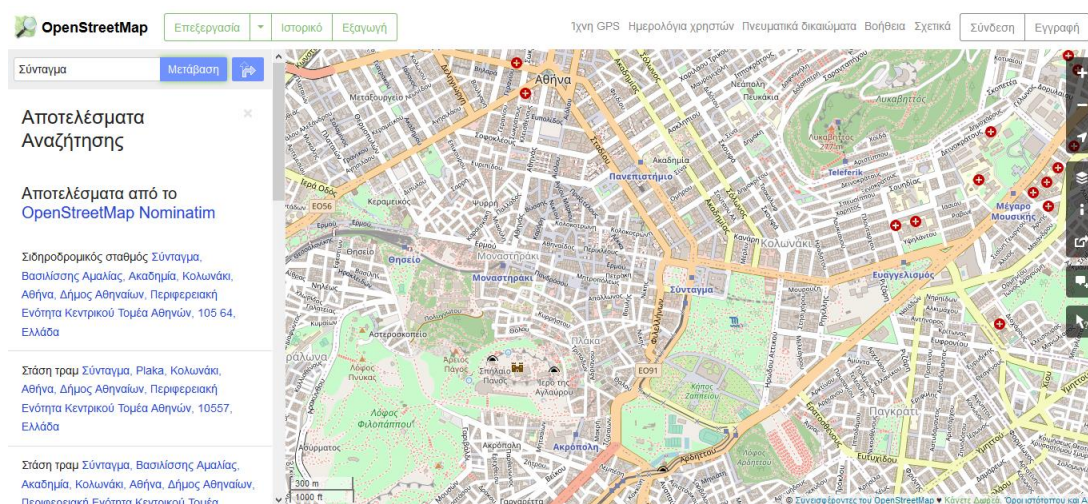
4.3.2. Εξαγωγή Χαρτών .osm

Θα αναλύσουμε δύο τρόπους εξαγωγής χαρτών τύπου .osm:

- Απευθείας εξαγωγή χαρτών από την ιστοσελίδα της OSM.
- Εξαγωγή χαρτών από την ιστοσελίδα Extract.BBBike.

4.3.2.1 Απευθείας Εξαγωγή Χαρτών από την ιστοσελίδα OSM

Προκειμένου να ξεκινήσουμε τη διαδικασία εξαγωγής βρίσκουμε πρώτα στον χάρτη την περιοχή που μας ενδιαφέρει. Για να το κάνουμε αυτό, είτε συμπληρώνουμε απευθείας στο πλαίσιο αναζήτησης - όπως βλέπουμε στην Εικόνα 4.2 - την περιοχή στην οποία επιθυμούμε να τρέξουμε το πείραμα (την περιοχή στην οποία χάθηκε ο missing person), είτε μετακινούμαστε εκεί χειροκίνητα (αριστερό κλικ + drag), όπως βλέπουμε στην εικόνα 4.3.



Εικόνα 4.2. Στο πάνω αριστερά μέρος της εικόνας βρίσκεται το πλαίσιο αναζήτησης. Στο συγκεκριμένο παράδειγμα αναζητούμε την πλατεία Συντάγματος.

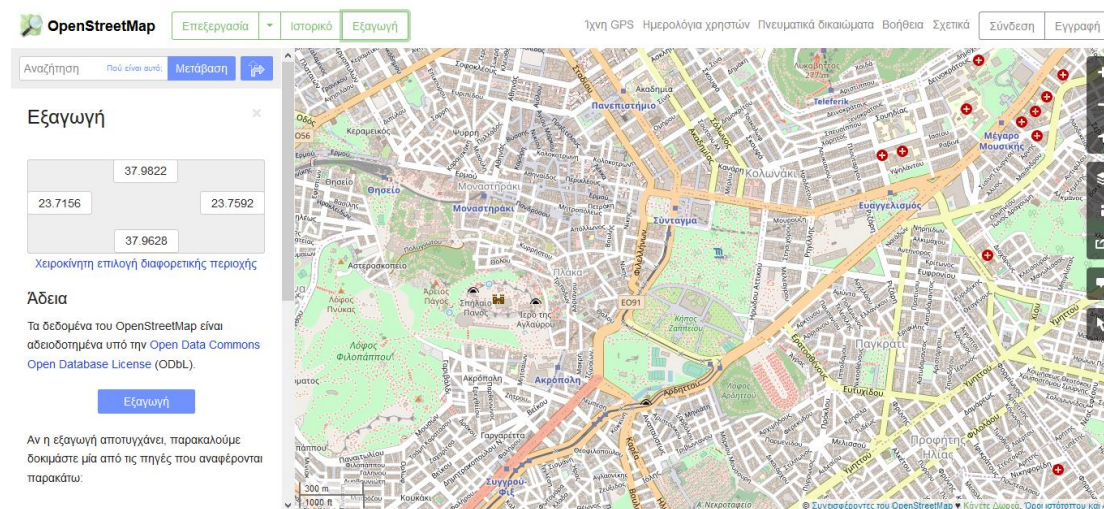
Πηγή: www.openstreetmap.org



Εικόνα 4.3. Χρήση του mouse για χειροκίνητη μετακίνηση στην επιθυμητή περιοχή με κλικ + drag στον χάρτη της OSM.

Πηγή: www.openstreetmap.org

Στη συνέχεια, πατάμε το κουμπί «Εξαγωγή», το οποίο βρίσκεται σταθερά στο header της ιστοσελίδας, και εμφανίζεται στο αριστερό μέρος της ιστοσελίδας το πλαίσιο εξαγωγής της Εικόνας 4.4.

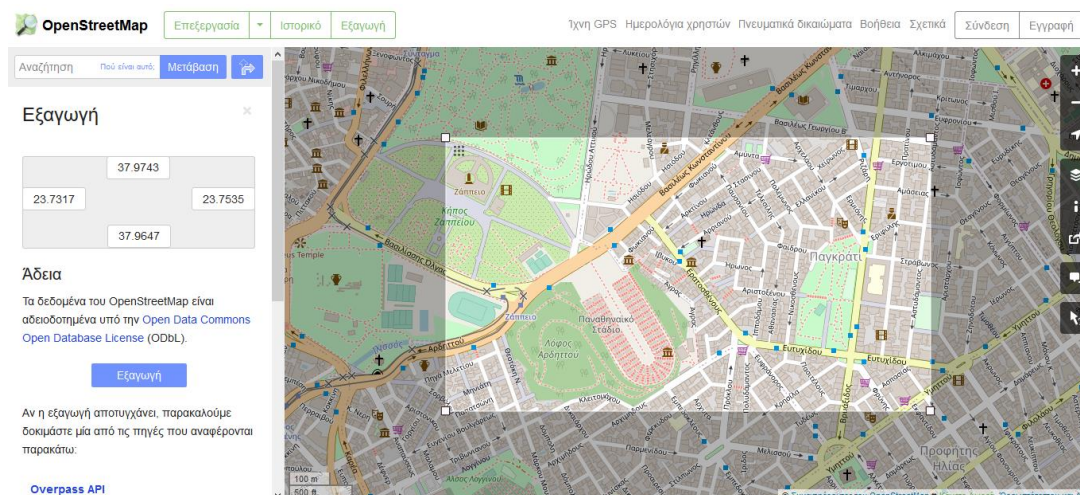


Εικόνα 4.4. Εμφάνιση του πλαισίου Εξαγωγής στο αριστερό μέρος της ιστοσελίδας, και εσωτερικά του πλαισίου εμφάνιση της επιλογής «Εξαγωγή» σε χαρακτηριστικό μπλε φόντο και της επιλογής «Χειροκίνητη επιλογή διαφορετικής περιοχής» με μπλε γραμματοσειρά.

Πηγή: www.openstreetmap.org

Πατώντας κατευθείαν την επιλογή «Εξαγωγή» σε χαρακτηριστικό μπλε φόντο (Εικόνα 4.4.), γίνεται μεταφόρτωση ολόκληρου του τμήματος χάρτη που απεικονίζεται εκείνη τη στιγμή.

Εναλλακτικά, στην περίπτωση που χρειαζόμαστε ένα τμήμα του χάρτη επιλέγουμε «Χειροκίνητη επιλογή διαφορετικής περιοχής» (Εικόνα 4.4.). Εμφανίζεται ένα τετράγωνο πλαίσιο το οποίο μπορούμε να μετακινήσουμε ώστε να περικλείει την επιθυμητή περιοχή όπως βλέπουμε στην Εικόνα 4.5.

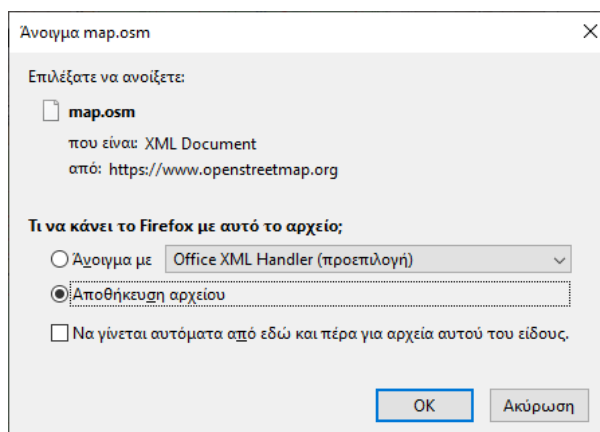


Εικόνα 4.5. Τετράγωνο χειροκίνητης επιλογής περιοχής (τμήματος του εικονιζόμενου χάρτη) προς εξαγωγή.

Πηγή: www.openstreetmap.org

Όταν έχουμε οριστικοποιήσει την επιλογή περιοχής, κάνουμε κλικ στο μπλε πλαίσιο επιλογής με τη σήμανση «Εξαγωγή».

Τότε εμφανίζεται το αναδυόμενο παράθυρο της Εικόνας 4.6. που μας ρωτά αν θέλουμε να αποθηκεύσουμε ή μόνο να ανοίξουμε το map.osm αρχείο. Επιλέγουμε την «Αποθήκευση Αρχείου» και πατάμε OK.



Εικόνα 4.6. Αναδυόμενο παράθυρο για χειρισμό αρχείου .osm

Σημείωση:

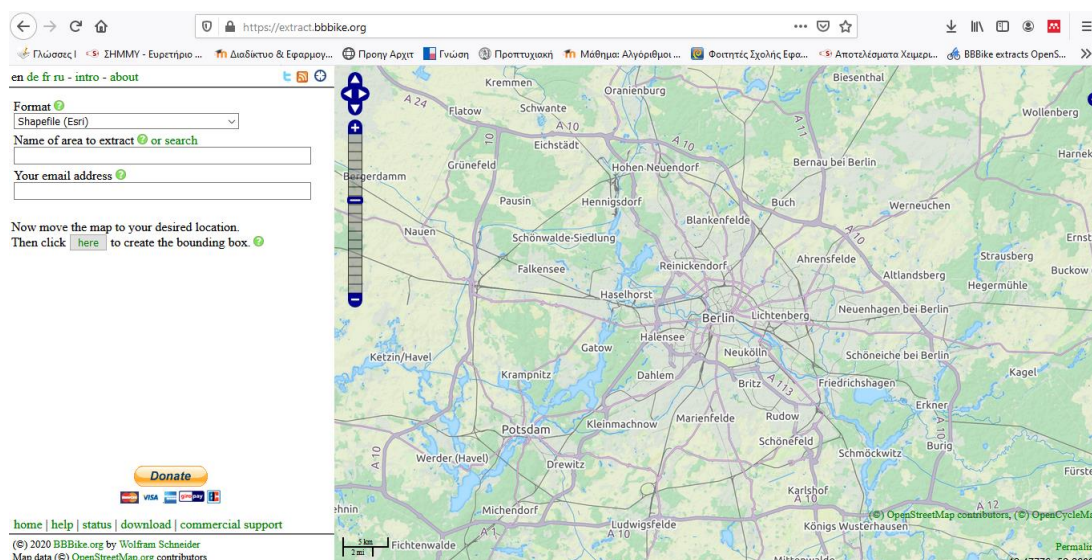
Στην περίπτωση που εμφανιστεί παράθυρο με το μήνυμα που φαίνεται στην Εικόνα 4.7., ο χάρτης που επιλέξαμε είναι μεγαλύτερος από τα επιτρεπτά όρια της πλατφόρμας, οπότε μετακινούμαστε στον δεύτερο τρόπο εξαγωγής που θα αναλύσουμε.

You requested too many nodes (limit is 50000). Either request a smaller area, or use planet.osm

Εικόνα 4.7. Προειδοποίηση για υπέρβαση ορίου μεγέθους επιλεγμένου χάρτη για εξαγωγή.

4.3.2.2 Εξαγωγή Χαρτών μέσω της Extract.BBBike

Ο δεύτερος τρόπος για εξαγωγή χαρτών τύπου .osm που θα αναλύσουμε είναι η εξαγωγή τους μέσω της ιστοσελίδας <https://extract.bbbike.org/> (Εικόνα 4.8.).



Εικόνα 4.8. Η αρχική σελίδα του τμήματος extract της ιστοσελίδας BBBike.

Πηγή: <https://extract.bbbike.org/>

Οι περιπτώσεις στις οποίες μας διευκολύνει η πλατφόρμα εξαγωγής BBBike είναι οι εξής:

- Όταν θέλουμε να εξαγάγουμε το χάρτη μίας περιοχής με συγκεκριμένο (πολυγωνικό) και όχι τετράγωνο σχήμα.
- Όταν υπερβαίνουμε το όριο μεγέθους της OSM. Συγκεκριμένα, το όριο μεγέθους εξαγωγής της BBBike είναι 512MB. Το όριο μεγέθους εξαγωγής της OSM δεν ορίζεται σαφώς σε MB, είναι παρ' όλα αυτά σημαντικά μικρότερο.

Η πλατφόρμα BBBike χρησιμοποιεί την Open Street Map ως βάση και μας επιτρέπει να εξαγάγουμε περιοχές χαρτών σε έναν σημαντικό αριθμό τύπων αρχείων.

Θα την χρησιμοποιούμε για να μεταφορτώσουμε αρχεία .osm όπως και πριν, αλλά τώρα θα μας δίνεται η δυνατότητα να προσαρμόσουμε τους χάρτες στις απαιτήσεις των

εφαρμογών μας, έχοντας την επιλογή και του μεγαλύτερου ορίου μεγέθους, και της προσαρμογής του σχήματός τους.

Προκειμένου να επιλέξουμε το επιθυμητό τμήμα χάρτη, μετακινούμαστε χειροκίνητα χρησιμοποιώντας το zoom in, zoom out (Εικόνα 4.9.) στο αριστερό μέρος της σελίδας και με αριστερό κλικ + drag φτάνουμε στην επιθυμητή περιοχή.



Εικόνα 4.9. Τα zoom controls του τμήματος extract της ιστοσελίδας BBBike.

Πηγή: <https://extract.bbbike.org/>

Αφότου έχουμε φτάσει στην επιθυμητή περιοχή του χάρτη, κάνουμε κλικ στο πλαίσιο επιλογής “here” (Εικόνα 4.10.).

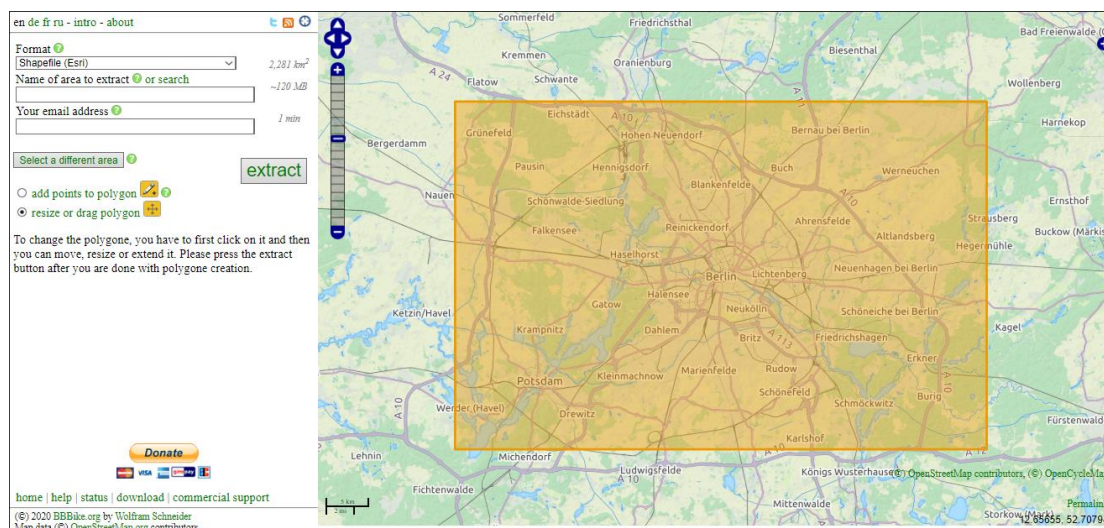
Now move the map to your desired location.
Then click here to create the bounding box. ?

Εικόνα 4.10. Το κουμπί here στο αριστερό τμήμα της ιστοσελίδας extract του BBBike.

Πηγή: <https://extract.bbbike.org/>

Αμέσως μόλις το πατήσουμε δημιουργείται ένα πορτοκαλί πλαίσιο οριοθέτησης (bounding box) τμήματος του χάρτη για μεταφόρτωση. Παράλληλα εμφανίζονται 3 νέες επιλογές στο αριστερό τμήμα της ιστοσελίδας, ακριβώς στο σημείο που επιλέξαμε το «here» (Εικόνα 4.11.). Οι 3 επιλογές είναι:

- a. Επιλογή διαφορετικής περιοχής (select a different area)
- b. Προσθήκη σημείων στο πολύγωνο (add points to polygon)
- c. Αλλαγή μεγέθους ή μετατόπιση του πολυγώνου (resize or drag polygon)



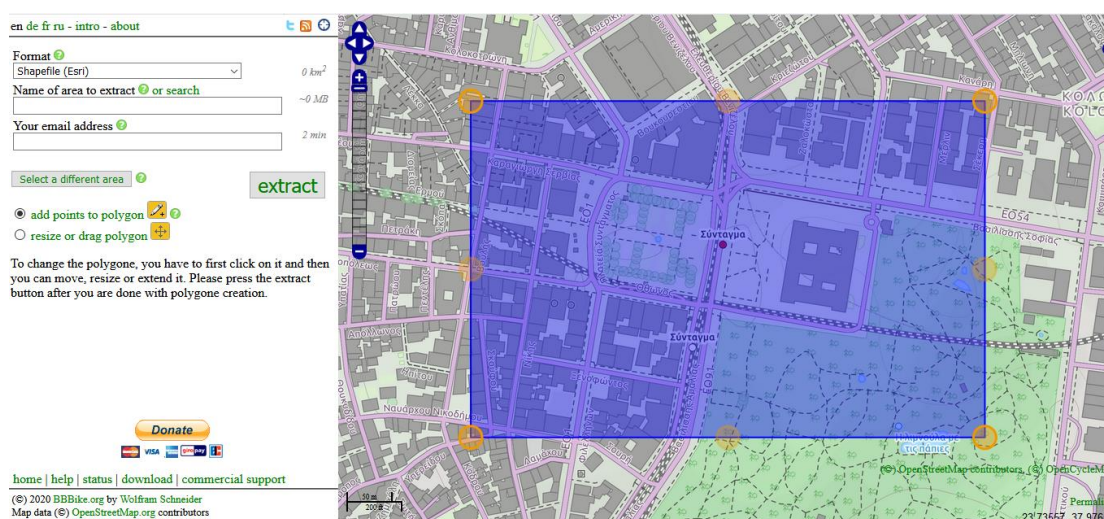
Εικόνα 4.11. Το πορτοκαλί πλαίσιο οριοθέτησης (bounding box) τμήματος του χάρτη για μεταφόρτωση και στο αριστερό μέρος της ιστοσελίδας, οι 3 νέες επιλογές για επεξεργασία του πλαισίου ή επανεκκίνησης της διαδικασίας με επιλογή διαφορετικής περιοχής χάρτη.

Πηγή: <https://extract.bbbike.org/>

Η πρώτη (a.) επιλογή αποτελεί επανεκκίνηση της διαδικασίας επιλογής περιοχής, ενώ οι δύο επόμενες αφορούν την επεξεργασία του πλαισίου οριοθέτησης, όπως αναλύεται παρακάτω.

Η δεύτερη (b.) επιλογή «**add points to a polygon**» μας επιτρέπει να επεξεργαστούμε το σχήμα του πλαισίου οριοθέτησης.

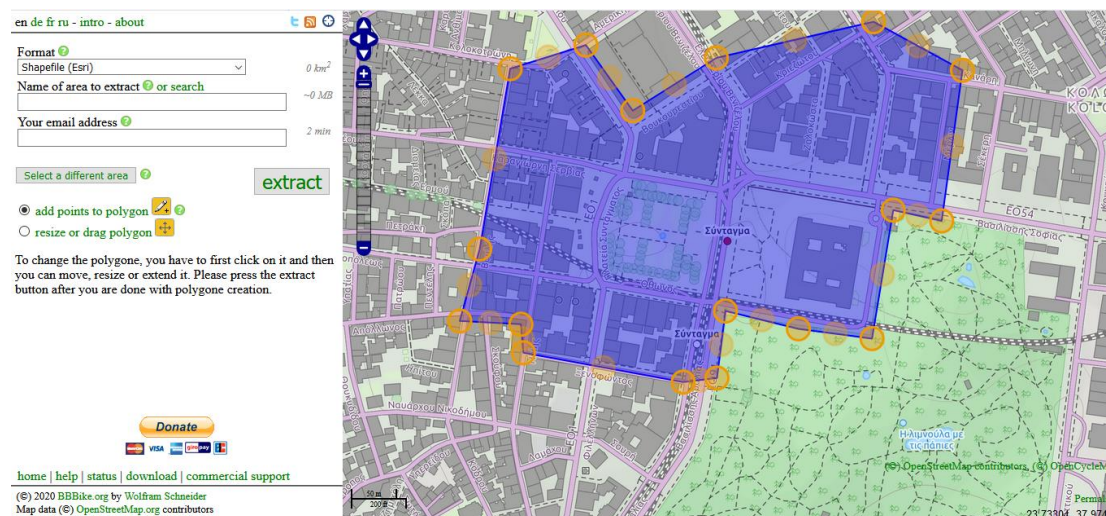
Επιλέγοντάς την και κάνοντας κλικ πάνω στο πλαίσιο οριοθέτησης που έχει εμφανιστεί πάνω στον χάρτη, εμφανίζονται τα κυκλικά σημεία (ακμές του πολυγώνου) από τα οποία μπορούμε να προσαρμόσουμε το σχήμα της περιοχής εξαγωγής, δημιουργώντας ακριβώς το τμήμα του χάρτη που χρειαζόμαστε για την προσομοίωσή (Εικόνα 4.12.).



