Αποκεντρωμένος Υπολογισμός και Μοντελοποίηση

3ο Σύνολο Ασκήσεων

Λουδάρος Ιωάννης (1067400) - Λένος Χρίστου (1063014) - Αντωνίου Πέτρος (1063017)

Μπορείτε να δείτε την τελευταία έκδοση του Project εδώ ή σκανάροντας τον κωδικό QR που βρίσκεται στην επικεφαλίδα.

Περιγραφή Αναφοράς

Παρακάτω παραθέτουμε τις απαντήσεις μας στο "3ο Σύνολο Προγραμματιστικών/Θεωρητικών Ασκήσεων" του μαθήματος "Αποκεντρωμένος Υπολογισμός και Μοντελοποίηση" καθώς και σχόλια τα οποία προέκυψαν κατά την εκπόνηση του.

Περιεχόμενα

Σύσταση Ομάδας

Ιωάννης Λουδάρος | 1067400

iloudaros@upnet.gr
Φοιτητής 5ου έτους

Λένος Χρίστου | 1063014

up1063014@upnet.gr
Φοιτητής

Αντωνίου Πέτρος | 1063017

up1063017@upnet.gr
Φοιτητής

Απαντήσεις

Θεωρητικές Ασκήσεις

Άσκηση 1

Έχουμε το παρακάτω σύστημα μέσου όρου με δύο παίκτες:

$x_1(t+1) = x_1(t)$

$x_2(t+1) = a\,x_1(t) + (1-a)x_2(t)$

---

1. Το Μητρώο A

Ο παίκτης 1 βλέπουμε ότι ποτέ δεν αλλάζει άποψη. Αντιθέτως, ο παίκτης 2 επηρεάζεται από τον παίκτη 1.

Άρα, $A = \begin{bmatrix} 1 & 0 \\ a & 1-a \end{bmatrix}$

Για να δείξουμε ότι το μητρώο A είναι στοχαστικό ως προς τις γραμμές θα πρέπει το άθροισμα της κάθε γραμμής να είναι ίσο με 1. Πράγματι, για την γραμμή 1 έχουμε $1 + 0 = 1$ και για την γραμμή 2 έχουμε $a + 1 - a = 1$.

---

2. Ιδιοτιμές και Ιδιοδιανύσματα

$$A - \lambda I = \begin{bmatrix} 1 & 0 \\ a & 1-a \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1-\lambda & 0 \\ 0 & 1-a-\lambda \end{bmatrix}$$

$$det(A - \lambda I) = 0 \Leftrightarrow (1-\lambda)(1-\alpha-\lambda) = 0$$

Άρα, οι ιδιοτιμές του A είναι οι $\lambda_1 = 1$ και $\lambda_2 = 1 - \alpha$.

$$(A - \lambda I)x = 0 \implies \begin{cases} (1-\lambda)x_1 = 0 \\ a\,x_1 + (1-a-\lambda)x_2 = 0 \end{cases}$$

Για $\lambda = 1$

$$\begin{cases} 0 \cdot x_1 = 0 \\ a\,x_1 - a\,x_2 = 0 \end{cases} \implies x_1 = x_2 \implies x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

Για $\lambda = 1 - \alpha$

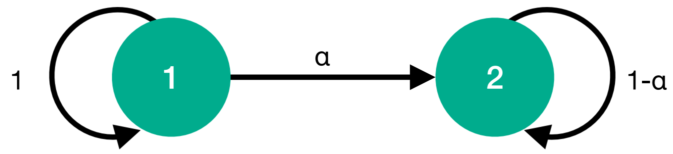
$$\begin{cases} \alpha \cdot x_1 = 0 \\ a\,x_1 + 0x_2 = 0 \end{cases} \implies x_1 = 0, x_2 \text{ αυθαίρετο, έστω ότι } x_2 = 1 \implies x = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Άρα τα ιδιοδιανύσματα του A είναι τα : $x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ και $x = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

---

3. Το γράφημα G

Το γράφημα είναι συνεκτικό, όχι ισχυρά όμως, γιατί δεν υπάρχει κατευθυνόμενο μονοπάτι από τον κόμβο 2 στον κόμβο 1.