Εικόνα 4.12. Το αρχικό, τετράγωνο πλαίσιο οριοθέτησης με τα κυκλικά σημεία (ακμές του πολυγώνου) τα οποίες μπορούμε να μετακινήσουμε, μεταβάλλοντας έτσι το σχήμα του πολυγώνου ώστε να προκύψει το κατάλληλο για το πείραμα τμήμα του χάρτη.

Πηγή: <https://extract.bbbike.org/>

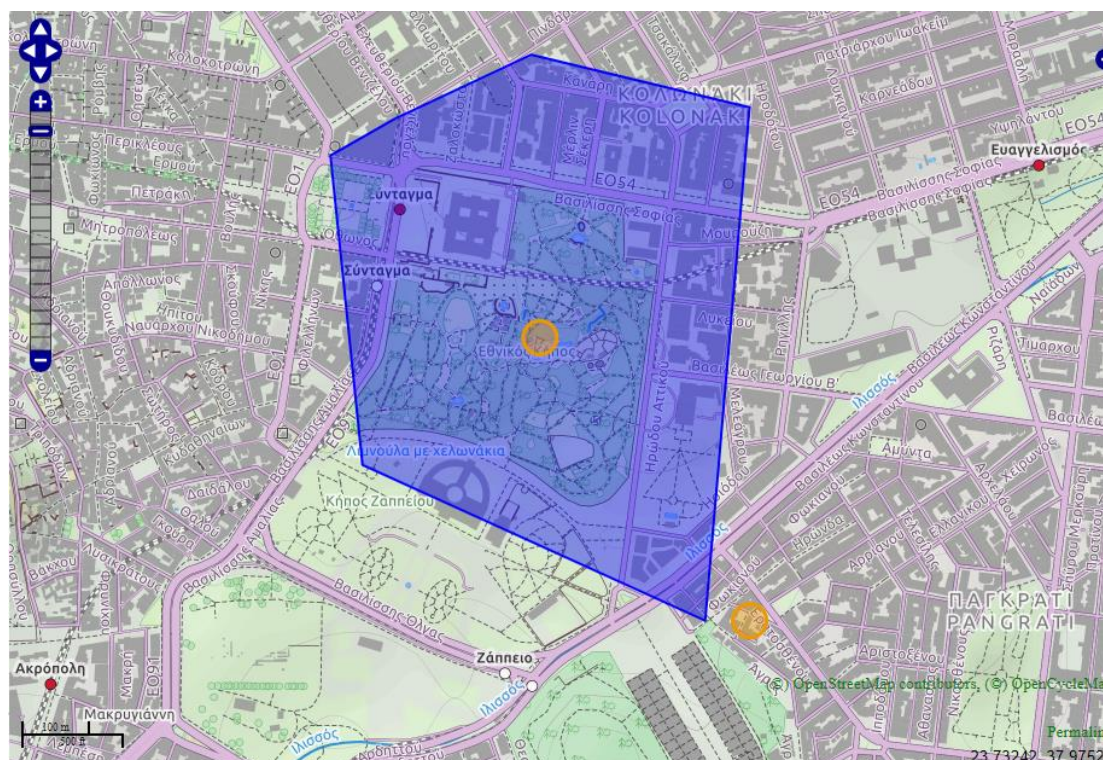
Κάθε φορά που προσαρμόζουμε κάποια από τις ακμές του πολυγώνου εμφανίζεται μία επιπλέον δίνοντας μας πολλαπλές δυνατότητες προσαρμογής. Στην Εικόνα 4.13. βλέπουμε ένα πλαίσιο οριοθέτησης το οποίο έχει προσαρμοστεί έτσι ώστε να περιλαμβάνει μόνο συγκεκριμένους δρόμους και κτήρια.



Εικόνα 4.13. Το προσαρμοσμένο πλαίσιο επιλογής του χάρτη προς εξαγωγή. Βλέπουμε τις νέες πολυγωνικές ακμές που δημιουργήθηκαν.

Πηγή: <https://extract.bbbike.org/>

Η τρίτη (c.) επιλογή «**resize or drag polygon**» μας επιτρέπει είτε να αλλάξουμε το μέγεθος του πολυγώνου, μικραίνοντας ή μεγαλώνοντάς το, είτε να το μετακινήσουμε, διατηρώντας και στις δύο περιπτώσεις το σχήμα του. Στην Εικόνα 4.14. έχει επιλεγθεί το «**resize or drag polygon**». Παρατηρούμε την εμφάνιση δύο νέων κυκλικών σημείων, ένα στη μέση του πολυγώνου και ένα στην κάτω δεξιά γωνία εκτός του πλαισίου. Με κλικ + drag στο κεντρικό κυκλικό σημείο μετακινούμε το πλαίσιο οριοθέτησης, ενώ με κλικ + drag στο δεύτερο αλλάζουμε το μέγεθός του πλαισίου.



Εικόνα 4.14. Οι δυνατότητες τροποποίησης του πλαισίου οριοθέτησης με την επιλογή του «**resize or drag polygon**» εμφανίζονται με τη μορφή δύο κυκλικών σημείων.

Πηγή: <https://extract.bbbike.org/>

Έχοντας ολοκληρώσει την επιλογή και προσαρμογή του πλαισίου που οριοθετεί τον χάρτη που θέλουμε να χρησιμοποιήσουμε, μετακινούμαστε στα 3 πεδία στην κορυφή του αριστερού πλαισίου (Εικόνα 4.15):

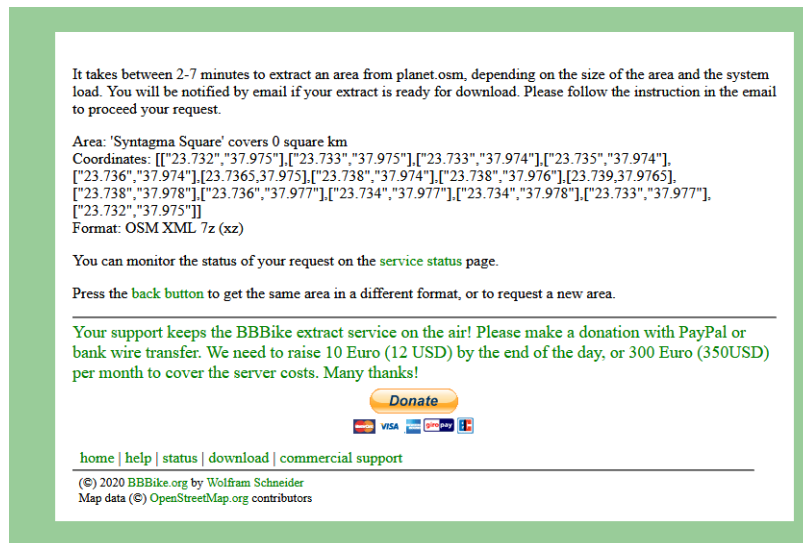
- Στο πεδίο «μορφή» (Format), επιλέγουμε το OSM XML 7z (xz).
- Στο πεδίο «όνομα περιοχής προς εξαγωγή» (Name of area to Extract) δίνουμε το όνομα που θέλουμε να έχει ο χάρτης μας.
- Στο πεδίο «διεύθυνση ηλεκτρονικού ταχυδρομείου» (Your email address) συμπληρώνουμε τη διεύθυνση ηλεκτρονικού ταχυδρομείου μας. Εκεί θα λάβουμε την ειδοποίηση ότι ο χάρτης μας είναι έτοιμος και τον σύνδεσμο από τον οποίο θα τον μεταφορτώσουμε.

Μόλις συμπληρώσουμε τα 3 πεδία, πατάμε το κουμπί «Εξαγωγή» («extract») που βρίσκεται ακριβώς από κάτω προκειμένου να ξεκινήσει η διαδικασία εξαγωγής του χάρτη μας.

Εικόνα 4.15. Το πλαίσιο συμπλήρωσης των απαραίτητων πεδίων και το πλαίσιο επιλογής «extract».

Πηγή: <https://extract.bbbike.org/>

Αφότου πατήσουμε το κουμπί Extract, μεταβαίνουμε αυτόματα στην σελίδα που φαίνεται στην Εικόνα 4.16.

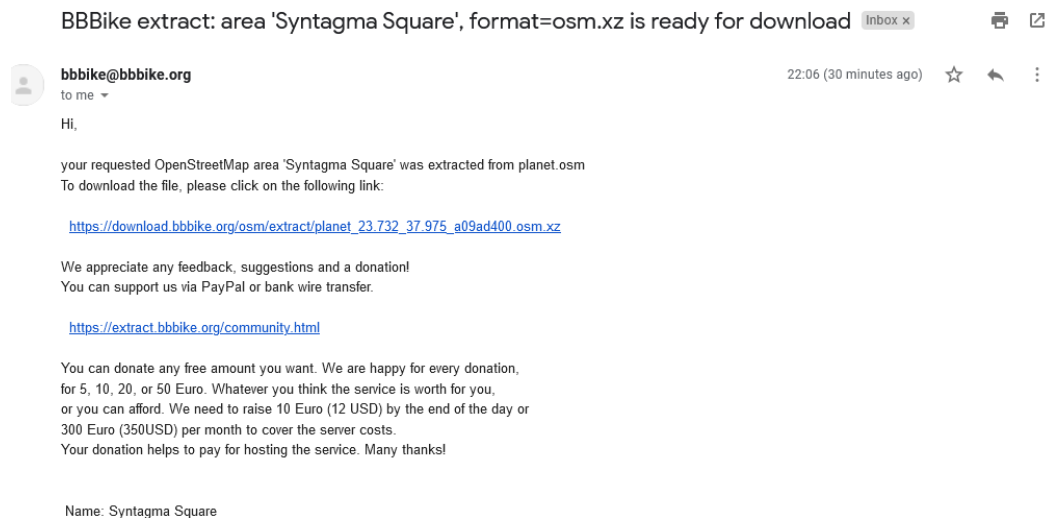


Εικόνα 4.16. Η σελίδα που εμφανίζεται μετά από επιτυχημένο αίτημα για extraction.

Πηγή: <https://extract.bbbike.org/>

Η διαδικασία της εξαγωγής ολοκληρώθηκε και θα χρειαστεί να περιμένουμε 2-7 λεπτά προκειμένου η πλατφόρμα να επεξεργαστεί το αίτημά μας, και να λάβουμε ειδοποίηση στο email που δηλώσαμε.

Μόλις λάβουμε το email, πατάμε τον σύνδεσμο τον οποίο περιέχει για να κατεβάσουμε το αρχείο μας. Ένα παράδειγμα τέτοιου email και αντίστοιχου συνδέσμου βλέπουμε στην Εικόνα 4.17.



Εικόνα 4.17. Ενδεικτικό email από το BBBike με το link για τη μεταφόρτωση του επιλεγμένου χάρτη.

Κάνοντας κλικ πάνω στον σύνδεσμο, εμφανίζεται η επιλογή για το άνοιγμα ενός αρχείου σε συμπιεσμένη μορφή. Το αποσυμπιέζουμε και το αποθηκεύουμε σε ένα κατάλληλο μέρος στον υπολογιστή μας.

Σημείωση: Εάν το αποσυμπιεσμένο αρχείο δεν ακολουθείται από *extension .osm*, δεν είναι δηλαδή τύπος αρχείου *osm* για το λειτουργικό μας σύστημα, απλώς αλλάζουμε τον τύπου αρχείου, μετατρέποντας το *extension* του σε *.osm*.

4.4. Κώδικας του Μοντέλου

Σε αυτό το σημείο θα αναλύσουμε τον κώδικα των σημαντικότερων επιλογών υλοποίησης της εφαρμογής μας όπως έχουν οριστεί στο **Κεφάλαιο 4.2**. Ο κώδικας της προσομοίωσης είναι διαθέσιμος για μεταφόρτωση στο αποθετήριο κώδικα (repository) Github, [εδώ](https://github.com/iasonlazzpap/Thesis-Project)¹⁰.

Πριν συνεχίσουμε όμως με την ανάλυση του κώδικα, είναι απαραίτητο να αναφερθούμε σύντομα σε κάποιες βασικές έννοιες.

Κλάση Πράκτορα (Agent Species):

Μία κλάση πράκτορα ορίζεται με βάση τα κοινά γνωρίσματα τα οποία υπάρχουν σε κάθε ένα μεμονωμένο πράκτορα (agent instance) αυτής της κλάσης.

Ένα απλό παράδειγμα για την κατανόηση της έννοιας των κλάσεων πρακτόρων είναι το εξής:

¹⁰ <https://github.com/iasonlazzpap/Thesis-Project>

Έστω ότι έχουμε την κλάση πράκτορα «άνθρωπος». Η κλάση αυτή χαρακτηρίζεται από γνωρίσματα τα οποία είναι κοινά για όλους τους ανθρώπους, για παράδειγμα έχουν ένα όνομα, έναν τόπο γέννησης, μπορούν να μιλούν μία ή περισσότερες γλώσσες, μπορούν να περπατούν, να οδηγούν κτλ. Αντίθετα, ένα instance αυτής της κλάσης είναι ένας συγκεκριμένος πράκτορας – άνθρωπος. Είναι ο Θησέας, γεννήθηκε στο Μαρούσι, μιλάει Ελληνικά και Ιταλικά, περπατάει, έχει δίπλωμα για αυτοκίνητο και οδηγεί.

Με βάση τα ανωτέρω, στην προσομοίωσή μας θα έχουμε την κλάση People Agent και ένα instance της, τον people315. Όλοι οι People Agents έχουν μεταξύ άλλων χαρακτηριστικών, ένα κτήριο για σπίτι, και ένα κτήριο στο οποίο δουλεύουν. Ο people315 όμως έχει ένα συγκεκριμένο κτήριο για σπίτι του, το building 42 και ένα συγκεκριμένο κτήριο στο οποίο δουλεύει, το building17.

Αντανακλαστικά Πρακτόρων (Reflex):

Ένα αντανακλαστικό ή μία συμπεριφορά (reflex) είναι ένα τμήμα κώδικα υπό μία ορισμένη συνθήκη, η οποία ελέγχεται αν πληρούνται (είναι αληθής) σε κάθε βήμα της προσομοίωσης. Στην προσομοίωσή μας το βήμα έχει οριστεί ως ένα λεπτό της ώρας, επομένως όλα τα αντανακλαστικά εξετάζονται μία φορά για κάθε λεπτό που περνάει. Σε περίπτωση που η συνθήκη προκύψει αληθής, οι εντολές που έχουν προγραμματιστεί στο αντανακλαστικό εκτελούνται, σε διαφορετική περίπτωση μετακινούμαστε παρακάτω στη ροή εκτέλεσης.

Τα αντανακλαστικά είναι σημαντικό κομμάτι του προγραμματισμού σε πλατφόρμα GAMA, και μας διευκολύνουν σημαντικά στην προσομοίωση ανθρώπινων κινήσεων και συμπεριφορών, διότι προσεγγίζουν στο πεδίο της προσομοίωσης τον τρόπο με τον οποίο οι άνθρωποι αντιδρούν σε πραγματικά ερεθίσματα.

Για παράδειγμα, ένα από τα αντανακλαστικά που έχουν οριστεί για όλους τους People Agents είναι το “time_to_go_home”. Δεδομένου ότι για κάθε έναν από τους People Agents έχει οριστεί διαφορετικό ωράριο εργασίας, όταν κάθε People Agent φτάσει στο τέλος του ωραρίου εργασίας του για τη συγκεκριμένη μέρα, τότε η συνθήκη του αντανακλαστικού “time_to_go_home” γίνεται αληθής. Αυτό οδηγεί στην αλλαγή του στόχου του συγκεκριμένου People Agent, όπου πλέον ο νέος στόχος προορισμού είναι το κτήριο-σπίτι του και συνεπώς θα ξεκινήσει τη μετακίνηση προς το σπίτι του.

Όταν ενεργοποιηθεί το αντανακλαστικό “time_to_go_home” για κάποιον συγκεκριμένο People Agent, π.χ. τον people315 στον οποίο αναφερθήκαμε προηγουμένως, στο επόμενο βήμα της προσομοίωσης θα ενεργοποιηθεί είτε το αντανακλαστικό του περπατήματος, είτε το αντανακλαστικό μετακίνησης με όχημα, ανάλογα με τις προδιαγραφές που έχουν τεθεί, έτσι ώστε ο people315 να φτάσει στο σπίτι του.

Παράμετρος (parameter):

Η πλατφόρμα GAMA μας δίνει τη δυνατότητα να ορίσουμε παραμέτρους εισόδου, οι τιμές των οποίων μπορούν να αλλάζουν από τον χρήστη εύκολα για κάθε εκτέλεση του

πειράματος μέσω του γραφικού περιβάλλοντος, χωρίς αλλαγές στον κώδικα της προσομοίωσης.

Η κάθε παράμετρος συνδέεται κατά την κωδικοποίησή της με μία καθολική μεταβλητή. Οι τιμές της παραμέτρου και της μεταβλητής κατ' ουσίαν ταυτίζονται. Η διαφορά έγκειται στο γεγονός ότι μόνο η παράμετρος εμφανίζεται στο περιβάλλον εκτέλεσης. Με κάθε αλλαγή της παραμέτρου από το χρήστη στο γραφικό περιβάλλον, αλλάζει και η τιμή της μεταβλητής στην εκτέλεση του κώδικα.

Για την περίπτωση που ο χρήστης δεν τροποποιήσει την παράμετρο, η καθολική μεταβλητή διαθέτει και διατηρεί μία προκαθορισμένη (default) τιμή.

Προκειμένου να γίνουν περισσότερο κατανοητές οι έννοιες παραμέτρου και καθολικής μεταβλητής, παραθέτουμε ένα παράδειγμα:

Στην προσομοίωσή μας ορίζουμε την παράμετρο “Number of people agents” που συνδέεται με την καθολική μεταβλητή “nb_people”.

Η “nb_people” έχει προκαθορισμένη τιμή ίση με 200:

```
int nb_people <- 200; //number of people in the simulation
```

Ο κώδικας που ορίζει την σύνδεση της παραμέτρου με την global μεταβλητή της είναι:

```
//Determines number of people agents in simulation using global var
nb_people
parameter "Number of people agents" var: nb_people category: "GIS" ;
```

Σημείωση: Το όρισμα “category” προσδιορίζει την κατηγορία στην οποία θα εμφανίζεται η παράμετρος στο γραφικό περιβάλλον. Με αυτό τον τρόπο, παράμετροι που αφορούν την ίδια κατηγορία ομαδοποιούνται.

Η παράμετρος παρουσιάζεται στο περιβάλλον της εκτέλεσης όπως φαίνεται στην Εικόνα 4.18.

Number of people agents 473

Εικόνα 4.18. Πλαίσιο παραμέτρου “Number of people agents” κατά την διάρκεια εκτέλεσης του πειράματος. Ο χρήστης μπορεί να αλλάξει την τιμή της παραμέτρου γράφοντας διαφορετικό αριθμό στο πλαίσιο.

Η καθολική μεταβλητή χρησιμοποιείται για να αλλάξει τη συμπεριφορά της προσομοίωσης, στο συγκεκριμένο παράδειγμα τον αριθμό των people agents. Το πραγματοποιεί αυτό με την κατάλληλη χρήση της στον κώδικα. Εδώ, ορίζει τον αριθμό των people agents που θα δημιουργηθούν στη συνάρτηση δημιουργίας τους:

```
//the function that creates the people agents
create people number: nb_people {
```

```
...
}
```

Με αντίστοιχο τρόπο δουλεύουν και οι υπόλοιπες παράμετροι της προσομοίωσης. Συνεπώς, εφεξής, θα χρησιμοποιούμε τη φράση «παραμετροποιήσιμο» αναφερόμενοι σε αυτή την συσχετισμένη λειτουργία, χωρίς να προβαίνουμε σε επιπλέον ανάλυση.

4.4.1. Κωδικοποίηση Χαρτών

a. Παρουσιάσαμε στο κεφάλαιο 4.3. τον τρόπο με τον οποία εξάγουμε πραγματικά δεδομένα χαρτών σε αρχεία τύπου .osm. Τώρα θα αναλύσουμε πώς διαχειριζόμαστε αυτά τα αρχεία στον κώδικά μας, προκειμένου να δημιουργήσουμε τον χάρτη της προσομοίωσης.

Ο χάρτης της προσομοίωσης αποτελείται από τα κτήρια (**building agents**), τους δρόμους (**road agents**) και τον **γράφο κίνησης**, ο οποίος δημιουργείται από τις θέσεις των δύο πρώτων. Η δημιουργία των building agents, των road agents αλλά και του γράφου κίνησης γίνεται μέσω της φόρτωσης του αρχείου .osm και της επεξεργασίας τους μέσω των κατάλληλων συναρτήσεων της GAMA.

Ένα αρχείο τύπου .osm περιέχει ένα πλήθος χαρακτηριστικών για τον χάρτη τον οποίο απεικονίζει. Τα χαρακτηριστικά αυτά περιλαμβάνουν τα κτήρια, τους δρόμους, αλλά και συμπληρωματικές πληροφορίες όπως μουσεία, πυροσβεστικούς σταθμούς, μετεωρολογικούς σταθμούς και πολλά ακόμη. Το πλήρες σύνολο χαρακτηριστικών που υποστηρίζει ένα αρχείο .osm μπορεί να βρεθεί [εδώ](https://wiki.openstreetmap.org/wiki/Map_Features)¹¹.

Για τους σκοπούς της προσομοίωσης, και προκειμένου να διατηρήσουμε τη συμβατότητα με όσο το δυνατόν μεγαλύτερο πλήθος χαρτών, θα φιλτράρουμε τα διαθέσιμα χαρακτηριστικά, εξάγοντας από το σύνολο τους, μόνο τα κτήρια και τους δρόμους τα οποία περιέχονται στο αρχείο. Για το σκοπό αυτό θα δημιουργήσουμε τη συνάρτηση “filtering”, την οποία θα εφαρμόσουμε σε ολόκληρο το αρχείο τύπου .osm.

```
//GIS Input//
//map used to filter the object to build from the OSM file according to
//attributes. for an exhaustive list, see:
//http://wiki.openstreetmap.org/wiki/Map_Features
map filtering <- ([ "highway"::["primary", "secondary", "tertiary",
                        "motorway", "living_street", "residential",
                        "unclassified"], "building"::["yes"]]);
```

Αρχικά ο χάρτης της προσομοίωσης δημιουργείται χρησιμοποιώντας το προφορτωμένο αρχείο τύπου .osm map(6).osm, Το αρχείο απεικονίζει μία πυκνοκατοικημένη περιοχή στο κέντρο της Αθήνας.

```
//OSM file to load
```

¹¹ https://wiki.openstreetmap.org/wiki/Map_Features

```
file<geometry> osmfile <-
  file<geometry>(osm_file("../includes/map(6).osm", filtering));
```

Το πλαίσιο (shape) μέσα στο οποίο θα περικλείεται ο χώρος μας, δημιουργείται αυτόματα από τα όρια του χάρτη χωρίς να χρειαστεί να δοθούν χειροκίνητα συγκεκριμένες διαστάσεις.

```
//compute the size of the environment from the envelope of the OSM file
geometry shape <- envelope(osmfile);
```

Θα ορίσουμε την βοηθητική κλάση “osm_agent” στην οποία θα φορτώσουμε τα χαρακτηριστικά road και building που εξάγουμε από το “osm_file”:

```
species osm_agent {
  string highway_str;
  string building_str;
}
```

Χρησιμοποιείται, επίσης, η κλάση πρακτόρων building η οποία ορίζει ότι όλα τα κτήρια θα έχουν έναν τύπο (π.χ. ουρανοξύστης) και το προεπιλεγμένο τους χρώμα θα είναι γκρι:

```
//define the building species
species building {
  string type;
  rgb color <- #gray ; //the color of each building

  aspect base {
    draw shape color: color ;
  }
}
```

Ορίζεται, επίσης, η κλάση road, σύμφωνα με την οποία όλοι οι δρόμοι θα έχουν έναν τύπο (π.χ. αστικός δρόμος) και ένα απαλό πράσινο χρώμα:

```
//define the road species
species road {
  string type;
  rgb color <- #gamagreen;

  aspect base {
    draw shape color: color ;
  }
}
```

Έχοντας ολοκληρώσει τις δηλώσεις τύπων μπορούμε να δημιουργήσουμε ένα instance του osm_agent από το “osmfile”:

```
//possibility to load all of the attributes of the OSM data: for an
//exhaustive list, see: http://wiki.openstreetmap.org/wiki/Map\_Features
```

```
create osm_agent from:osmfile with:
[highway_str::string(read("highway")),
building_str::string(read("building"))];
```

Χρησιμοποιώντας τον γενικό agent, osm_agent κατασκευάζουμε όλους τους building και road agents:

```
//possibility to load all of the attributes of the OSM data: for an
exhaustive list, see: http://wiki.openstreetmap.org/wiki/Map_Features
create osm_agent from:osmfile with:
[highway_str::string(read("highway")),
building_str::string(read("building"))];

//from the created generic agents, creation of the selected agents
ask osm_agent {
  if (length(shape.points) = 1 and highway_str != nil ) {
    //do nothing, crossroad
  }
  else {
    if (highway_str != nil ) {
      create road with: [shape ::shape, type:: highway_str];
    } else if (building_str != nil){
      create building with: [shape ::shape];
    }
  }
  //do the generic agent die
  do die;
}
```

Τέλος, χρησιμοποιώντας το σύνολο των road agents δημιουργούμε τον γράφο κίνησης (graph) πάνω στον οποίο θα κινούνται οι people agents και ο missing person.

```
//create the graph (already initialized) as an edge graph
the_graph <- as_edge_graph(road);
```

b. Προκειμένου να μπορεί ο χρήστης να αλλάζει τον χάρτη της προσομοίωσης σε κάθε πείραμα, δημιουργήσαμε μία παράμετρο η οποία συνδέεται με την καθολική μεταβλητή osmfile η οποία ορίζει τον χάρτη εισόδου.

```
parameter "Simulation Map (type: .osm)" var: osmfile category: "GIS" ;
```

4.4.2. Προγραμματισμός People Agents

Η κλάση People Agent ορίζει τα γενικά χαρακτηριστικά όλων των ανθρώπων που κινούνται στην προσομοίωση, εκτός του χαμένου παιδιού. Ο ορισμός της κλάσης γίνεται με όμοιο τρόπο με προηγούμενους, ενδιαφέρον παρουσιάζεται όμως στις επιμέρους υλοποιήσεις των προδιαγραφών που θέσαμε.

a. Προσαρμόσιμος αριθμός People Agents από τον χρήστη.

Όπως αναλύσαμε στο παράδειγμα παραμέτρων του κεφαλαίου 4.4., ο προσαρμόσιμος αριθμός People Agents υλοποιείται μέσω της παραμέτρου “Number of people agents” και της καθολικής μεταβλητής “nb_people”.

```
//Determines num of people agents in sim using global var nb_people
parameter "Number of people agents" var: nb_people category: "GIS" ;
```

b. Σε κάθε People Agent αντιστοιχεί ένας συγκεκριμένος χώρος σε ένα κτήριο στο οποία διαμένει (“home_spot”) και ένας συγκεκριμένος χώρος σε ένα κτήριο στο οποίο δουλεύει (“work_spot”). Η τυχαία επιλογή ενός κτηρίου για σπίτι και ενός κτηρίου για δουλειά για κάθε agent instance, γίνεται μέσω της συνάρτησης της GAMA “one_of”. Η συνάρτηση “one_of” παίρνει ως όρισμα μία κλάση και επιστρέφει έναν τυχαίο instance της κλάσης αυτής. Χρησιμοποιώντας την “one_of” θα επιλέξουμε τυχαία από τον πληθυσμό των building agents, ένα κτήριο σπίτι (living_place) και ένα κτήριο δουλειάς (working_place) για κάθε instance People Agent. Μέσα στο κτήριο που επιλέχθηκε κάθε φορά, επιλέγουμε ένα συγκεκριμένο σημείο, ως χώρο του agent για να προσομοιώσουμε το διαμέρισμα ή γραφείο του, αντίστοιχα.

```
//define a living and a working place for each agent from the imported
buildings
living_place <- one_of(building) ;
working_place <- one_of(building) ;

//define specific spot inside building where agent resides or works
home_spot <- any_location_in (living_place);
work_spot <- any_location_in (working_place);
```

c. Οι μετακινήσεις των People Agents συνδέονται με το ωράριο εργασίας τους, αφού όπως έχουμε ορίσει μετακινούνται από και προς τη δουλειά τους. Το ωράριο εργασίας έχει ένα εύρος το οποίο είναι παραμετροποιήσιμο από το χρήστη μέσω τιμών ελαχίστου και μεγίστου.

Ορίζουμε προκαθορισμένες τιμές και τις καθολικές μεταβλητές τους:

```
//variables concerning the times that people go and leave work
respectively
int min_work_start <- 7;
int max_work_start <- 9;
int min_work_end <- 16;
int max_work_end <- 18;
```

Και τις αντίστοιχες παραμέτρους:

```
parameter "Earliest hour to start work" category: "People" var:
min_work_start min: 2 max: 8 step: 0.5;
parameter "Latest hour to start work" var: max_work_start category:
"People" min: 8 max: 12;
parameter "Earliest hour to end work" var: min_work_end category:
"People" min: 12 max: 16;
parameter "Latest hour to end work" var: max_work_end category: "People"
min: 16 max: 23;
```


Με τον ίδιο τρόπο, για την υλοποίηση των μετακινήσεων των People Agents είτε με τα πόδια είτε με μέσα μεταφοράς, αρχικοποιούνται οι καθολικές μεταβλητές που αφορούν το εύρος ταχυτήτων για περπάτημα και για οδήγηση (χρήση μέσων μεταφοράς):

```
//variables concerning the speed that the agents are traveling. Measured
in km/h
float min_walking_speed <- 3 #km / #h;
float max_walking_speed <- 6 #km / #h;
float min_driving_speed <- 5 #km / #h;
float max_driving_speed <- 20 #km / #h;
```

Και συνδέονται με τις αντίστοιχες παραμέτρους:

```
parameter "minimum speed" var: min_walking_speed category: "People" min:
0.1 #km/#h ;
parameter "maximum speed" var: max_walking_speed category: "People" max:
50 #km/#h;
parameter "minimum speed" var: min_driving_speed category: "People" min:
0.1 #km/#h ;
parameter "maximum speed" var: max_driving_speed category: "People" max:
50 #km/#h;
```

d. Σε κάθε μετακίνηση ενός People Agent υπάρχει μία αφετηρία και ένας προορισμός πάνω στο αστικό περιβάλλον της προσομοίωσης. Η διαδρομή αυτή είναι, όπως και στις περισσότερες περιπτώσεις στην πραγματική ζωή η βέλτιστη δυνατή, και ακολουθεί στην προσομοίωσή μας την συντομότερη διαδρομή του γράφου κίνησης.

Η μετακίνηση πάνω στο γράφο κίνησης γίνεται με τον ίδιο τρόπο είτε πρόκειται για μετακίνηση του agent με τα πόδια είτε με μεταφορικά μέσα, απλά γίνεται με διαφορετικές ταχύτητες. Η μετακίνηση και στις δύο περιπτώσεις γίνεται με την ενεργοποίηση ενός αντανακλαστικού του People Agent.

Στο παράδειγμα μας παρακάτω όπου η λογική μεταβλητή “walking_bool” είναι αληθής, μόλις τεθεί προορισμός για τον agent, δηλαδή η μεταβλητή “ **the_target** ” σταματήσει να είναι κενή, ενεργοποιείται το αντανακλαστικό “ **walk** ”.

Μόλις ενεργοποιηθεί το “ **walk** ”, ο agent θα ακολουθήσει, με μία τυχαία ταχύτητα περπατήματος εντός του εύρους που έχει τεθεί, το βέλτιστο μονοπάτι πάνω στο γράφο κίνησης. Το βέλτιστο μονοπάτι δημιουργείται στη GAMA μέσω της συνάρτησης “ **goto** ” στην οποία δίνουμε σαν ορίσματα τον προορισμό και τον γράφο.

Μετά τη δημιουργία του μονοπατιού, ο agent ακολουθεί επαναληπτικά τα κομμάτια του μονοπατιού μέχρι να φτάσει στον προορισμό του, και η μεταβλητή “ **the_target** ” να γίνει ξανά κενή.

```
reflex walk when: (the_target !=nil and walking_bool){
  //boolean indicator initialization
  driving_bool <- false;
  speed <- min_walking_speed + rnd (max_walking_speed -
min_walking_speed) ;
  path_path_followed <- goto(target: the_target, on:the_graph,
return_path: true);
  list<geometry> segments <- path_followed.segments;
```

```

loop line over: segments {
  float dist <- line.perimeter;
}
if the_target = location {
  the_target <- nil;
  //boolean indicator returning to default
  //write "Walking boolean indicator returning to
default";
  walking_bool <- false;
}
}

```

Με ακριβώς ανάλογο τρόπο υλοποιείται και το αντανakλαστικό που αφορά τη μετακίνηση με μεταφορικά μέσα “driving”.

4.4.3. Προγραμματισμός Missing Person

Η κλάση missing person αφορά το χαμένο παιδί της προσομοίωσης και ορίζει τα γενικά χαρακτηριστικά που το αφορούν.

Όπως είναι αναμενόμενο, η κλάση αυτή παρουσιάζει πολλά κοινά χαρακτηριστικά με την People Agents, αφού και οι δύο κλάσεις αφορούν ανθρώπους. Η κλάση missing person ορίζεται στον κώδικα της προσομοίωσης με ανάλογο τρόπο με τις κλάσεις που προαναφέρθηκαν. Τα σημεία που παρουσιάζουν ενδιαφέρον είναι τα ιδιαίτερα σημεία συμπεριφοράς που απαιτείται να έχει ο missing person στην προσομοίωση. Τα ιδιαίτερα αυτά σημεία έχουν αναλυθεί στις προδιαγραφές. Σε αυτή την ενότητα θα επικεντρωθούμε στην υλοποίησή τους στον κώδικα.

a. Η κίνηση του missing person πάνω στο γράφο είναι ανάλογη της κίνησης των People Agent πάνω στο γράφο, με τη διαφορά ότι λόγω σύμβασης η μετακίνηση του γίνεται μόνο με τα πόδια.

b. Στην προσομοίωση μπορεί να εισαχθεί τόσο η αρχική θέση όσο και η αρχική ημερομηνία και ώρα του πειράματος.

Αρχική Θέση:

Ο χρήστης επιλέγει ένα από τα κτήρια του χάρτη ως αρχική θέση μέσω της παραμέτρου “Starting Position” η οποία συνδέεται με την καθολική μεταβλητή “MP_Starting_Pos_name” (αρχικοποιείται στο κενό, δηλαδή παραμένει σε μία τιμή αντίστοιχη με το προγραμματιστικό NULL -nil στην GAML, και παραμένει κενή αν ο χρήστης δεν εισάγει αρχική θέση).

```

parameter "Starting Position" var: MP_Starting_Pos_name category:
"Missing_Person";

```

Αν ο χρήστης εισάγει αρχική θέση, με τη μορφή του ονόματος ενός κτηρίου, αναζητείται η λίστα με τα ονόματα όλων των κτηρίων για να ελεγχθεί εάν το όνομα κτηρίου που δόθηκε είναι έγκυρο και υπάρχει στη λίστα.

Προκειμένου ο χρήστης να εισάγει έγκυρο κτήριο πρέπει, μέσω της επισκόπησης του χάρτη στην αρχική κατάσταση εκτέλεσης κανονικών πειραμάτων, για την οποία θα μιλήσουμε στο Κεφάλαιο 5.2.1, να εντοπίσει στον χάρτη της προσομοίωσης το κτήριο που θέλει να ορίσει ως αρχική θέση και να κάνει δεξί κλικ πάνω σε αυτό ώστε να βρει το όνομα του συγκεκριμένου building agent.

Τα ονόματα των agent instances στη GAMA ορίζονται ως το όνομα της κλάσης στην οποία ανήκουν ακολουθούμενα από έναν αριθμό. Έστω λοιπόν ότι ο χρήστης θέλει να εισάγει ως αρχική θέση του missing person το κτήριο “building237”. Θα το εισάγει στην αντίστοιχη παράμετρο και με την ανανέωση της προσομοίωσης θα ελεγχθεί αν το string “building237” που εισήγαγε ο χρήστης είναι έγκυρο building agent. Το “building237” υποθέτουμε ότι αποτελεί σε αυτή την περίπτωση έγκυρο building agent (υπάρχουν δηλαδή τουλάχιστον 237 κτήρια στο χάρτη) και συνεπώς η εισαγωγή είναι έγκυρη. Ορίζεται τότε ως η αρχική θέση του missing person η τοποθεσία του κτηρίου αυτού, και το χρώμα του κτηρίου αλλάζει σε μπορντό ώστε να εντοπίζεται εύκολα.

Σε περίπτωση που η εισαγωγή δεν είναι έγκυρη, εμφανίζεται η ειδοποίηση “Wrong Input Starting Position” και επιλέγεται τυχαίο κτήριο σαν αρχική θέση.

Το ίδιο, χωρίς την εμφάνιση ειδοποίησης όμως, συμβαίνει και στην περίπτωση που ο χρήστης δεν εισαγάγει αρχική θέση.

```
create missing_person number: nb_missing {
  ...
  if (MP_Starting_Pos_name != nil){
    ask building.population {
      if (name = MP_Starting_Pos_name) {
        MP_Starting_Pos <- location;
        write "Starting Position Coordinates";
        write MP_Starting_Pos;
        myself.living_place <- self;
        myself.input_flag <- true;
        self.color <- #maroon;
      }
    }
  }

  if(!input_flag) {
    write "Wrong Input Starting Position";
    living_place <- one_of(building) ;
    location <- any_location_in (living_place);
  }
  else {
    location <- MP_Starting_Pos;
  }
}
else{
```

```

        living_place <- one_of(building) ;
        location <- any_location_in (living_place);
    }
    objective <- "running";
    ...
}

```

Αρχική Ημερομηνία και Ώρα:

Κατά όμοιον τρόπο με τη λογική του αρχικού σημείου, υποστηρίζεται η δυνατότητα για εισαγωγή της ημερομηνίας και ώρας κατά την οποία χάθηκε ο missing person, μέσω της αντίστοιχης παραμέτρου:

```

parameter "Start Time" category: "Activate Extended Parameters"
var:a_boolean_to_enable_parameters1 enables: [year_missing,
month_missing, day_missing, hour_missing, minute_missing];

//Start Time Activatable Parameters
parameter "Year" var: year_missing category: "Start Time";
parameter "Month" var: month_missing category: "Start Time";
parameter "Day" var: day_missing category: "Start Time";
parameter "Hour" var: hour_missing category: "Start Time";
parameter "Minute" var: minute_missing category: "Start Time";

```

Σε περίπτωση που δεν υπάρχει ενεργοποίηση της παραμέτρου με εισαγωγή στοιχείων από το χρήστη, η προσομοίωση αρχικοποιείται με την ημερομηνία και ώρα του υπολογιστικού συστήματος στο οποίο εκτελείται (δηλαδή πραγματική ημερομηνία και ώρα για υπολογιστή που ρυθμίζει το ρολόι του σύμφωνα με τον παγκόσμιο ιστό).

```

date true_date <- #now;
int year_missing <- true_date.year;
int month_missing <- true_date.month;
int day_missing <- true_date.day;
int hour_missing <- true_date.hour;
int minute_missing <- true_date.minute;
date starting_date <-
date([year_missing,month_missing,day_missing,hour_missing,minute_missing,
0]); // [Year, Month, Day, Hour, Minute, Sec]

```

c. Ο missing person κινείται, όπως και οι People Agents, πάντα με βάση κάποιο προορισμό και ακολουθώντας τη βέλτιστη διαδρομή πάνω στον γράφο κίνησης. Η υλοποίηση αυτής της λειτουργίας είναι παρόμοια με εκείνη που παρουσιάσαμε προηγουμένως. Το πώς επιλέγεται αυτός ο προορισμός παρουσιάζει περισσότερο ενδιαφέρον.

Πιο συγκεκριμένα, ο χρήστης έχει τη δυνατότητα μέσω της αντίστοιχης παραμέτρου να εισαγάγει ένα σημείο ενδιαφέροντος το οποίο ο missing person έχει περισσότερες πιθανότητες να επισκέπτεται με κάποια συχνότητα.

```

parameter "PoI building name" var: Point_of_Interest1_name category:
"Missing_Person";

```

Αν εισαχθεί κάποιο σημείο ενδιαφέροντος, αρχικά ελέγχεται αν είναι έγκυρο, όπως συνέβαινε και με την αρχική θέση. Σε περίπτωση που η εισαγωγή είναι έγκυρη τότε αποθηκεύεται η τοποθεσία του κτηρίου – σημείου ενδιαφέροντος και το κτήριο αυτό αλλάζει χρώμα σε μπλε ώστε να είναι ευδιάκριτο κατά την εκτέλεση του πειράματος.

```
create missing_person number: nb_missing {
  ...
  if(Point_of_Interest1_name != nil) {
    ask building.population {
      if (name = Point_of_Interest1_name) {
        Point_of_Interest1 <- location;
        write Point_of_Interest1;
        myself.input_flag <- true;
        self.color <- #gamablue;
      }
    }
  }
  ...
}
```

Κάθε φορά που επιλέγεται ο επόμενος προορισμός του missing person, δίνεται μία πιθανότητα (ως προεπιλογή ορίζεται ίση με 40%) ο προορισμός αυτός να είναι το σημείο ενδιαφέροντος που εισήχθη, μέσω της πιθανοτικής συνάρτησης “flip()” της GAMA. Η συμπληρωματική πιθανότητα, ορίζει ως νέο προορισμό του missing person ένα τυχαίο κτήριο.

```
reflex run when: objective = "running" and the_target = nil {
  if(Point_of_Interest1 != nil and flip(0.4)){
    the_target <- Point_of_Interest1;
  }
  else {
    the_target <- point(one_of(building));
    // casted one_of(building) to point type!!! one_of(the_graph.vertices);
  }
  arrived <- current_hour;
}
```

Στην περίπτωση που το σημείο ενδιαφέροντος δεν είναι έγκυρο ή δεν δόθηκε καθόλου, ορίζεται ως κενό, και η επιλογή προορισμού για τον missing person γίνεται πάντα τυχαία.

d. Προκειμένου να προσομοιώσουμε την κίνηση του χαμένου παιδιού όσο το δυνατόν πιο ρεαλιστικά, έχουμε πάρει τη σύμβαση ότι ανάμεσα σε δύο διαδρομές του παιδιού θα υπάρχει μία παύση, για ξεκούραση ή για επαναπροσδιορισμό προορισμού.