---

4. Σύγκλιση Αλγορίθμου

Για τον παίκτη 1:

$$\lim_{t \to \infty} A^t x_1(0) = [1\,1\,...\,1]^T \cdot s^T \cdot x_1(0) = [1\,1\,...\,1]^T \cdot [1\,1]^T \cdot x_1(0)$$

Για τον παίκτη 2:

$$\lim_{t \to \infty} A^t x_2(0) = [1\,1\,...\,1]^T \cdot s^T \cdot x_2(0) = [1\,1\,...\,1]^T \cdot [1\,1]^T \cdot x_2(0)$$

Άσκηση 2

---

1. Πρώτη Περίπτωση

Όταν το 80% της υπολογιστικής ισχύος στο δίκτυο Bitcoin τρέχει την υλοποίηση A, εφόσον υπερβαίνει το 50%, υπερισχύει η λάθος πλειοψηφία κατά την διαδικασία επαλήθευσης. Έτσι, θα υπάρχει η δυνατότητα αλλαγής του ledger (υπολογισμός νέων headers). Ως αποτέλεσμα, στο blockchain θα καταγραφεί το μπλοκ με την διπλοξοδευμένη συναλλαγή ως έγκυρο.

---

2. Δεύτερη Περίπτωση

Όταν το 80% της υπολογιστικής ισχύος στο δίκτυο Bitcoin τρέχει την υλοποίηση B, εφόσον υπερβαίνει το 50%, υπερισχύει η σωστή πλειοψηφία κατά την διαδικασία επαλήθευσης. Έτσι, δεν θα υπάρχει η δυνατότητα αλλαγής του ledger (υπολογισμός νέων headers). Ως αποτέλεσμα, στο blockchain δεν θα καταγραφεί το μπλοκ με την διπλοξοδευμένη συναλλαγή ως έγκυρο.

Άσκηση 3

Ο Bob δεν πρέπει να δεχτεί το συγκεκριμένο πρωτόκολλο, καθώς η Alice μπορεί να στείλει το s πριν λήξει το συμβόλαιο του Bob (CM), αλλά μετά την λήξη του συμβολαίου της Alice (CA). Έτσι, ο Bob δεν θα λάβει το κόκκινο νόμισμα από την Alice και θα της επιστραφεί, ενώ η Alice θα λάβει το πράσινο νόμισμα από τον Bob, αφού θα ενεργοποιηθεί το CM.

Προγραμματιστικές Ασκήσεις

Άσκηση 1

Υλοποιήσαμε το μοντέλο στα πλαίσια της Άσκησης. Κάθε δρων υποκείμενο έχει μια αρχική άποψη δύο διαστάσεων η οποία είναι αποθηκευμένη στην μεταβλητή `initial_opinion`. Πέρα από αυτό, έχει έναν στοχαστικό διάνυσμα (η λίστα `trust`) που του δείχνει πόσο βάρος δίνει σε κάθε άλλη άποψη. Επίσης, έχει αποθηκευμένο το πόσο ισχυρογνώμον είναι, στην μεταβλητή `stubbornness`. Τέλος, ενδιαφέρον έχει η μεταβλητή `staying_still`, η οποία σε κάθε βήμα αποθηκεύει το αν το δρων υποκείμενο παρέμεινε ακίνητο ή όχι.