Ο χρόνος παύσης αυτός έχει μία προκαθορισμένη τιμή:

```
int time_to_rest <- 3;
```

Η τιμή του χρόνου παύσης είναι παραμετροποιήσιμη:

```
parameter "Time for missing person to rest" var: time_to_rest category:
"Missing_Person" ;
```

Η υλοποίηση της παύσης του missing person γίνεται με την αλλαγή του σκοπού του, ο οποίος αντιστοιχεί στην μεταβλητή “objective”, σε ξεκούραση, “resting”. Η αλλαγή αυτή διαρκεί από την χρονική στιγμή που φτάνει στον προηγούμενο προορισμό του μέχρι το τέλος του διαστήματος παύσης που έχει οριστεί.

```
reflex get_some_rest when: objective = "resting" and (current_hour =
(arrived + time_to_rest)mod 24) {
    objective <- "running";
    m_p_resting <- false;
}
```

4.4.4. Κωδικοποίηση Πιθανοτικού Μοντέλου Εύρεσης

a,b,c. Προκειμένου να αναβρεθεί ο missing person στην προσομοίωση πρέπει να συναντηθεί σε μία κοντινή απόσταση με κάποιον People Agent (κατά σύμβαση περίπου 4 πραγματικά μέτρα, 7 στην προσομοίωση λόγω διαφορετικής κλίμακας).

Δεν αρκεί όμως να συναντηθεί σε κοντινή απόσταση με οποιονδήποτε People Agent. Πρέπει ο συγκεκριμένος People Agent που θα συναντήσει να βρίσκεται σε επιφυλακή, δηλαδή να γνωρίζει ότι ο missing person αγνοείται, να θυμάται τα χαρακτηριστικά του και να μπορεί να τον αναγνωρίσει.

Για να προσομοιώσουμε την επιφυλακή μόνο κάποιων εκ των People Agents για την εύρεση του παιδιού δημιουργήσαμε ένα πιθανοτικό μοντέλο εύρεσης στο οποίο όταν ένας People Agent βρίσκεται σε κοντινή απόσταση με τον missing person θα συνειδητοποιεί ότι είναι αγνοούμενος μόνο κάποιες φορές, όσες ορίζονται από τις πιθανότητες του μοντέλου.

Το πιθανοτικό μοντέλο περιλαμβάνει το συνδυασμό επιφυλακής των People Agents, και κατάστασης - συσχέτισης κίνησης μεταξύ People-Agents και missing person (περπάτημα, οδήγηση κτλ.).

Στο πιθανοτικό μοντέλο για τις οποίες υπάρχουν διαφορετικές πιθανότητες με την παράμετρο επιφυλακής να διαπερνά και τις τρεις. Η πρώτη είναι η πιθανότητα ο People Agent να συναντήσει τον missing person ενώ ο People Agent περπατάει, η δεύτερη ενώ ο People Agent μετακινείται με κάποιο όχημα και η τρίτη ενώ ο People Agent βρίσκεται στο εσωτερικό ενός κτηρίου.

Οι πιθανότητες για κάθε σενάριο έχουν προκαθορισμένες τιμές:

```
//variables concerning the probability of finding the missing person when
near them
float proba_find_walking <- 0.4;
float proba_find_driving <- 0.2;
float proba_find_resting <- 0.05;
```

Οι πιθανότητες κάθε σεναρίου είναι παραμετροποιήσιμες από τον χρήστη, με ελάχιστη τιμή 0.01 και μέγιστη τιμή 1, έτσι ώστε να μπορούν να διατηρούν τον ορισμό τους ως πιθανότητες:

```
parameter "Probability of finding ms if walking" var: proba_find_walking
category: "Probabilities" min: 0.01 max: 1.0;
parameter "Probability of finding ms if driving" var: proba_find_driving
category: "Probabilities" min: 0.01 max: 1.0;
parameter "Probability of finding ms while resting" var: proba_find_resting
category: "Probabilities" min: 0.01 max: 1.0;
```

Στη συνέχεια, δημιουργούμε σε κάθε People Agent ένα αντανακλαστικό, με το οποίο σε κάθε στιγμή ελέγχουμε τους agents όλων των ειδών που βρίσκονται κοντά του (“agents_at_distance(7)”). Εάν σε αυτούς περιλαμβάνεται ο missing person, τότε, ανάλογα με το αν ο People Agent αυτός εκείνη τη στιγμή, περπατάει, μετακινείται με όχημα ή βρίσκεται σε κτήριο μετακινούμαστε στο αντίστοιχο σενάριο αν (“if”).

Στο σενάριο που ο πράκτορας περπατάει, η ροή εκτέλεσης μετακινείται στο “if(walking_bool)”, οπότε χρησιμοποιώντας τη συνάρτηση “flip” της GAMA, και αναλόγως με την τιμή της αντίστοιχης πιθανότητας “proba_find_walking”, ο People Agent είτε θα βρεθεί είτε όχι.

Για παράδειγμα αν ένας People Agent περπατάει και βρεθεί σε κοντινή απόσταση με τον missing person με το “proba_find_walking” να έχει την προκαθορισμένη τιμή, δηλαδή 40%, ο People Agent αυτός θα βρει το χαμένο παιδί τέσσερις στις δέκα φορές που θα βρεθεί σε αυτήν την κατάσταση.

Αν το παιδί βρεθεί από τον Agent, αυξάνεται ο γενικός δείκτης εύρεσης του παιδιού “times_found” κατά ένα, και το ίδιο και ο ειδικός δείκτης εύρεσης του παιδιού από People Agent που περπατούσαν, “times_found_walking”. Τους δείκτες αυτούς θα τους συναντήσουμε ξανά στην επόμενη ενότητα Εκτέλεσης και Εποπτείας.

Κατ’ αναλογία ορίζονται και τα άλλα δύο σενάρια.

```
reflex missing_person_nearby when: agents_at_distance(7) contains_any
missing_person {
    if(walking_bool){
        write "Walking and near " + self;
        if(flip(proba_find_walking)){
            times_found <- times_found + 1;
            times_found_walking <- times_found_walking + 1;
            write "Took a walk and stars aligned, FOUND by
" + self + " Times Found " + times_found;
        }
    }
    else if(driving_bool){
        write ("Driving and near" + self);
        if(flip(proba_find_driving)){
            times_found <- times_found + 1;
            times_found_driving <- times_found_driving + 1;
            write "Prayers to driving gods helped, FOUND by
" + self + " Times Found " + times_found;
        }
    }
}
```



```

    }
  }
  else {
    //write "Resting Phase and Near";
    if(flip(proba_find_resting) and m_p_resting = false){
      times_found <- times_found + 1;
      times_found_resting <- times_found_resting + 1;
      write "Quarantine is King, FOUND by " +self +
Times Found " +times_found;
    }
  }
}

```

4.4.5. Στοιχεία Κίνησης

Η κίνηση των People Agent μέσα στην πόλη υποστηρίζει δύο λειτουργίες. Η πρώτη και προκαθορισμένη -που προσιδιάζει στις πραγματικές συνθήκες μετακίνησης των ανθρώπων- ορίζει ότι ένας άνθρωπος μετακινείται με τα πόδια για κάποιες μικρές αποστάσεις και με κάποιο μεταφορικό μέσο για μεγαλύτερες. Η δεύτερη επιτρέπει μεγαλύτερο έλεγχο στο χρήστη, δίνοντάς του τη δυνατότητα να ελέγξει τα ποσοστά των People Agent που περπατούν και εκείνων που κινούνται με κάποιο όχημα.

Οι ανωτέρω λειτουργίες επιτυγχάνονται ως εξής:

Ορίζονται και αρχικοποιούνται οι παρακάτω μεταβλητές:

```

float demographic_driving <- 0.0;
float demographic_walking <- 0.0;
bool demographic_bool;

```

Ορίζονται επίσης οι παράμετροι με την συμπλήρωση των οποίων από τον χρήστη επιλέγεται η κατάλληλη λειτουργία για το πείραμα.

Συγκεκριμένα, αν συμπληρωθεί από το χρήστη κάποια από τις δύο παραμέτρους “Drivers in Area (%) (Fill only one)” ή “Walkers in Area (%) (Fill only one)”, τότε από την προκαθορισμένη λειτουργία κοντινών και μακρινών αποστάσεων μεταφερόμαστε αυτόματα στη λειτουργία στοιχείων κίνησης με τα ποσοστά που ορίζονται από τον χρήστη. Στην περίπτωση που ο χρήστης δεν συμπληρώσει καμία από τις δύο παραμέτρους, παραμένει η προκαθορισμένη λειτουργία κοντινών και μακρινών αποστάσεων.

Στο σενάριο στο οποίο ο χρήστης συμπληρώνει μία εκ των δύο παραμέτρων, ο χρήστης ορίζει μόνο το ένα από τα δύο ποσοστά επί τοις 100 και το άλλο θα ισούται με το υπόλοιπο ποσοστό για συμπλήρωση του 100%. Αν για παράδειγμα το ένα ποσοστό είναι 40%, το άλλο θα ισούται με 60%, δηλαδή το ένα ποσοστό είναι συμπληρωματικό του άλλου. Το ποσοστό που εισάγεται από τον χρήστη αντιστοιχεί είτε στην καθολική μεταβλητή “demographic_driving” είτε στην καθολική μεταβλητή “demographic_walking”. Η τιμή της καθολικής μεταβλητής που αντιστοιχεί στην παράμετρο που ο χρήστης αφήνει κενή προκύπτει συμπληρωματικά με τον τρόπο που προαναφέρθηκε.


```
//Demographic Data Activatable Parameters
parameter "Drivers in Area (%) (Fill only one)" var: demographic_driving
category: "Demographics";
parameter "Walkers in Area (%) (Fill only one)" var: demographic_walking
category: "Demographics";
```

Σε αναλογία με τις παραπάνω επιλογές, κατά τη διάρκεια της αρχικοποίησης της προσομοίωσης, εάν έχει δοθεί ποσοστό κίνησης από τον χρήστη, γίνεται αληθής η λογική μεταβλητή “**demographic_bool**” και αναλόγως με το ποιο ποσοστό κίνησης, αυτό των πεζών ή αυτό των οδηγών έχει δοθεί από τον χρήστη, υπολογίζουμε το συμπληρωματικό του. Σε περίπτωση που δεν έχει δοθεί ποσοστό κίνησης, η λογική μεταβλητή “**demographic_bool**” γίνεται ψευδής έτσι ώστε να λειτουργήσει η προκαθορισμένη λειτουργία αποστάσεων:

```
if(demographic_driving != 0.0 or demographic_walking != 0.0) {
  demographic_bool <- true;
  if(demographic_driving = 0.0) {
    demographic_driving <- 100 - demographic_walking;
  }
  else {
    demographic_walking <- 100 - demographic_driving;
  }
}
else {demographic_bool <- false;}
```

Στη συνέχεια, κατά τη δημιουργία του κάθε People Agent ελέγχεται η “**demographic_bool**”.

Στην περίπτωση που η “**demographic_bool**” είναι αληθής και μόνο τότε, ο agent αυτός κατανέμεται σύμφωνα με τα ποσοστά που δόθηκαν σε μία από τις δύο κατηγορίες «οδηγών» ή «πεζών» χρησιμοποιώντας την γνωστή μας συνάρτηση “**flip**” και τη λογική μεταβλητή “**local_demo_driving**” (αληθής για «οδηγούς», ψευδής για «πεζούς»).

```
create people number: nb_people {
  ...
  if (demographic_bool){
    if(flip(demographic_driving/100)) {
      //write("local demo driving = true");
      local_demo_driving <- true;
    }
    else {
      //write("local demo walking = true");
      local_demo_driving <- false;
    }
  }
  ...
}
```

Τέλος, θα αναφερθούμε σύντομα στα δύο αντανakλαστικά “**reflex time_to_work**” και “**reflex time_to_go_home**”. Τα αντανakλαστικά αυτά ελέγχουν συνεχώς την ώρα.

Μόλις φτάσει η ώρα εκκίνησης εργασίας του People Agent, σύμφωνα με το ωράριό του, ενεργοποιείται το “`reflex time_to_work`” και δημιουργείται νέος προορισμός για τον agent, το κτήριο της εργασίας του. Αντίστοιχα, μόλις το ωράριο του agent φτάσει στο τέλος του, ενεργοποιείται το “`reflex time_to_go_home`”, και ο νέος προορισμός είναι το σπίτι του agent.

Το μόνο που μένει να διευκρινιστεί είναι το πώς θα μετακινηθεί ο People Agent προς το νέο του προορισμό. Αυτό συμβαίνει ως εξής:

Εάν η λειτουργία στοιχείων κίνησης δεν έχει ενεργοποιηθεί μέσω της “`demographic_bool`”, τότε ελέγχεται το μέγεθος της απόστασης. Εάν η απόσταση είναι μικρή (μικρότερη του ενός χιλιομέτρου όπως έχουμε αναφέρει) τότε ο agent θα μετακινηθεί περπατώντας, σε διαφορετική περίπτωση, θα μεταφερθεί με όχημα. Εάν, αντίθετα, η λειτουργία στοιχείων κίνησης έχει ενεργοποιηθεί, η “`demographic_bool`” είναι αληθής, και στην περίπτωση αυτή έχει οριστεί από την αρχικοποίηση του agent το αν θα περπατάει ή όχι, έτσι ώστε το σύνολο των agent να ακολουθεί τα δοθέντα ποσοστά.

```
//this reflex sets the target when it's time to work and changes the
objective of the agent to working
reflex time_to_work when: current_hour = start_work_hour and current_min
= start_work_min and objective = "resting"{
  objective <- "working" ;
  the_target <- work_spot;
  distance <- 1.5 * (living_place distance_to working_place);
  small_distance <- distance < 1 #km;
  if(demographic_bool){
    if(local_demo_driving){driving_bool <- true;}}
    else {walking_bool <- true;}}
  }
  else{
    if(small_distance) {walking_bool <- true;}}
    else {driving_bool<-true;}}
  }
}

//this reflex sets the target when it's time to go home and
changes the objective of the agent to resting
reflex time_to_go_home when: current_hour = end_work_hour and
current_min = end_work_min and objective = "working"{
  objective <- "resting" ;
  the_target <- home_spot;
  distance <- 1.5 * (living_place distance_to working_place);
  small_distance <- distance < 1 #km;
  if(demographic_bool){
    if(local_demo_driving){driving_bool <- true;}}
    else {walking_bool <- true;}}
  }
  else{
    if(small_distance) {walking_bool <- true;}}
    else {driving_bool<-true;}}
  }
}
```

4.4.6. Κωδικοποίηση Λειτουργιών Εκτέλεσης

a. Η κανονική λειτουργία εκτέλεσης της προσομοίωσης κωδικοποιείται ως πείραμα τύπου γραφικού περιβάλλοντος (gui). Σε αυτό το σημείο δηλώνεται το σύνολο των παραμέτρων και οι καθολικές μεταβλητές με τις οποίες συνδέονται. Δημιουργείται έτσι το πάνελ παραμέτρων. Ορίζονται επίσης οι οθόνες της προσομοίωσης “`chart_display`” και “`city_display`”, και το περιεχόμενο το οποίο απεικονίζει η κάθε μία από αυτές.

Στην “`city_display`” γίνονται ορατά στην οθόνη της προσομοίωσης όλα τα instances των building, road και People agents καθώς και το μοναδικό instance της κλάσης missing person, το χαμένο παιδί (κατά σύμβαση ο missing person). Με την εμφάνιση των agents αυτών στην οθόνη, ο χρήστης αποκτάει εποπτεία του πειράματος. Βλέπει τον χάρτη που σχηματίζεται από τα κτήρια και τους δρόμους καθώς και όλες τις κινήσεις των People Agents και του missing person πάνω στον χάρτη αυτόν.

Ο έλεγχος της ροής της προσομοίωσης γίνεται από τον χρήστη μέσω του πάνελ ελέγχου ροής εκτέλεσης προσομοίωσης για πειράματα gui της πλατφόρμας GAMA. Η αλλαγή των τιμών των παραμέτρων γίνεται σε γραφικό περιβάλλον από το χρήστη χρησιμοποιώντας το πάνελ παραμέτρων. Τα δύο αυτά πάνελ θα αναλυθούν εκτενέστερα στο Κεφάλαιο 5.2.

Στην “`chart_display`” δημιουργούνται δύο ειδών διαγράμματα και συνδέονται με τις κατάλληλες μεταβλητές απεικόνισης. Μέσω της “`chart_display`” οπτικοποιούνται σημαντικές πληροφορίες για την εκτέλεση του πειράματος κανονικής λειτουργίας όπως το πόσοι άνθρωποι δουλεύουν ή είναι σπίτι για κάθε χρονική στιγμή, ή το πόσες φορές μέχρι εκείνη τη χρονική στιγμή έχει βρεθεί το χαμένο παιδί στο συγκεκριμένο πείραμα.

Οι δύο οθόνες της προσομοίωσης αναλύονται περαιτέρω στο Κεφάλαιο 5.2.1. και δίνονται παραδείγματα πειραμάτων κανονικής λειτουργίας στα οποία προσφέρουν χρήσιμη εποπτεία.

```
experiment find_missing_person type: gui {
    parameter "Simulation Map (type: .osm)" var: osmfile category:
        "GIS";

    //Determines number of people agents in simulation using globla
    var nb_people
    parameter "Number of people agents" var: nb_people category: "GIS";
    ...
    output {

        display chart_display refresh:every(1#cycles) {
            chart "People Status" type: pie style: exploded size: {1,
                0.5} position: {0, 0.5}{
                data "Working" value: people count
                    (each.objective="working") color: #magenta ;
                data "Resting" value: people count
                    (each.objective="resting") color: #blue ;
```

```

    }
    chart "Finding Missing Person" type: series size: {1, 0.5}
    position: {0,0} {
    data "Times missing person was found" value: times_found
    color: #red;
    data "Times missing person was close to being found"
    value: close_call color: #green;
    }

    }
    display city_display type: opengl {

    species building aspect: base; //refresh: false;
    species road aspect: base; // refresh: false;
    species missing_person aspect: base ;
    species people aspect: base;

    }
    ...
}

```

b. Για την υλοποίηση των δύο διαφορετικών batch πειραμάτων ορίζουμε δύο διαφορετικά experiment τύπου Batch (“**type: batch**”), το “**Batch_Optimization_No_Times_Found**” και το “**Batch_Optimization_First_Time**”.

Για το κάθε ένα από τα δύο πειράματα ορίζεται ένας στόχος. Ο στόχος αυτός είναι μία συνθήκη η οποία μόλις ικανοποιηθεί από την συγκεκριμένη επανάληψη του πειράματος η επανάληψη αυτή θα τερματιστεί, θα αλλάξει η τιμή της παραμέτρου που εξερευνάται και θα ξεκινήσει η επόμενη επανάληψη. Ο στόχος αυτός ορίζεται με την εντολή “**until:**”.

Σύντομη αναφορά αξίζει να γίνει και στις εντολές “**batch repeat: 2 keep_seed: true**” μέσω των οποίων εκτελούμε την κάθε επανάληψη του πειράματος δύο φορές διατηρώντας όλες τις παραμέτρους και την ίδια γεννήτρια τυχαιότητας (γεννήτρια παραγωγής τυχαίων αριθμών που χρησιμοποιούνται σε διάφορες εσωτερικές λειτουργίες της προσομοίωσης) προκειμένου τα αποτελέσματα που εξάγονται από το πείραμά μας να έχουν την μέγιστη δυνατή στατιστική αξιοπιστία.

Στο κύριο μέρος κωδικοποίησης ορίζεται η παράμετρος που εξερευνάται η οποία και στα δύο δικά μας πειράματα batch είναι η “**Number of People in Area**”, ο αριθμός δηλαδή των People Agents στην προσομοίωση. Ορίζεται μία κατώτερη τιμή των ανθρώπων στην προσομοίωση, μία ανώτερη και το βήμα αύξησης σε κάθε επανάληψη.

Το κατηγορημα “**method**” συμπληρώνεται με τη μέθοδο βελτιστοποίησης που θα χρησιμοποιήσει το συγκεκριμένο batch πείραμα. Και στα δύο δικά μας πειράματα θα χρησιμοποιήσουμε την εξαντλητική - “**exhaustive**” μέθοδο εξερεύνησης στην οποία εξερευνούνται όλες οι τιμές της παραμέτρου, ξεκινώντας από την ελάχιστη και αυξάνοντας σύμφωνα με την τιμή του βήματος, μέχρι να φτάσουμε την μέγιστη. Η GAMA υποστηρίζει ένα σύνολο μεθόδων εξερεύνησης. Περισσότερες πληροφορίες

για το σύνολο των μεθόδων εξερεύνησης που υποστηρίζει η GAMA μπορούν να βρεθούν [εδώ](#)¹².

Τέλος, μέσω του αντανakλαστικού “`save_results_explo`”, τα δύο σύνολα αποτελεσμάτων της κάθε επανάληψης (όπως ορίσαμε παραπάνω τρέχουμε κάθε επανάληψη δύο φορές με τις ίδιες παραμέτρους) εξάγονται σε ένα αρχείο .csv στις κατάλληλες στήλες προκειμένου να μπορούν να αναλυθούν και να χρησιμοποιηθούν περαιτέρω από τον χρήστη.

```
experiment Batch_Optimization_No_Times_Found type: batch repeat: 2
    keep_seed: true until: ( (time / #day) > 4) {
        parameter "Number of People in Area" var: nb_people min:800
            max:1000 step: 20;
        method exhaustive maximize: times_found;
        //method tabu maximize: times_found iter_max: 10 tabu_list_size: 3;

        reflex save_results_explo {
            ask simulations {
                save [int(self),nb_people, self.times_found,
                    self.times_found_walking, self.times_found_driving,
                    self.times_found_resting]
                to: "../results_no_times.csv" type: "csv" rewrite: (
                    int(self) = 0) ? true : false header: true;
            }
        }
    }

experiment Batch_Optimization_First_Time type: batch keep_seed: true
until: ( times_found = 1 ) {
    parameter "Number of People in Area" var: nb_people min:800
        max:1000 step: 20;

    //method exhaustive maximize: times_found;
    method tabu maximize: times_found iter_max: 10 tabu_list_size: 3;

    reflex save_results_explo {
        ask simulations {
            save [int(self),nb_people, (time / #day), (time / #minute) ]
                to: "../results_first_time_mp.csv" type: "csv"
                rewrite: (int(self) = 0) ? true : false header: true;
        }
    }
}
```

¹² <https://gama-platform.github.io/wiki/ExplorationMethods>

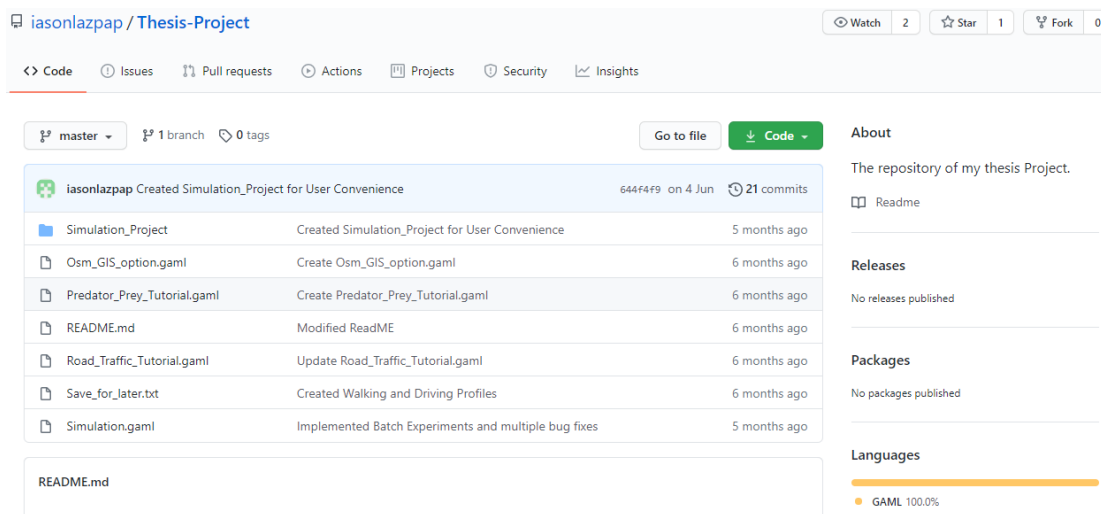
Κεφάλαιο 5. Εκτέλεση και Εποπτεία της Προσομοίωσης

5.1. Φόρτωση Προσομοίωσης στην GAMA

Το πρώτο βήμα για την εκτέλεση της προσομοίωσης είναι η φόρτωσή της στην GAMA.

Αυτή θα πραγματοποιηθεί με την μεταφόρτωση του φακέλου της προσομοίωσης (Thesis-Project) από το Github repository της, το οποίο βρίσκεται [εδώ](https://github.com/iasonlazpap/Thesis-Project)¹³.

Ακολουθώντας τον υπερσύνδεσμο φτάνουμε στην σελίδα περιεχομένων του Thesis-Project (Εικόνα 5.1.).



Εικόνα 5.1. Το Github repository με το σύνολο των αρχείων της προσομοίωσης.

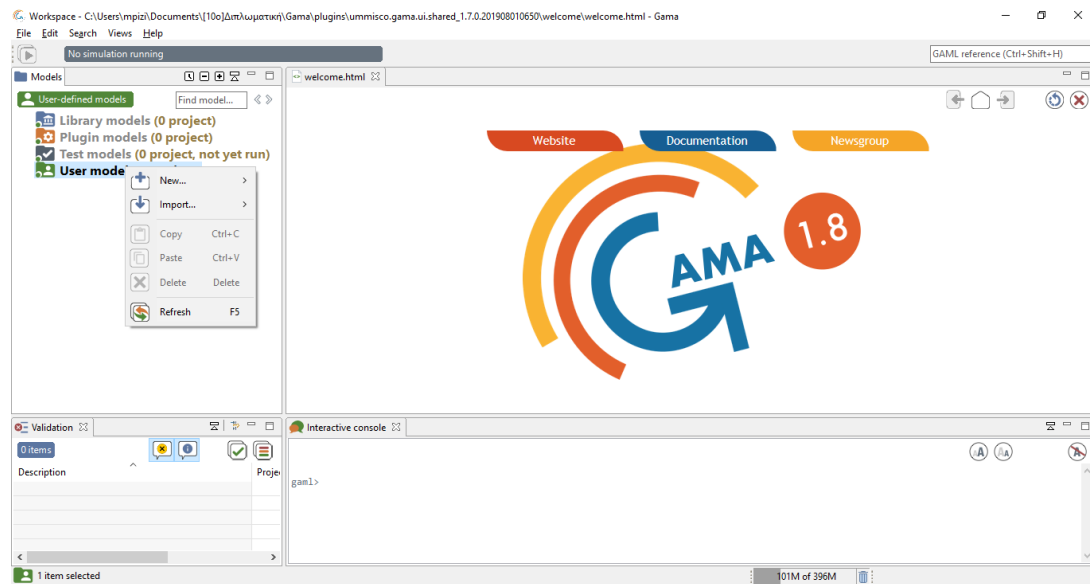
Πηγή: <https://github.com/iasonlazpap/Thesis-Project>

Εκεί, επιλέγουμε το πράσινο κουτί “Code” και “Download ZIP”. Έτσι, μεταφορτώνουμε στον υπολογιστή μας το “Thesis-Project-master” και το αποσυμπιέζουμε. Τώρα έχουμε αποθηκευμένο τοπικά τον φάκελο ο οποίος περιέχει το “Simulation_Project”, το GAMA Project το οποίο μπορούμε να εισάγουμε αυτούσιο στην GAMA.

Στη συνέχεια, έχοντας ολοκληρώσει την εγκατάσταση της GAMA στο κεφάλαιο 3.4., την εκκινούμε.

Εντοπίζουμε το παράθυρο εξερεύνησης μοντέλων στην GAMA και πατάμε **δεξί κλικ** πάνω στο **User Models** (Εικόνα 5.2.).

¹³ <https://github.com/iasonlazpap/Thesis-Project>

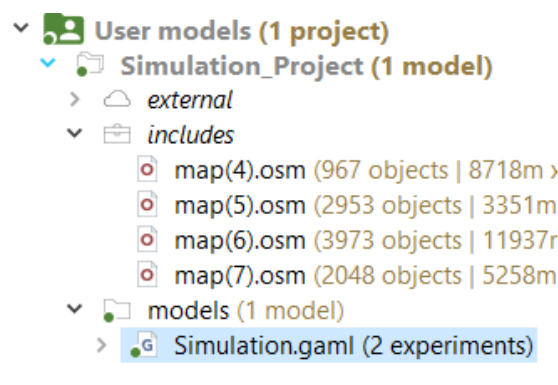


Εικόνα 5.2. Η αρχική της πλατφόρμας GAMA και στα αριστερά, εντός του πλαισίου μοντέλων, ο φάκελος User Models.

Επιλέγουμε **Import** και **Gama Project**. Στο αναδυόμενο παράθυρο που εμφανίζεται επιλέγουμε **“Browse”**.

Εντοπίζουμε τον φάκελο που περιέχει το “Simulation_Project”. Το επιλέγουμε και ολοκληρώνουμε πατώντας **Finish**.

Τώρα, στο παράθυρο εξερεύνησης μοντέλων της GAMA, κάτω από το User Models εμφανίζεται ο φάκελος “Simulation_Project”. Αναπτύσσοντας τα περιεχόμενα του φακέλου, μέσα στον υποφάκελο “models”, βρίσκουμε το αρχείο **“Simulation.gaml”** (Εικόνα 5.3.).



Εικόνα 5.3. Η θέση του Simulation.gaml που περιέχει τον κώδικα της προσομοίωσης, μέσα στο Simulation_Project.

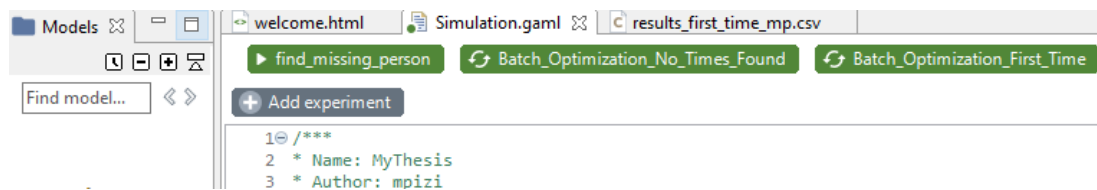
Με **διπλό κλικ** στο “Simulation.gaml”, ολοκληρώνεται η εισαγωγή της προσομοίωσής μας στην GAMA.

5.2. Εκτέλεση της Προσομοίωσης

Έχοντας ολοκληρώσει τη διαδικασία φόρτωσης της προσομοίωσης στην GAMA μπορούμε να προχωρήσουμε στην εκτέλεσή της.

Όπως αναλύσαμε προηγουμένως, η προσομοίωσή υποστηρίζει δύο λειτουργίες εκτέλεσης. Το πείραμα κανονικής λειτουργίας ονομάζεται **“find_missing_person”**, ενώ τα δύο batch πειράματα που δημιουργήσαμε ονομάζονται **“Batch_Optimization_No_Times_found”** και **“Batch_Optimization_First_Time”**.

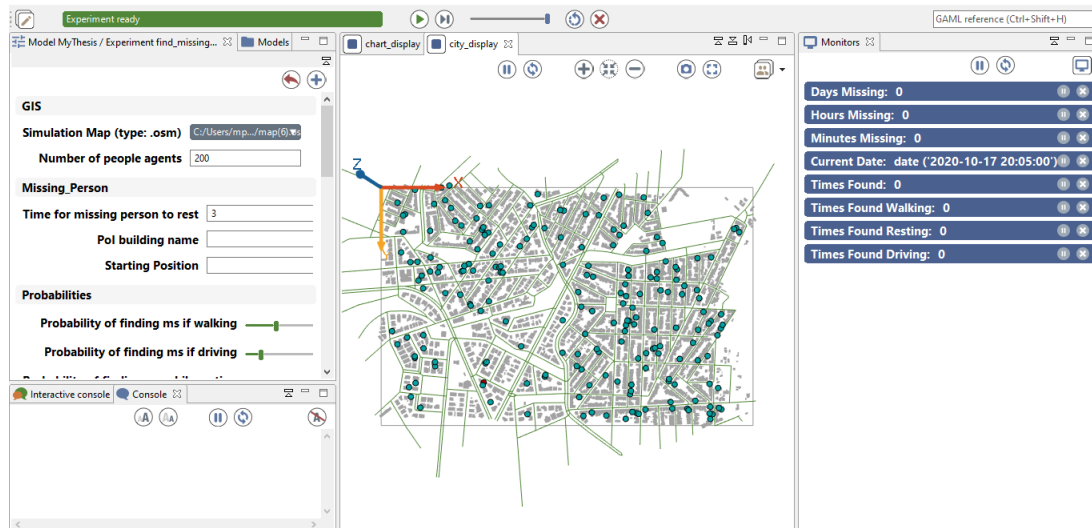
Η περιοχή εκτέλεσης βρίσκεται πάνω από τον κώδικα, με τα ονόματα των δύο ειδών πειραμάτων εντός πράσινων πλαισίων προς επιλογή (Εικόνα 5.4.).



Εικόνα 5.4. Τα πλαίσια για εκκίνηση εκτέλεσης “find_missing_person”, “Batch_Optimization_No_Times_found” και “Batch_Optimization_First_Time”.

5.2.1. Πείραμα Κανονικής Λειτουργίας και Εποπτεία

Με διπλό κλικ πάνω στο “find_missing_person” μετακινούμαστε στην λειτουργία εκτέλεσης κανονικού πειράματος (Εικόνα 5.5.).



Εικόνα 5.5. Η εικόνα ενός κανονικού πειράματος με τον προκαθορισμένο χάρτη map(6).osm που απεικονίζει μία γειτονιά στο Σύνταγμα της Αθήνας.

Παρατηρούμε ότι στην κορυφή της οθόνης βρίσκεται οριζόντια το πάνελ ελέγχου ροής (Εικόνα 5.6).



Εικόνα 5.6. Πάνελ ελέγχου ροής προσομοίωσης.

Στην Εικόνα 5.7. βλέπουμε τα πλήκτρα που υπάρχουν στο πάνελ ελέγχου ροής μαζί με μία σύντομη περιγραφή της λειτουργίας τους.

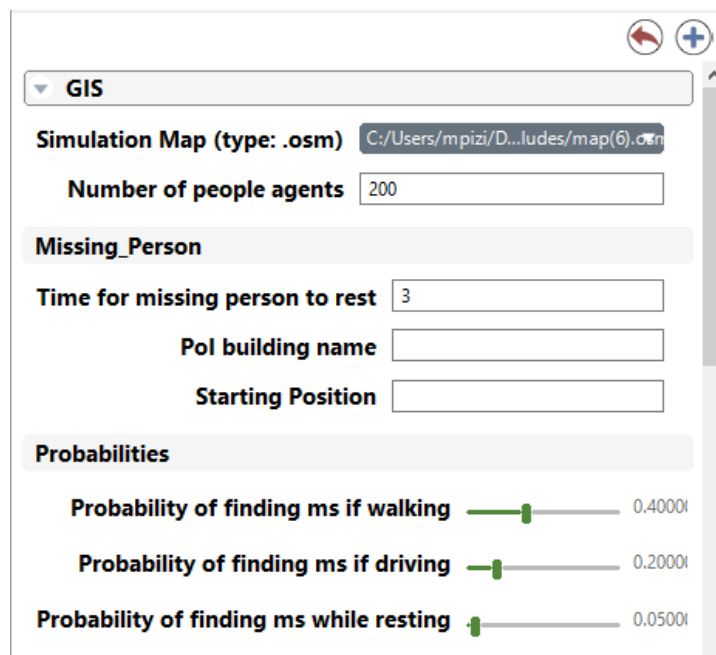


Εικόνα 5.7. Από αριστερά προς τα δεξιά, το πλήκτρο εκκίνησης της προσομοίωσης, το πλήκτρο επόμενου βήματος, η ρύθμιση της ταχύτητας της προσομοίωσης, το πλήκτρο ανανέωσης παραμέτρων και επανεκκίνησης πειράματος και η έξοδος από την λειτουργία εκτέλεσης.

Κάτω από το οριζόντιο πάνελ ελέγχου ροής, η οθόνη χωρίζεται σε 3 μέρη (από αριστερά προς τα δεξιά):

1. Πάνελ Παραμέτρων

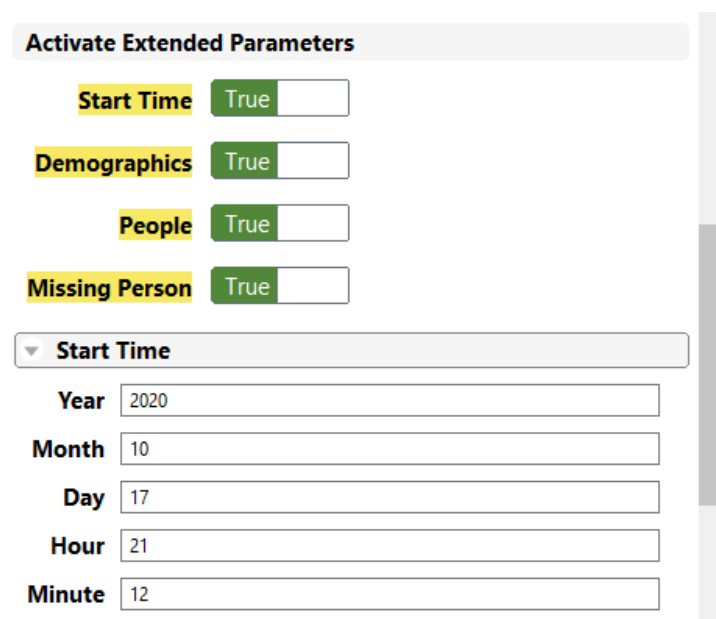
Το πάνελ παραμέτρων είναι το γραφικό περιβάλλον στο οποίο ο χρήστης μπορεί να αλλάξει τις τιμές των παραμέτρων, και αντιστοίχως τις καθολικές μεταβλητές με τις οποίες αυτές συνδέονται, όπως αναλύθηκε προηγουμένως. Βλέπουμε όλες τις διαθέσιμες παραμέτρους στις Εικόνες 5.8., 5.9. και 5.10.



The screenshot shows a software interface with three main sections:

- GIS**: Contains a dropdown menu, a text field for "Simulation Map (type: .osm)" with the path "C:/Users/mpizi/D...ludes/map(6).osm", a text field for "Number of people agents" with the value "200", and a section for "Missing_Person" with fields for "Time for missing person to rest" (value 3), "Pol building name", and "Starting Position".
- Probabilities**: Contains three sliders for finding a missing person (ms) in different states:
 - "Probability of finding ms if walking" with a value of 0.40000.
 - "Probability of finding ms if driving" with a value of 0.20000.
 - "Probability of finding ms while resting" with a value of 0.05000.

Εικόνα 5.8. Πρώτο μέρος του πάνελ παραμέτρων που απεικονίζει τις παραμέτρους των κατηγοριών GIS, Missing_Person και Probabilities.



The screenshot shows the second part of the parameter panel with two main sections:

- Activate Extended Parameters**: A section with four toggle switches, all set to "True":
 - Start Time
 - Demographics
 - People
 - Missing Person
- Start Time**: A section with five text fields for specifying the time:
 - Year: 2020
 - Month: 10
 - Day: 17
 - Hour: 21
 - Minute: 12

Εικόνα 5.9. Δεύτερο μέρος του πάνελ παραμέτρων που απεικονίζει τις παραμέτρους των κατηγοριών Activate Extended Parameters (ενεργοποιεί με επιλογή του χρήστη τις κατηγορίες που ακολουθούν), και Start Time.

Demographics

Drivers in Area (%) (Fill only one)

Walkers in Area (%) (Fill only one)

People

Earliest hour to start work 7 [2..8] every 0.5

Latest hour to start work 09 [8..12] every 1

Earliest hour to end work 16 [12..16] every 1

Latest hour to end work 18 [16..23] every 1

minimum speed

maximum speed

Missing_Person Ext

minimum speed for missing person

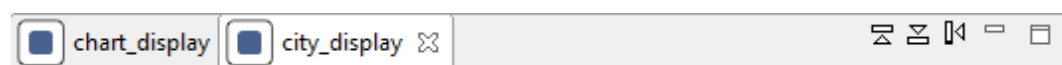
maximum speed for missing person

Εικόνα 5.10. Τρίτο μέρος του πάνελ παραμέτρων που απεικονίζει τις παραμέτρους των κατηγοριών Demographics, People και Missing_Person Ext.

Σημείωση: Προκειμένου οι αλλαγές που κάνουμε σε μία παράμετρο ή σε έναν συνδυασμό παραμέτρων να αλλάξουν και στο πείραμα, απαιτείται η ανανέωση και εκκίνηση του από την αρχή. Αυτό πραγματοποιείται με το πλήκτρο ανανέωσης του πάνελ ελέγχου ροής (Εικόνα 5.7.).

2. Οθόνες Προσομοίωσης (displays)

Στην εκτέλεση του πειράματος κανονικής λειτουργίας έχουμε την επιλογή για την εναλλαγή ανάμεσα σε δύο οθόνες πληροφοριών (Εικόνα 5.11.).



Εικόνα 5.11. Οι επιλογή για εναλλαγή ανάμεσα στις δύο οθόνες της κανονικής λειτουργίας εκτέλεσης της προσομοίωσης.