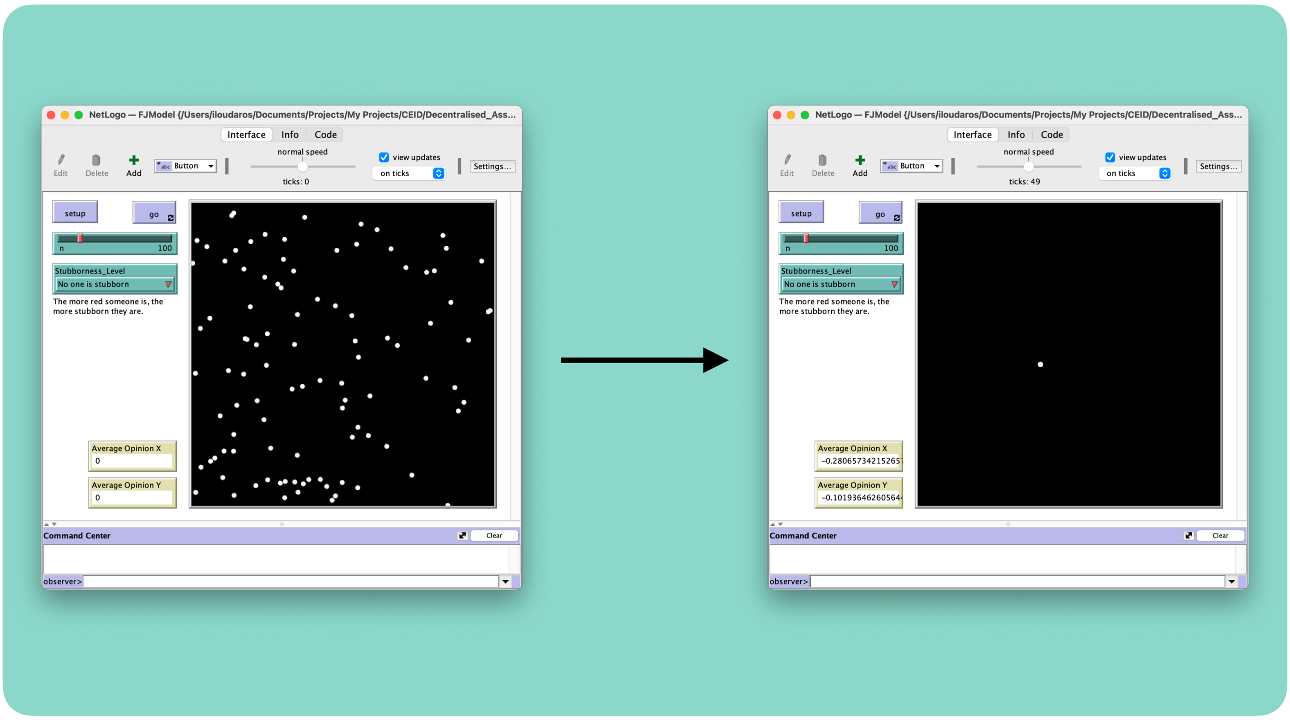
Η συνθήκη τερματισμού του αλγορίθμου είναι να έχουν όλοι οι πράκτορες, τιμή `staying_still = true`. Θα παρατηρήσετε ότι αυτή η συνθήκη είναι αρκετά αυστηρή. Το μοντέλο μας συνεχίζει να παρουσιάζει αλλαγές ακόμη και όταν δεν μπορούμε να τις παρατηρήσουμε (μέχρι δηλαδή η απόσταση μεταξύ δυο διαδοχικών απόψεων να είναι μικρότερη από το έψιλον της μηχανής). Αν μας ενδιαφέρει αποκλειστικά η οπτικοποίηση του αποτελέσματος, θα μπορούσαμε να χαλαρώσουμε την συνθήκη, όπως προτείνεται από την εκφώνηση.

Μπορείτε να τρέξετε το μοντέλο μας, ανοίγοντας το αρχείο “FJModel.nlogo”.

Το γραφικό περιβάλλον του μοντέλου είναι αρκετά απλό. Αφού επιλέξετε τον αριθμό των πρακτόρων και το Stubbonness Level, μπορείτε να πατήσετε “setup” και ύστερα go ώστε να πραγματοποιηθεί η δράση του μοντέλου.