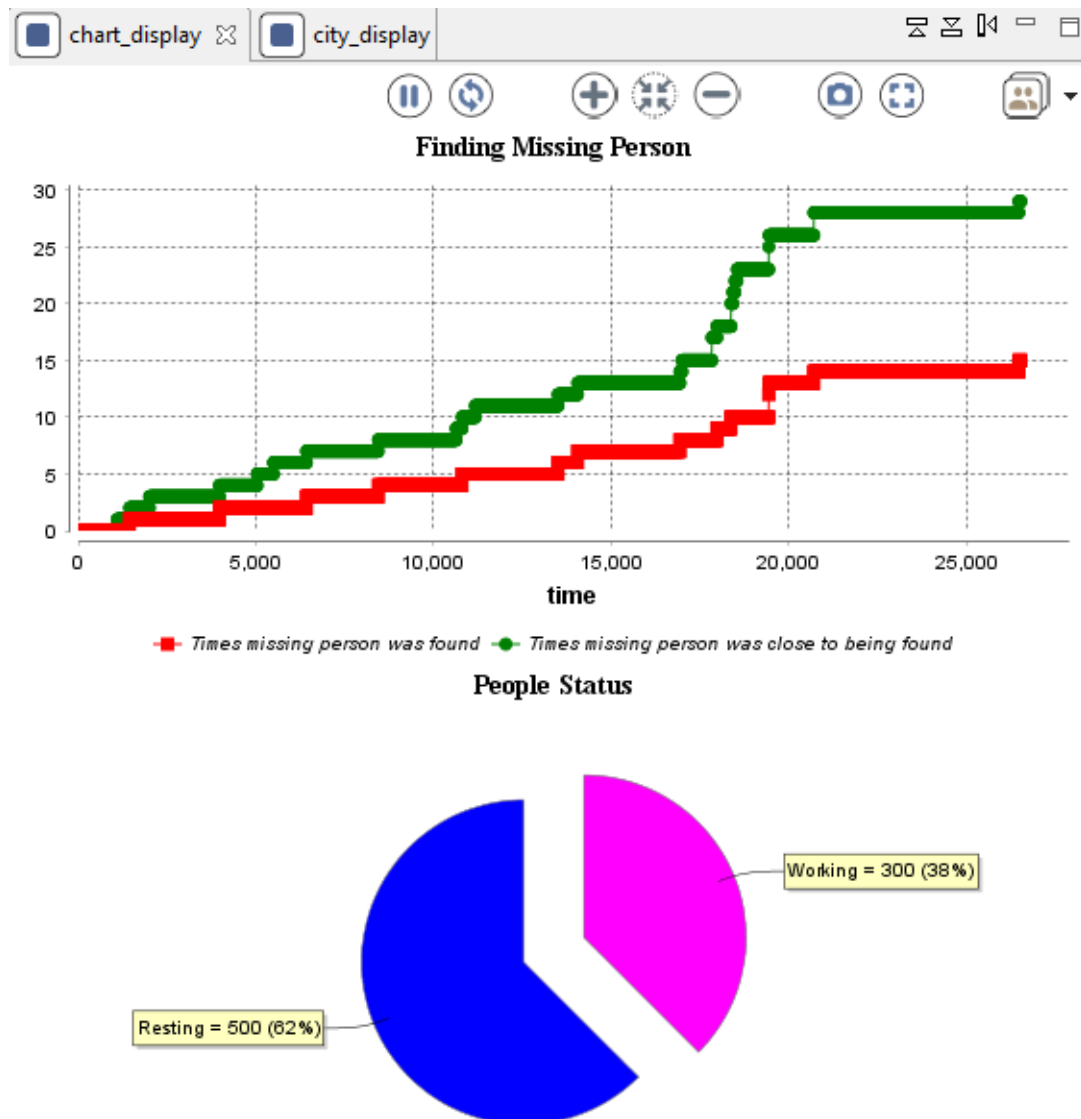
Η πρώτη οθόνη είναι η “**chart_display**” η οποία απεικονίζει χρήσιμες πληροφορίες για την εκτέλεση της προσομοίωσης με τη μορφή διαγραμμάτων.

Η GAMA έχει τη δυνατότητα για προβολή ενός συνόλου διαγραμμάτων. Για τους σκοπούς της εποπτείας της προσομοίωσής μας έχουν επιλεγεί δύο διαγράμματα διαφορετικών ειδών.

Το πρώτο διάγραμμα της “**chart_display**” απεικονίζει ταυτόχρονα το πόσες φορές κάποιος People Agent ήταν κοντά στο να βρει τον missing person

(βρίσκονταν σε απόσταση μικρότερη ή ίση των 4 μέτρων) και το πόσες φορές έχει εντοπιστεί ο missing person, σε σχέση με τον χρόνο (σε συνολικά λεπτά) για τον οποίο τρέχει η προσομοίωση (Εικόνα 5.12.).

Το δεύτερο διάγραμμα της “**chart_display**” είναι διάγραμμα πίτας και απεικονίζει το ποσοστό των People Agents που βρίσκονται στην εργασία τους και το ποσοστό των People Agents που βρίσκονται στο σπίτι τους σε πραγματικό χρόνο πειράματος (Εικόνα 5.12.).



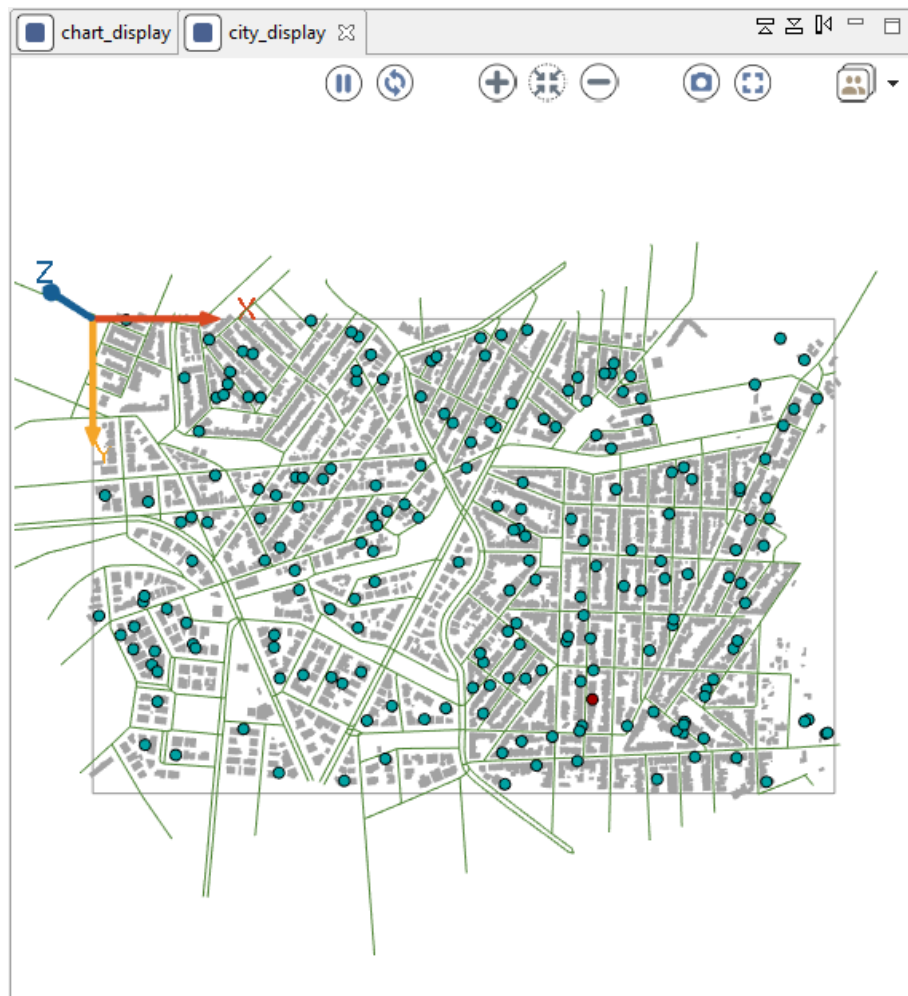
Εικόνα 5.12. Εποπτεία της οθόνης “chart_display” κατά τη διάρκεια εκτέλεσης ενός πειράματος.

Η δεύτερη οθόνη είναι η “**city_display**”.

Η “city_display” απεικονίζει τον χάρτη της προσομοίωσης πάνω στον οποίο βλέπουμε με διαφορετικό χρώμα όλους τους agents: τα κτήρια (γκρι), τους

δρόμους (απαλό πράσινο), τους People Agents (τιρκουάζ) και τον missing person (μπορντό).

Μετά την εκκίνηση της προσομοίωσης βλέπουμε επίσης στον χρόνο της προσομοίωσης (όπως έχει προαναφερθεί, 1 βήμα ή ένας κύκλος της προσομοίωσης, ισούται με 1 λεπτό της ώρας) την κίνηση των People Agents και του missing person πάνω στο χάρτη (Εικόνα 5.13.).



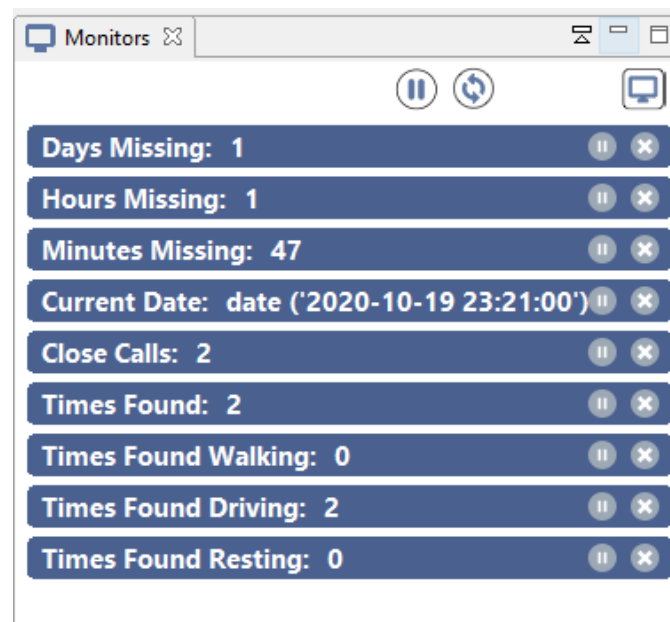
Εικόνα 5.13. Η οθόνη “city_display” κατά την εκτέλεση ενός πειράματος σε περιοχή του Συντάγματος στην Αθήνα.

3. Παρακολούθηση Μεταβλητών (monitors)

Στο πάνελ παρακολούθησης μεταβλητών ο χρήστης έχει τη δυνατότητα να παρακολουθεί την τιμή μερικών σημαντικών για την ροή εκτέλεσης μεταβλητών.

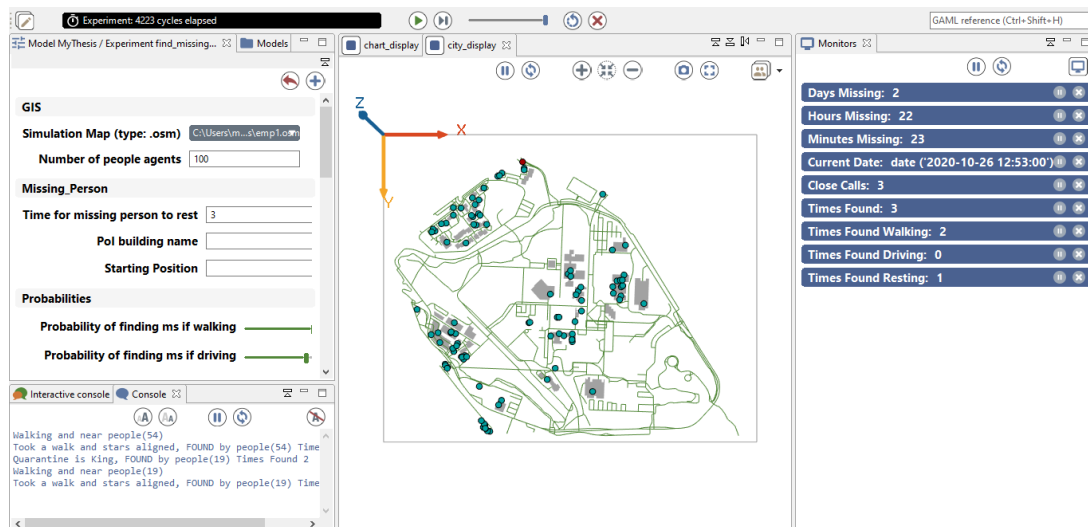
Στην προσομοίωσή μας, στο πάνελ παρακολούθησης μεταβλητών έχουν εισαχθεί μεταβλητές που σχετίζονται με τον χρόνο της προσομοίωσης, όπως η τρέχουσα ημερομηνία και ώρα της προσομοίωσης, ο χρόνος σε ημέρες, ώρες σε ακέραιο αριθμό και πρόσθετα λεπτά (π.χ. 1 ημέρα, 1 ώρα και 47 λεπτά στην

Εικόνα 5.14.) κατά τον οποίο αγνοείται ο missing person, καθώς και μεταβλητές σχετικές με την εύρεσή του, όπως το πόσες φορές ήταν κοντά στο να βρεθεί – Close Calls, δηλαδή κάποιος People Agent πέρασε από απόσταση μικρότερη των 4 μέτρων από δίπλα του (συμπεριλαμβανομένων των φορών που τελικά βρέθηκε), το πόσες φορές εντοπίστηκε συνολικά – Times Found (δηλαδή βρέθηκε κοντά σε κάποιον People Agent ο οποίος ήταν σε επιφυλακή για την εύρεσή του, κάτι που προσομοιώνεται από το πιθανοτικό μοντέλο) αλλά και ειδικότερα, το πόσες φορές εντοπίστηκε ενώ ο People Agent που τον εντόπισε περπατούσε – Times Found Walking, οδηγούσε – Times Found Driving, ή βρισκόταν μέσα σε κτήριο -Times Found Resting (Εικόνα 5.14.).

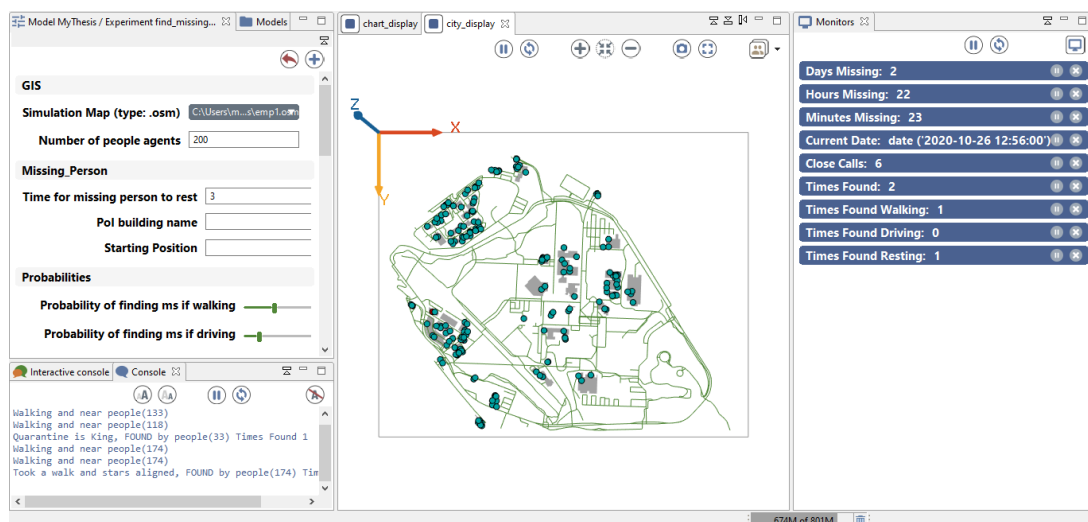


Εικόνα 5.14. Εποπτεία του πάνελ παρακολούθησης μεταβλητών κατά τη διάρκεια πειράματος.

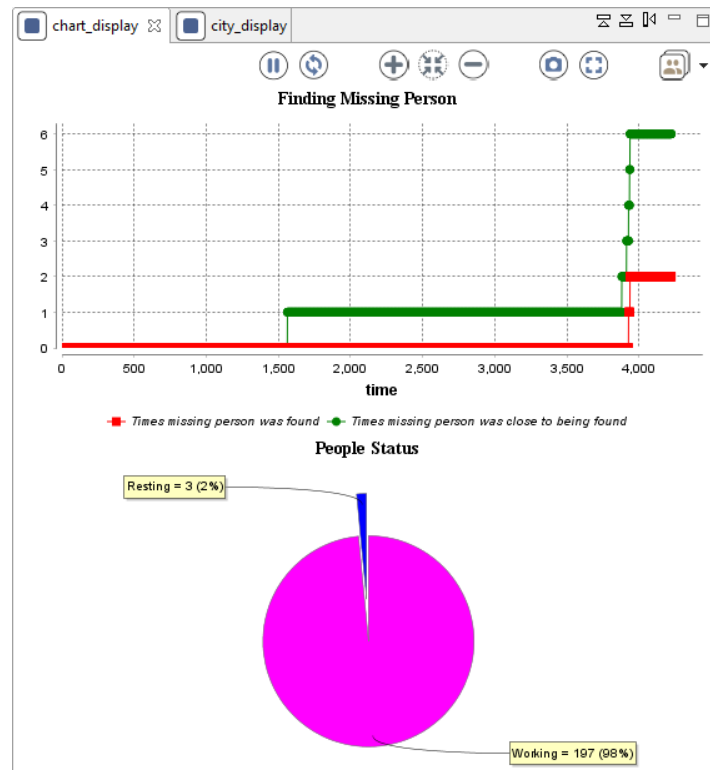
Παρακάτω (Εικόνες 5.15. – 5.20) δίνονται κάποια ακόμη παραδείγματα-πειράματα προσομοίωσης με χρήση διαφορετικών χαρτών που ανήκουν σε διαφορετικά μέρη της Αθήνας και της Θεσσαλονίκης και διαφορετικών παραμέτρων, για να αναδειχθούν οι διαφορετικές δυνατότητες παραμετροποίησης.



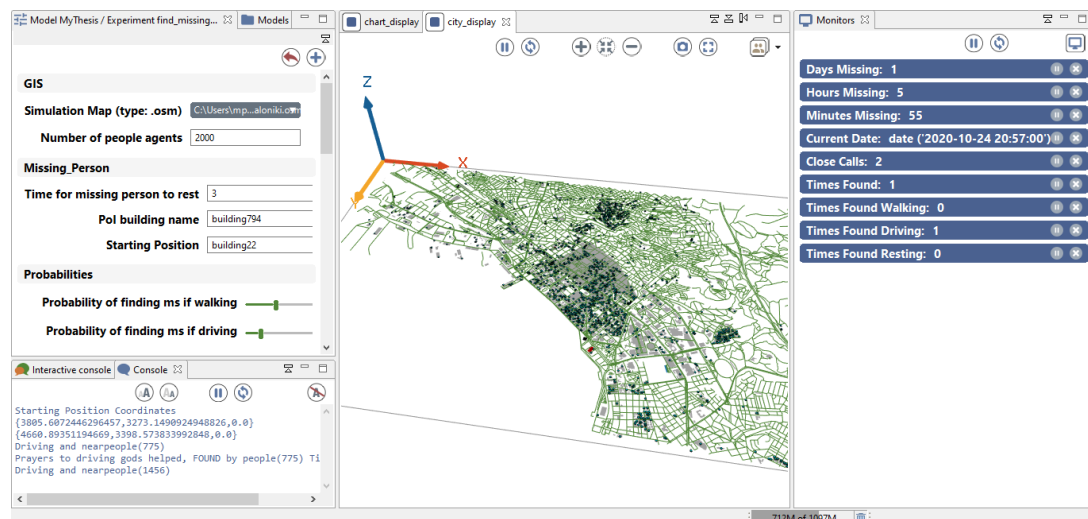
Εικόνα 5.15. Εκτέλεση πειράματος κανονικής λειτουργίας στο χώρο του Πολυτεχνείου σε μία μέρα χαμηλής προσέλευσης (100 άνθρωποι κινούνται στην Πολυτεχνειούπολη), αλλά με την πλειοψηφία των φοιτητών και καθηγητών σε επιφυλακή για την εύρεση του παιδιού. Παρατηρούμε τα πολύ καλά αποτελέσματα εύρεσης του χαμένου παιδιού, για διάστημα αναζήτησής του 2 μέρες, 22 ώρες και 23 λεπτά, καθώς και την ταύτιση των close calls με τις φορές που το παιδί βρέθηκε λόγω της υψηλής επιφυλακής.



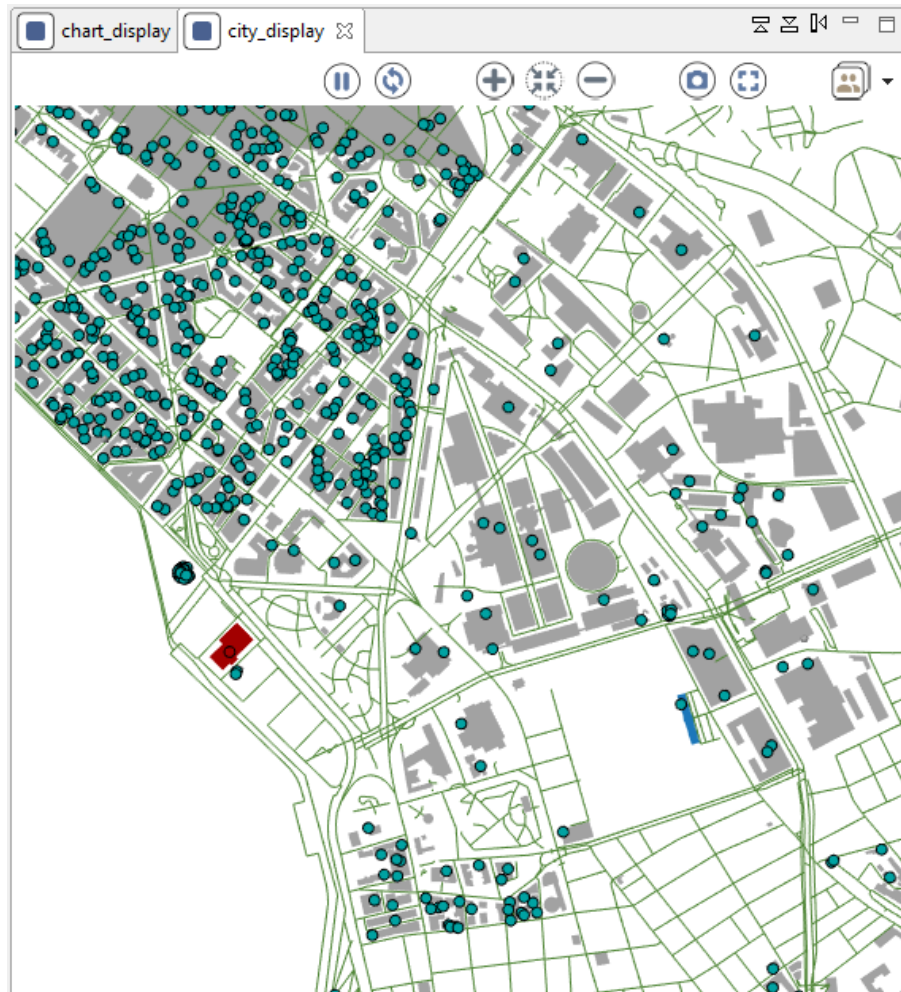
Εικόνα 5.16. Εκτέλεση πειράματος στον ίδιο χάρτη της Πολυτεχνειούπολης με πριν, αλλά με διπλάσιο πληθυσμό People Agents και σημαντικά μικρότερα ποσοστά επιφυλακής. Στο ίδιο διάστημα αναζήτησης 2 μερών, 22 ωρών και 23 λεπτών με πριν παρατηρούμε χειρότερα αποτελέσματα με πολλά περισσότερα close calls.



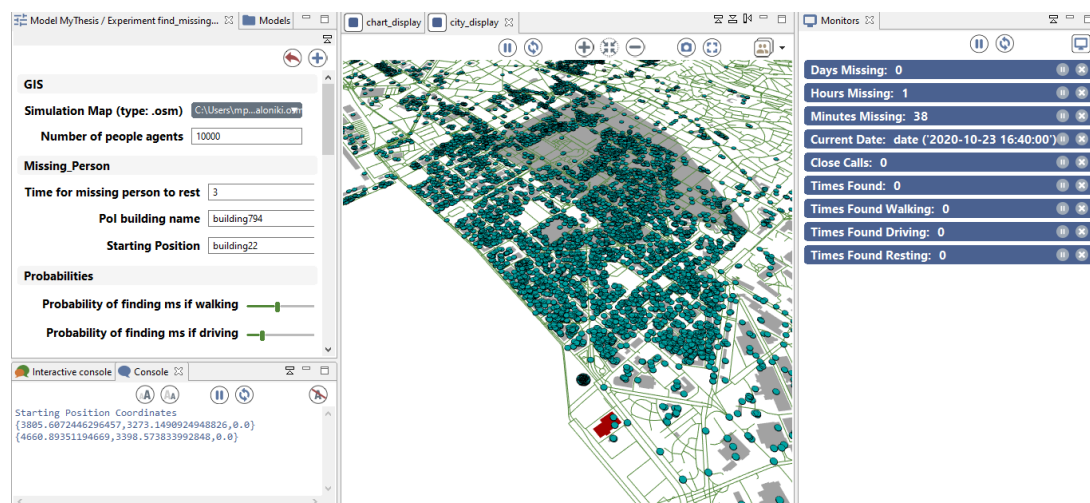
Εικόνα 5.17. Η “chart_display” με τα διαγράμματα του πειράματος της Εικόνας 5.16.



Εικόνα 5.18. Εξαγωγή από το Extract.BBBike προσαρμοσμένου χάρτη του κέντρου της Θεσσαλονίκης (περιοχή 13 τετραγωνικών χλμ. από το λιμάνι της πόλης μέχρι τον Λευκό Πύργο) και εισαγωγή του στην προσομοίωση. Ο πληθυσμός των ανθρώπων είναι μικρός, στα 2000 άτομα, παρ’ όλα αυτά, λόγω των κεντρικών σημείων της αρχικής θέσης και του σημείου ενδιαφέροντος που εισαγάγαμε για τον missing person (Εικόνα 5.19.), σε 1 μέρα, 5 ώρες και 55 λεπτά προσομοίωσης ήταν κοντά στον να βρεθεί 2 φορές και μία από αυτές η ανεύρεσή του ήταν επιτυχής.



Εικόνα 5.19. Τομή της προσομοίωσης πριν αυτή ξεκινήσει με ορισμένο από τον χρήστη αρχικό σημείο για τον missing person (Βασιλικό Θέατρο, πλησίον του Λευκού Πυργου) σε μπορντό και σημείο ενδιαφέροντος (Πάρκο του Πεδίου του Άρεως) σε μπλέ.



Εικόνα 5.20. Το πείραμα της Θεσσαλονίκης με πληθυσμό People Agents 10.000. Παρατηρούμε ότι όσο αυξάνεται το μέγεθος της περιοχής μοντελοποίησης και ο

πληθυσμός των ανθρώπων που κινούνται σε αυτήν, αυξάνονται οι υπολογιστικοί πόροι που απαιτούνται για την μοντελοποίηση.

5.2.2. Πείραμα Εκτέλεσης Πολλαπλών Προσομοιώσεων (Batch Experiment)

Η λειτουργία των batch πειραμάτων στην GAMA είναι περισσότερο πολύπλοκη από την λειτουργία πειραμάτων κανονικής λειτουργίας, και απευθύνεται σε προχωρημένους χρήστες.

Αυτό συμβαίνει διότι, σε αντίθεση με την λειτουργία πειραμάτων κανονικής λειτουργίας η οποία επιτρέπει στον χρήστη να προσαρμόζει τις παραμέτρους γραφικά από το πάνελ παραμέτρων χωρίς να χρειαστεί να τροποποιήσουν τον κώδικα της προσομοίωσης, η λειτουργία batch δεν υποστηρίζει τη δυνατότητα αυτή.

Προκειμένου ο χρήστης να εκτελέσει μία batch προσομοίωση με τις τιμές των παραμέτρων που επιθυμεί, θα πρέπει να αλλάξει τον χάρτη και τις προκαθορισμένες τιμές των καθολικών μεταβλητών μέσα στον κώδικα, και να δηλώσει το εύρος των τιμών των μεταβλητών που θέλει να εξερευνηθούν στο batch πείραμα.

Δεδομένων των παραπάνω, στο κεφάλαιο αυτό δεν θα ασχοληθούμε με την εκμάθηση της λειτουργίας batch (περισσότερες πληροφορίες για την εκτέλεση πειραμάτων batch σε προσομοιώσεις GAMA μπορούν να βρεθούν [εδώ](#)¹⁴), παρά θα παρουσιάσουμε το παράδειγμα δύο απλών πειραμάτων batch, καθένα με διαφορετικό στόχο, τα οποία διατηρούν όλες τις προκαθορισμένες τιμές που έχουν προγραμματιστεί από προεπιλογή στον κώδικα της προσομοίωσης (χάρτης, πιθανότητες εύρεσης ανά περίπτωση, εύρη ταχυτήτων κτλ.), και εξερευνούν την επίτευξη του στόχου τους με την αλλαγή μίας μόνο μεταβλητής, του αριθμού των People Agents που κινούνται στο χάρτη.

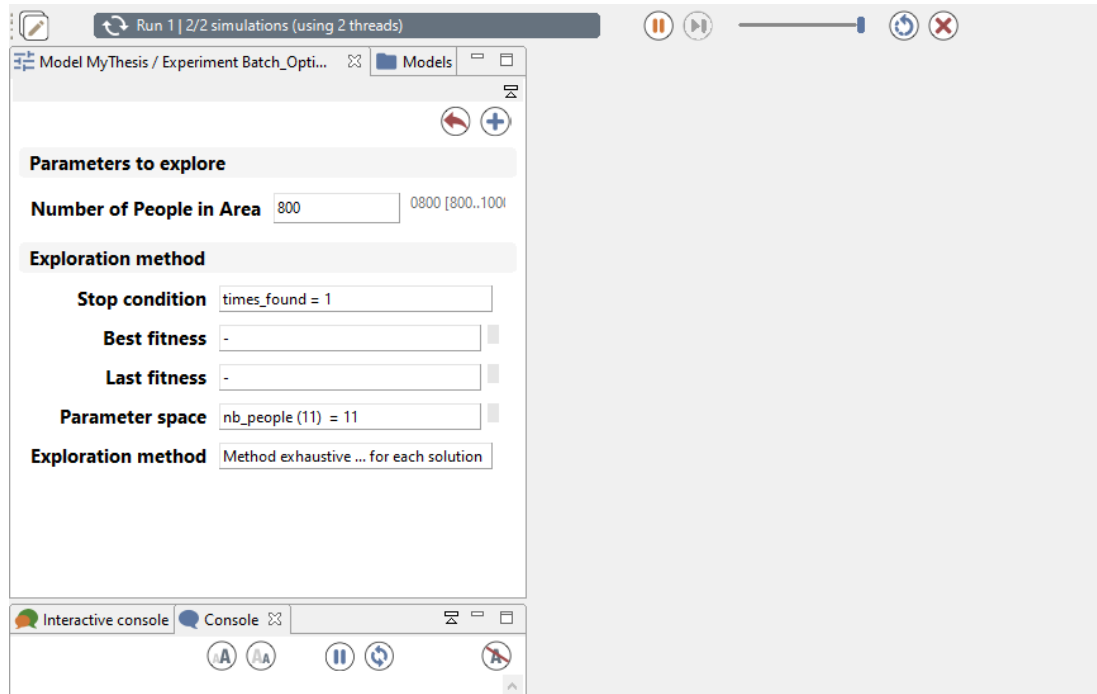
5.2.2.1 Χρόνος Πρώτης Εύρεσης Missing Person

Το Batch πείραμα αυτό ονομάζεται “Batch_Optimization_First_Time” και ο στόχος του είναι η εξερεύνηση του αριθμού των People Agents που υπάρχουν στην προσομοίωση για τον οποίο επιτυγχάνεται ελαχιστοποίηση του χρόνου μέσα στον οποίο θα βρεθεί ο missing person.

Το πείραμα θα τρέξει με προκαθορισμένες όλες τις μεταβλητές εκτός από την “nb_people” η οποία καθορίζει τον αριθμό των People Agents. Θα ξεκινήσει με 800 People Agents και θα αυξάνει τον αριθμό τους κατά 20 μέχρι να φτάσει τους 1000. Για κάθε αριθμό People Agents, θα τρέξουμε την προσομοίωση δύο φορές, θα έχουμε δηλαδή δύο επαναλήψεις με ακριβώς ίδιο σεντ παραμέτρων, έτσι ώστε να πετύχουμε μεγαλύτερη στατιστική αξιοπιστία των αποτελεσμάτων μας.

¹⁴<https://gama-platform.github.io/wiki/BatchExperiments#:~:text=Batch%20experiments%20allow%20to%20execute,are%20detailed%20in%20this%20page.>

Για να εκτελέσουμε το πείραμα, κάνουμε διπλό κλικ στο πλαίσιο “Batch_Optimization_First_Time” (Εικόνα 5.4.), το οποίο μας οδηγεί στη λειτουργία εκτέλεσης. Εκεί, πατάμε το πλήκτρο εκκίνησης της προσομοίωσης (Εικόνα 5.7.) και το πείραμα ξεκινάει (Εικόνα 5.21.).



Εικόνα 5.21. Πείραμα Batch σε εκτέλεση.

Είναι σημαντικό να σημειωθεί ότι λόγω των πολλαπλών επαναλήψεων, τα batch πειράματα είναι απαιτητικά σε πόρους και χρονοβόρα σε εκτέλεση, επομένως το πείραμα μας ενδέχεται να καθυστερήσει να ολοκληρωθεί.

Όταν το πείραμα μας θα έχει ολοκληρωθεί, στο πάνελ ελέγχου ροής προσομοίωσης εμφανίζεται το μήνυμα “Batch over.” συνοδευόμενο από τον αριθμό των επιμέρους πειραμάτων και των συνολικών αριθμών προσομοιώσεων, που όπως αναλύσαμε παραπάνω είναι στην συγκεκριμένη περίπτωση ο διπλάσιος του αριθμού επιμέρους πειραμάτων (Εικόνα 5.22.).



Εικόνα 5.22. Μήνυμα στο πάνελ ελέγχου ροής προσομοίωσης που σηματοδοτεί την ολοκλήρωση του Batch Πειράματος. Δίνονται επίσης ο αριθμός των πειραμάτων (runs) και των προσομοιώσεων (simulations) που πραγματοποιήθηκαν.

Μετά την ολοκλήρωση του πειράματος δημιουργείται το αρχείο “results_first_time_mr.csv” και αποθηκεύεται στον φάκελο της προσομοίωσης. Το

αρχείο περιέχει το σύνολο των αποτελεσμάτων, δηλαδή τους χρόνους για τους οποίους βρέθηκε για πρώτη φορά ο missing person για κάθε πλήθος People Agent που δοκιμάσαμε (Εικόνα 5.23.).

int(self)	nb_people	time / #day	time / #minute
0	800	0.14305555555555555	206.0
1	800	0.14305555555555555	206.0
2	820	1.7520833333333333	2523.0
3	820	1.7520833333333333	2523.0
4	840	1.1138888888888888	1604.0
5	840	1.1138888888888888	1604.0
6	860	0.7472222222222222	1076.0
7	860	0.7472222222222222	1076.0
8	880	0.08055555555555555	116.0
9	880	0.08055555555555555	116.0
10	900	2.2006944444444444	3169.0
11	900	2.2006944444444444	3169.0
12	920	0.24375	351.0
13	920	0.24375	351.0
14	940	0.12361111111111111	178.0
15	940	0.12361111111111111	178.0
16	960	5.1270833333333333	7383.0
17	960	5.1270833333333333	7383.0
18	980	2.7340277777777777	3937.0
19	980	2.7340277777777777	3937.0
20	1000	0.18958333333333333	273.0
21	1000	0.18958333333333333	273.0

Εικόνα 5.23. Το αρχείο “results_first_time_mp.csv”. Από αριστερά προς τα δεξιά, ο αριθμός προσομοίωσης, ο αριθμός των ανθρώπων σε κάθε προσομοίωση, ο συνολικός χρόνος πρώτης εύρεσης του χαμένου παιδιού σε μέρες και ο συνολικός χρόνος πρώτης εύρεσης του χαμένου παιδιού σε λεπτά.

5.2.2.2 Φορές Εύρεσης Missing Person σε Συγκεκριμένο Χρόνο

Το δεύτερο batch πείραμα το οποίο θα εκτελέσουμε ονομάζεται “Batch_Optimization_No_Times_found” και έχει σαν στόχο την εξερεύνηση του αριθμού των ανθρώπων που πρέπει να κινούνται στον χάρτη, έτσι ώστε μέσα σε ένα συγκεκριμένο χρονικό πλαίσιο (στη συγκεκριμένη εφαρμογή σε κάθε επανάληψη τρέχουμε την προσομοίωση για 4 μέρες), να μεγιστοποιούνται οι φορές που θα βρεθεί το χαμένο παιδί.

Όπως και προηγουμένως, όλες οι μεταβλητές εκτός του “nb_people” έχουν τις προκαθορισμένες τιμές τους. Διατηρούμε επίσης το κάτω όριο 800 ανθρώπων, το άνω όριο των 1000, το βήμα των 20 καθώς και την διπλή εκτέλεση του κάθε πειράματος για μεγαλύτερη εγκυρότητα αποτελεσμάτων.

Η διαδικασία εκτέλεσης είναι ακριβώς όμοια με το πείραμα “Batch_Optimization_First_Time”, με μόνη διαφορά ότι αυτή τη φορά επιλέγουμε με διπλό κλικ το πλαίσιο “Batch_Optimization_No_Times_found” και το τελικό αρχείο που δημιουργείται είναι το “results_no_times.csv”, το οποίο αποθηκεύεται στον φάκελο της προσομοίωσης και μπορεί να επεξεργαστεί περαιτέρω από τον χρήστη (Εικόνα 5.24.)

int(self)	nb_people	self.times_found	self.times_found_walking	self.times_found_driving	self.times_found_resting
14	940	7	4	2	1
12	920	6	4	2	0
0	800	5	2	0	3
10	900	5	2	1	2
21	1000	5	4	1	0
1	800	4	2	0	2
9	880	4	1	2	1
11	900	4	3	0	1
13	920	4	4	0	0
17	960	4	3	0	1
2	820	3	1	0	2
5	840	3	3	0	0
16	960	3	1	0	2
20	1000	3	2	1	0
3	820	2	1	0	1
8	880	2	2	0	0
19	980	2	0	2	0
4	840	1	1	0	0
6	860	1	1	0	0
15	940	1	1	0	0
7	860	0	0	0	0
18	980	0	0	0	0

Εικόνα 5.24. Το αρχείο “results_no_times.csv”. Από αριστερά προς τα δεξιά, ο αριθμός προσομοίωσης, ο αριθμός των ανθρώπων σε κάθε προσομοίωση, ο αριθμός φορών που βρέθηκε συνολικά το χαμένο παιδί, καθώς και ο αριθμός των φορών που βρέθηκε από ενώ ο People Agent που το αναγνώρισε περπατούσε, οδηγούσε ή βρισκόταν σπίτι αντίστοιχα.

Κεφάλαιο 6. Συμπεράσματα και Προοπτικές

Η παρούσα διπλωματική εργασία δημιουργεί μία προστιθέμενη αξία γνώσης αναφορικά με την αξιοποίηση τεχνικών Agent-Based Modeling, τη χρήση της πλατφόρμας GAMA και την προσομοίωση συνθηκών ανθρώπινης κινητικότητας, ενώ παράγει και ως ειδικότερο αποτέλεσμα την προσομοίωση εύρεσης ενός χαμένου παιδιού σε αστικό περιβάλλον.

Θα διατυπωθούν στη συνέχεια ορισμένα σημαντικά συμπεράσματα που μπορούν να εξαχθούν από την παρούσα διπλωματική εργασία, όπως επίσης και σκέψεις για την περαιτέρω αξιοποίηση και ανάπτυξη των εργαλείων, των τεχνικών και των μεθόδων που χρησιμοποιήθηκαν στην εργασία αυτή.

Μέσα από την αναλυτική μελέτη του Agent-Based Modeling μπορούμε να συμπεράνουμε πως από την κίνηση και συμπεριφορά απλών, μεμονωμένων μερών ενός συστήματος μπορούν να προκύψουν ολικές πολύπλοκες και πολυσύνθετες συμπεριφορές, όπως στο παράδειγμα που είδαμε με τους φιλάθλους στο γήπεδο και το ανθρώπινο κύμα που σχημάτιζαν.

Με την μοντελοποίηση ετερογενών απλών μερών και τον προγραμματισμό των κανόνων που διέπουν τη συμπεριφορά τους, τα μέρη αρχίζουν να αλληλεπιδρούν το ένα με το άλλο, λειτουργούν πλέον ως ένα σύνολο, και έτσι η κατάσταση κάθε μέρους δεν εξαρτάται μόνο από τη δική του συμπεριφορά. Καθορίζεται επίσης από την αλληλεπίδρασή του με άλλα μέρη αλλά και από την αλληλεπίδρασή του με σύνολα μερών. Δημιουργούνται και αναδύονται έτσι συμπεριφορές οι οποίες δεν προγραμματίστηκαν στην προσομοίωση, αλλά προέκυψαν από την προσομοίωση. Τέτοια παραγωγή διαφορετικού ολικού αποτελέσματος σε σχέση με τα απλά μέρη συναντάμε και στην φύση σε περιπτώσεις όπου το όλον είναι εμφανέστερο και δραστικά ισχυρότερο από το σύνολο των μερών, όπως το θαλάσσιο κύμα ή η παλίρροια. Σε άλλες περιπτώσεις, όπως σε εκείνη της ανθρώπινης κινητικότητας, η αλληλεπίδραση των επιμέρους μερών μπορεί να οδηγήσει ακόμα και στην παραγωγή νέας γνώσης.

Λαμβάνοντας υπόψιν αυτές τις δυνατότητες και την αξία του Agent-Based Modeling ως προς την προσέγγισή και την παραγωγή αποτελεσμάτων και γνώσης, αναζητήθηκαν οι εφαρμογές εκείνες οι οποίες θα επωφελούνταν περισσότερο από μοντελοποίηση με Agent-Based Modeling. Ένα από τα συμπεράσματα στα οποία καταλήξαμε είναι ότι περιπτώσεις προβλημάτων τα οποία μπορούν να περιγραφούν στην φυσική τους ερμηνεία με όρους Agent-Based Modeling και agents μπορούν να μοντελοποιηθούν με πολύ λογικό και σχεδόν φυσικό τρόπο χρησιμοποιώντας το Agent-Based Modeling.

Τα προβλήματα ανθρώπινης κινητικότητας αποτελούν ένα χαρακτηριστικό τέτοιο είδος προβλημάτων. Στα προβλήματα ανθρώπινης κινητικότητας, σύνολα ανθρώπων μπορούν να οριστούν ως επιμέρους κλάσεις agents, να μοντελοποιηθούν οι συμπεριφορές τους, να παρατηρηθούν οι αλληλεπιδράσεις τους και να εξερευνηθούν οι δημιουργούμενες και αναδυσόμενες συμπεριφορές και φαινόμενα. Η προσέγγιση με Agent-Based Modeling αποτελεί ένα σημαντικό βήμα στην εύρεση λύσεων για πολλά προβλήματα ανθρώπινης κινητικότητας τα οποία δεν μπορούν να επιλυθούν χρησιμοποιώντας παραδοσιακές προσεγγίσεις.

Μία ακόμη πολύ σημαντική συνιστώσα των προβλημάτων ανθρώπινης κινητικότητας, η οποία σε πολλές περιπτώσεις δεν μπορεί να λείπει κατά την μοντελοποίηση

προσεγγίσεων λύσεων τους, εκτός φυσικά από την συμπερίληψη του χρόνου που πάντα έχει ιδιαίτερη σημασία, είναι η χωρική συνιστώσα. Πιο συγκεκριμένα, συμπεράναμε το μεγάλο ενδιαφέρον που αναπτύσσεται σχετικά με την ενσωμάτωση GIS και Agent-Based Modeling, και τις προοπτικές που δημιουργούνται από τη συνδυασμένη χρήση τους. Όπως είδαμε, η αναπαράσταση του κόσμου ως μία σειρά στρωμάτων από ένα GIS δίνει τη δυνατότητα δημιουργίας ενός τεχνητού κόσμου που προσομοιώνει τον πραγματικό, με σημαντικό βαθμό ακρίβειας αναπαράστασης που εξαρτάται από την ποιότητα και αξιοπιστία των γεωγραφικών δεδομένων και των δεδομένων των χαρτών μας. Στον τεχνητό αυτό κόσμο μπορούν να «κατοικήσουν» οι agents μας, και η προσομοίωση της λύσης του κινητικού προβλήματος που εξερευνούμε μπορεί να γίνει με πραγματικά δεδομένων χαρτών.

Τα πλεονεκτήματα και οι δυνατότητες που προκύπτουν από τον συνδυασμό Agent-Based Modeling και GIS μας οδήγησαν στην αναζήτηση μίας πλατφόρμας που να υλοποιεί κατά τον βέλτιστο τρόπο τον συνδυασμό αυτό, και να ικανοποιεί ταυτόχρονα το πλήθος άλλων προδιαγραφών που θέσαμε, προδιαγραφές για εύκολη χρήση, πλούσιο γραφικό περιβάλλον, επεκτασιμότητα και παραμετροποιησιμότητα. Η αναζήτηση αυτή ολοκληρώθηκε με την επιλογή της πλατφόρμας GAMA.

Κατά τη διαδικασία ανάπτυξης της προσομοίωσης, χρήσης της GAMA, αλλά και της δημιουργίας του οδηγού για την χρήση της GAMA και της προσομοίωσή μας από τον χρήστη, εξαγάγαμε ένα σύνολο συμπερασμάτων και για την πλατφόρμα GAMA.

Οι δυνατότητες που προσφέρει η χρήση της GAMA για δημιουργία μοντέλων Agent-Based Modeling είναι εξαιρετικά μεγάλες, αφού, εκτός από την υποστήριξη για προγραμματισμό μοντέλων Agent-Based Modeling με πολλαπλά επίπεδα, μπορεί επίσης να συμπεριλάβει εξαιρετικά καινοτόμες τεχνολογίες, και ταυτόχρονα διαθέτει μία εκτεταμένη βιβλιοθήκη με συναρτήσεις ειδικά κατασκευασμένες για την ανάπτυξη μοντέλων Agent-Based Modeling για πολύ σημαντικό εύρος εφαρμογών. Παράλληλα, η ιδιαίτερη προσέγγιση της πλατφόρμας GAMA σε σχέση με το Agent-Based Modeling, να αντιμετωπίζει όχι μόνο τους ανθρώπους της προσομοίωσής μας, ή μόνο τα κτήρια ή τους δρόμους (δίνοντας παραδείγματα από την προσομοίωσή μας), αλλά όλα τα στοιχεία της προσομοίωσης και ακόμη και λειτουργίες της ίδιας της πλατφόρμας (π.χ. πειράματα ή οθόνες) ως agents, προσδίδοντας τους χαρακτηριστικά, συμπεριφορές και ιδιότητες, αποτελεί μία πολύ ενδιαφέρουσα λογική, και βοηθάει να εισαχθεί κανείς πλήρως στη φιλοσοφία του Agent-Based Modeling.

Από την άλλη πλευρά, μέσα από την ενασχόληση με την πλατφόρμα GAMA αναδείχθηκαν και οι προκλήσεις που υπάρχουν στον προγραμματισμό μοντέλων με Agent-Based Modeling, αλλά και η δυσκολία ενασχόλησης τόσο χρηστών όσο και προγραμματιστών με την πλατφόρμα. Ο προγραμματισμός μοντέλων Agent-Based Modeling σε GAMA συγκεκριμένα, αλλά και στο σύνολο των πλατφορμών ανάπτυξης που είναι διαθέσιμες, απαιτεί την εκμάθηση ενός νέου τρόπου προσέγγισης που λαμβάνει υπόψιν agents, συμπεριφορές και αλληλεπιδράσεις, που διαφέρει σημαντικά από την διαδικασία προγραμματισμού με παραδοσιακούς όρους απλών μεταβλητών και επαναλήψεων.

Ταυτόχρονα, η πλατφόρμα GAMA στη λειτουργία της είναι σχετικά εύχρηστη για οποιονδήποτε χρήστη, τουλάχιστον όσον αφορά την εκτέλεση πειραμάτων κανονικής λειτουργίας, -όπως είδαμε, τα batch πειράματα ενέχουν εξαιρετικές δυσκολίες για μη

έμπειρους χρήστες-, παρ' όλα αυτά, η διαδικασία εγκατάστασης και η συνεχής ενημέρωση τόσο της GAMA όσο και του JDK που την συνοδεύει, δημιουργούν δυσκολίες και εμπόδια για την συμμετοχή περισσότερων χρηστών στο οικοσύστημα ανάπτυξης εφαρμογών με GAMA και Agent-Based Modeling. Αξιολογήσαμε επίσης ως σημαντικό βήμα προς τη σωστή κατεύθυνση την ανάπτυξη διαδικτυακής έκδοσης της πλατφόρμας GAMA, η οποία θα ολοκληρωθεί μελλοντικά, έτσι ώστε προγραμματιστές και χρήστες να μπορούν να γράφουν και να προσομοιώνουν μοντέλα σε ένα πρόγραμμα περιήγησης ιστού, χωρίς να χρειάζεται να εγκαταστήσουν στον υπολογιστή τους την πλήρη έκδοση της GAMA.

Για την αντιμετώπιση κάποιων εκ των παραπάνω δυσκολιών, και με στόχο τη δημιουργία περαιτέρω δυνατοτήτων πειραματισμού με ανερχόμενες τεχνολογίες όπως το Agent-Based Modeling και η ανάπτυξη σε GAMA για περισσότερους ερευνητές και χρήστες, μαζί με την επίλυση του συγκεκριμένου πρακτικού προβλήματος της εύρεσης ενός χαμένου παιδιού σε αστικό περιβάλλον, την οποία εξερευνά και μοντελοποιεί η προσομοίωσή μας, δίνονται αναλυτικές οδηγίες ανάπτυξης των σημαντικότερων σημείων της κωδικοποίησης, οδηγίες εγκατάστασης και χρήσης.

Μερικά από τα σημαντικότερα συμπεράσματα που προσφέρει η παρούσα διπλωματική εργασία προέρχονται από τη λειτουργία της προσομοίωσης.

Με την εκτέλεση των πειραμάτων κανονικής λειτουργίας επιβεβαιώσαμε την επιπλέον ευελιξία που προσφέρει στον χρήστη η χρήση των παραμέτρων γραφικού περιβάλλοντος, και το πόσο εύκολη καθιστά την εκτέλεση πειραμάτων με διαφορετικούς συνδυασμούς χαρτών, ταχυτήτων κίνησης, τιμών του πιθανοτικού μοντέλου και των άλλων παραμετροποιήσιμων επιλογών. Επαληθεύσαμε την σημασία όχι μόνο των αξιόπιστων δεδομένων χαρτών, αλλά και την βελτίωση της ακρίβειας που προκύπτει με την εξαγωγή χαρτών που έχουν πολυγωνικό σχήμα, προσαρμοσμένο στις ανάγκες του εκάστοτε πειράματός μας, όπως οι χάρτες που εξάγουμε μέσω της Extract.BBBike. Συνειδητοποιήσαμε επίσης τη σημασία των δεδομένων εποπτείας, και το πόσο μπορούν τα κατάλληλα διαγράμματα και η δυνατότητα παρακολούθησης των κατάλληλων μεταβλητών να βελτιώσουν τη διαδικασία πειραματισμού.

Από την εκτέλεση πληθώρας πειραμάτων κανονικής λειτουργίας είδαμε στην πράξη το πόσο ριζικά επηρεάζουν την ανεύρεση ή μη του παιδιού έστω και μικρές αλλαγές στις παραμέτρους, καθώς και την σημαντική συνεισφορά του πιθανοτικού μοντέλου που υλοποιήσαμε στην επίτευξη εύλογων αποτελεσμάτων ανεύρεσης του παιδιού. Σημαντική συνεισφορά σε μία πιο ρεαλιστική μοντελοποίηση ήταν και η εισαγωγή των σημείων ενδιαφέροντος και της συγκεκριμένη αρχικής θέσης και ώρας, από την οποία το παιδί ξεκίνησε να αγνοείται, αφού παρατηρήσαμε διαφορά στους χρόνους εύρεσης σε πειράματα που εμπεριείχαν σημεία ενδιαφέροντος σε κεντρικά πολυσύχναστα σημεία της προσομοίωσης. Καταλήξαμε έτσι στο συμπέρασμα ότι η μεγαλύτερη γνώση για το παιδί και οι πληροφορίες σχετικά με αυτό ενισχύουν την προσπάθεια ανεύρεσής του, ακόμα και στην προσομοίωση.

Από τα πειράματα εκτέλεσης πολλαπλών προσομοιώσεων batch, καταλήξαμε στη ουσία της εκμετάλλευσης της σημερινής τεράστιας υπολογιστικής δύναμης από μοντέλα Agent-Based Modeling, αφού μέσω των επαναληπτικών αυτών πειραμάτων δώσαμε στο χρήστη τη δυνατότητα να εξερευνήσει, χωρίς χρονικούς περιορισμούς, και με την εξαγωγή ουσιωδών στατιστικών αποτελεσμάτων, το πώς σημαντικές

παράμετροι όπως το πλήθος των ανθρώπων που κινείται στο χάρτη επηρεάζουν τον στόχο της ανεύρεσης του χαμένου παιδιού, και μάλιστα με πολλαπλές ερμηνείες της έννοιας «ανεύρεση» για την προσομοίωση (ανεύρεση είτε την πρώτη φορά είτε αριθμός φορών ανεύρεσης σε συγκεκριμένο χρόνο). Αναλύθηκε επίσης η δυσκολία που παρουσιάζουν τα batch πειράματα σε σχέση με την αλλαγή στόχων, τιμών παραμέτρων αλλά και ορισμό παραμέτρων εξερεύνησης, αφού όλες αυτές οι επιλογές δεν δύνανται να γίνουν με τη χρήση γραφικού περιβάλλοντος, αλλά πρέπει να πραγματοποιηθούν μέσω τροποποιήσεων στον κώδικα ορισμού του κάθε πειράματος, κάτι που απαιτεί σημαντικά βαθύτερο επίπεδο γνώσης του αντικειμένου από τον χρήστη.

Από την εκτέλεση batch πειραμάτων εξερευνήσαμε το πώς επηρεάζει την ανεύρεση του παιδιού ο αριθμός των ανθρώπων που κινούνται στην προσομοίωση. Δημιουργώντας πειράματα με δύο διαφορετικά σενάρια, αφενός το σενάριο τερματισμού με την πρώτη εύρεση του παιδιού και εξαγωγής του χρόνου εύρεσης, και αφετέρου το σενάριο τερματισμού σε συγκεκριμένο χρονικό διάστημα και εξαγωγής των φορών εύρεσης σε συνδυασμό με τα στοιχεία τρόπου κίνησης των ανθρώπων που εντόπισαν το παιδί, μάθαμε περισσότερα για το πώς η αύξηση ή η μείωση του πληθυσμού των ανθρώπων στον ίδιο χάρτη μπορεί να έχει σημαντική επίδραση ως προς τα αποτελέσματα εύρεσης του παιδιού. Πιο συγκεκριμένα, μεγαλύτερος πληθυσμός ανθρώπων σε επιφυλακή αυξάνει σημαντικά την αποτελεσματικότητα ως προς την εύρεση του παιδιού μέσω της προσομοίωσης, όπως λογικά θα αναμέναμε να συμβαίνει στην πραγματικότητα. Η αύξηση της επιφυλακής αποτελεί το κλειδί για τον εντοπισμό του παιδιού και όχι η απλή αύξηση του πληθυσμού, μέρος του οποίου μπορεί τυχαία να εντοπίσει το παιδί.

Οι τύποι αυτοί πειραμάτων, κανονικής λειτουργίας και batch, καθώς και η ανάλυση των επιλογών για παραμετροποίηση, εκτέλεση και χρήση τους δημιουργεί την προστιθέμενη αξία που επιθυμούσαμε να προσφέρουμε σε σχέση με την προσομοίωση της μοντελοποίησης κίνησης ανθρώπων σε αστικό περιβάλλον, και συγκεκριμένα της ανεύρεσης ενός χαμένου παιδιού.

Από την εμπειρία που αποκτήσαμε μέσα από την ανάλυση και την εμβάθυνση στη χρήση Agent-Based Modeling και GAMA με την μελέτη των εννοιών και την ανάπτυξη της προσομοίωσης, μπορούμε να διατυπώσουμε ως συμπεράσματα εκπορευόμενα από την παρούσα εργασία και κάποιες σκέψεις για την περαιτέρω αξιοποίηση και ανάπτυξη της συγκεκριμένης προσομοίωσης, και γενικότερα της προσομοίωσης της ανθρώπινης κινητικότητας σε αστικό περιβάλλον.

Ορισμένες πτυχές που θα μπορούσαν να αναπτυχθούν σε μελλοντικές εργασίες είναι:

- Περισσότερο ρεαλιστική προσομοίωση της κίνησης ανθρώπων μέσα στην πόλη, με ενσωμάτωση στοιχείων κυκλοφοριακής συμφόρησης (περισσότερες πληροφορίες για την ενσωμάτωση στοιχείων κυκλοφοριακής συμφόρησης στην πλατφόρμα GAMA μπορούν να βρεθούν [εδώ](https://gama-platform.github.io/wiki/RoadTrafficModel)¹⁵), μοντελοποίηση της κίνησης των agents οι οποίοι ευλόγως θα επιχειρούν να αποφεύγουν τους δρόμους με μεγάλη κυκλοφοριακή συμφόρηση, σε συνδυασμό με μοντελοποίηση του προφίλ παιδιών που κατά περίπτωση μπορεί να αποφεύγουν τα σημεία με

¹⁵ <https://gama-platform.github.io/wiki/RoadTrafficModel>

συνωστισμό, και εισαγωγή επιπλέον τρόπων κίνησης (MMM, ταξί, ακόμη και πλοίων και αεροπλάνων). Εισαγωγή επίσης διαφορετικών προφίλ κίνησης για καθημερινές και Σαββατοκύριακα.

- Μοντελοποίηση της μετακίνησης των ανθρώπων όχι μόνο από το χώρο εργασίας τους προς το σπίτι και αντίστροφα, αλλά προσομοίωση του συνόλου των μετακινήσεων με επέκταση των ήδη λαμβανομένων υπόψιν, ώστε να περιλαμβάνουν και μετακινήσεις για διασκέδαση, κοινωνικές επισκέψεις, αγορές (με αντίστοιχη μοντελοποίηση κτηρίων π.χ. εμπορικών κέντρων, θεάτρων κτλ.).
- Περισσότερο ρεαλιστική αναπαράσταση των κτηριακών υποδομών της περιοχής του χάρτη, με εξαγωγή επιπλέον χαρακτηριστικών από τον χάρτη που εισάγεται στην προσομοίωση, όπως συγκεκριμένη χρήση κτηρίου (εταιρεία, μπαρ, χώρος εκδηλώσεων κτλ.) και δημιουργία σεναρίων επίσκεψης κτηρίων από τους agents ανάλογα με τη χρήση των κτηρίων αυτών.
- Υλοποίηση εισόδου και εξόδου των agents από τον χάρτη της προσομοίωσης. Για παράδειγμα κάποιοι agents ζουν, δουλεύουν ή απλά μετακινούνται κάποιες φορές εκτός του χάρτη της προσομοίωσης και δίνεται στον χρήστη η δυνατότητα να επιλέγει το ποσοστό των agent που το κάνουν αυτό.
- Αναλυτική μελέτη των σχετικών δημοσιευμένων ερευνών και δημιουργία προφίλ διαφορετικών ηλικιών και περιπτώσεων εξαφάνισης παιδιών (παιδί που έχει φύγει από το σπίτι, παιδί που χάθηκε λόγω μικρής ηλικίας, παιδί που απήχθη κτλ.) και αντίστοιχη μοντελοποίηση της συμπεριφοράς του missing person. Προσθήκη δυνατότητας επιλογής από το χρήστη του προφίλ του παιδιού που επιλέγεται για κάθε πείραμα.
- Στην περίπτωση της πραγματικής περίπτωσης ενός παιδιού που χάθηκε, εισαγωγή των στοιχείων του στην προσομοίωση, μαζί με το προφίλ τα χαρακτηριστικά και τις πληροφορίες που γνωρίζουμε γι' αυτό, και εκτέλεση πειραμάτων batch σε διάφορα σενάρια, σε συνδυασμό με την ενσωμάτωση νευρωνικών δικτύων και μηχανικής μάθησης για την εύρεση της διαδρομής που πιθανώς να ακολουθήσει στην προσομοίωση και κατ' επέκταση στην πραγματική ζωή.
- Μοντελοποίηση συστήματος ειδοποίησης των agents ότι το χαμένο παιδί βρίσκεται στην περιοχή τους με βάση κριτήρια γεωγραφικού εντοπισμού, σε αντιπαράβολή με την λειτουργία επιφυλακής βάσει του πιθανοτικού μοντέλου που έχουμε ορίσει. Οι agents λαμβάνοντας την ειδοποίηση θα μεταβούν σε κατάσταση επιφυλακής ώστε να ξεκινήσουν να ψάχνουν ενεργά το παιδί.

Βιβλιογραφία

- [1] R. Axelrod, “Advancing the Art of Simulation in the Social Sciences,” 1997.
- [2] C. M. Macal, “Everything you need to know about agent-based modelling and simulation,” *J. Simul.*, vol. 10, no. 2, pp. 144–156, May 2016, doi: 10.1057/jos.2016.7.
- [3] C. M. Macal and M. J. North, “Agent-based modeling and simulation,” in *Proceedings - Winter Simulation Conference*, 2009, pp. 86–98, doi: 10.1109/WSC.2009.5429318.
- [4] C. M. Macal and M. J. North, “Tutorial on agent-based modeling and simulation part 2: How to model with agents,” in *Proceedings - Winter Simulation Conference*, Dec. 2006, pp. 73–83, doi: 10.1109/WSC.2006.323040.
- [5] C. M. MacAl and M. J. North, “Tutorial on agent-based modelling and simulation,” *J. Simul.*, 2010, doi: 10.1057/jos.2010.3.
- [6] M. J. North and C. M. Macal, *Managing Business Complexity: Discovering Strategic Solutions with Agent-Based Modeling and Simulation*, vol. 9780195172. Oxford University Press, 2007.
- [7] B. L. Drees *et al.*, “Derivation of genetic interaction networks from quantitative phenotype data,” *Genome Biol.*, 2005, doi: 10.1186/gb-2005-6-4-r38.
- [8] A. Crooks, A. Heppenstall, and N. Malleon, “Agent-Based Modeling,” in *Comprehensive Geographic Information Systems*, 2017.
- [9] E. Bonabeau, “Agent-based modeling: Methods and techniques for simulating human systems,” *Proc. Natl. Acad. Sci. U. S. A.*, 2002, doi: 10.1073/pnas.082080899.
- [10] R. Garcia, “Uses of agent-based modeling in innovation/new product development research,” *J. Prod. Innov. Manag.*, vol. 22, no. 5, pp. 380–398, Sep. 2005, doi: 10.1111/j.1540-5885.2005.00136.x.
- [11] P. C. L. Silva, P. V. C. Batista, H. S. Lima, M. A. Alves, F. G. Guimarães, and R. C. P. Silva, “COVID-ABS: An agent-based model of COVID-19 epidemic to simulate health and economic effects of social distancing interventions,” *Chaos, Solitons and Fractals*, vol. 139, p. 110088, Oct. 2020, doi: 10.1016/j.chaos.2020.110088.
- [12] R. J. Dawson, R. Peppe, and M. Wang, “An agent-based model for risk-based flood incident management,” *Nat. Hazards*, vol. 59, no. 1, pp. 167–189, Oct. 2011, doi: 10.1007/s11069-011-9745-4.
- [13] A. T. Crooks and C. J. E. Castle, “The integration of agent-based modelling and geographical information for geospatial simulation,” in *Agent-Based Models of Geographical Systems*, 2012.
- [14] A. Crooks, *Geocomputation: A Practical Primer*, no. January. 2015.
- [15] I. Benenson and P. M. Torrens, *Geosimulation: Automata-based Modeling of Urban Phenomena*. 2006.
- [16] D. Â. Võ, A. Drogoul, and J. D. Zucker, “Multi-level agent-based modeling: A generic approach and an implementation,” in *Frontiers in Artificial Intelligence and Applications*, 2013, vol. 252, pp. 91–101, doi: 10.3233/978-1-61499-254-7-91.
- [17] P. Taillandier *et al.*, “Participatory modeling and simulation with the gama

- platform,” *JASSS*, vol. 22, no. 2, Mar. 2019, doi: 10.18564/jasss.3964.
- [18] P. Taillandier *et al.*, “Building, composing and experimenting complex spatial models with the GAMA platform,” *Geoinformatica*, vol. 23, no. 2, pp. 299–322, Apr. 2019, doi: 10.1007/s10707-018-00339-6.
- [19] T. K. Nguyen, B. Gaudou, T. V. Ho, and N. Marilleau, “Application of PAMS collaboration platform to simulation-based researches in soil science: The case of the micro-organism project,” 2009, doi: 10.1109/RIVF.2009.5174623.
- [20] B. Gaudou, N. Marilleau, and T. V. Ho, “Toward a Methodology of Collaborative Modeling and Simulation of Complex Systems,” 2010.
- [21] A. Chakraborty, B. Wilson, S. Sarraf, and A. Jana, “Open data for informal settlements: Toward a user’s guide for urban managers and planners,” *J. Urban Manag.*, vol. 4, no. 2, pp. 74–91, Dec. 2015, doi: 10.1016/j.jum.2015.12.001.
- [22] S. J. Miller, K. Scott, A. Uhl, and D. Weise, “MDA Distilled,” *AddisonWesley Prof.*, 2004.