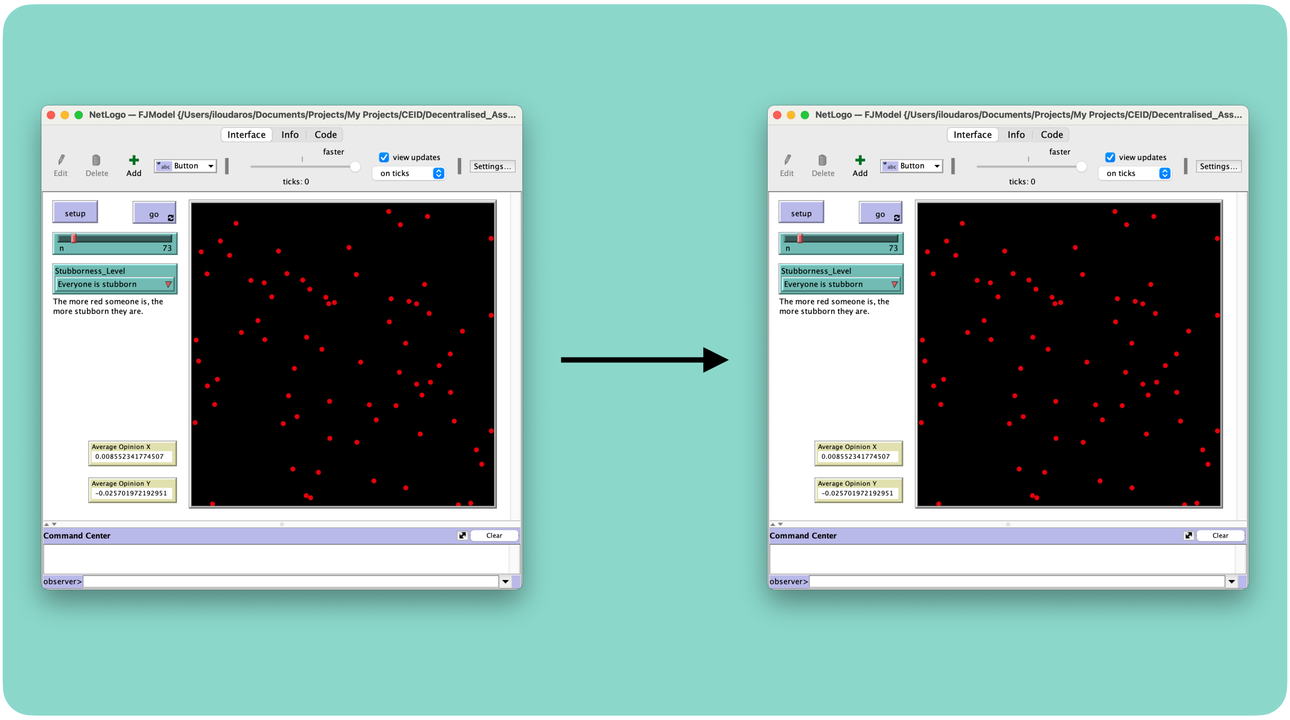
Κανένας δεν επιμένει στην αρχική του γνώμη

Επιλέγοντας την ρύθμιση “No one is stubborn”, πολύ γρήγορα παίρνουμε για ομοφωνία κάτι κοντά στο κεντροειδές των απόψεων ως ομοφωνία. Αυτό είναι λογικό, αφού ο κάθε πράκτορας συμπεριλαμβάνει στην άποψη του την άποψη των υπόλοιπων πρακτόρων με κάποιο βάρος. Αξίζει να σημειωθεί πως δεν έχουμε πάρει μέτρα ώστε να εξαλήψουμε φαινόμενα περιοδικότητας. Συνεπώς, με πιθανότητα όλο και μικρότερη όσο ανεβαίνει το πλήθος των πρακτόρων, υπάρχει περίπτωση ο αλγόριθμος να μην τερματίσει. Διαφορετικά, θα έχουμε ομοφωνία.



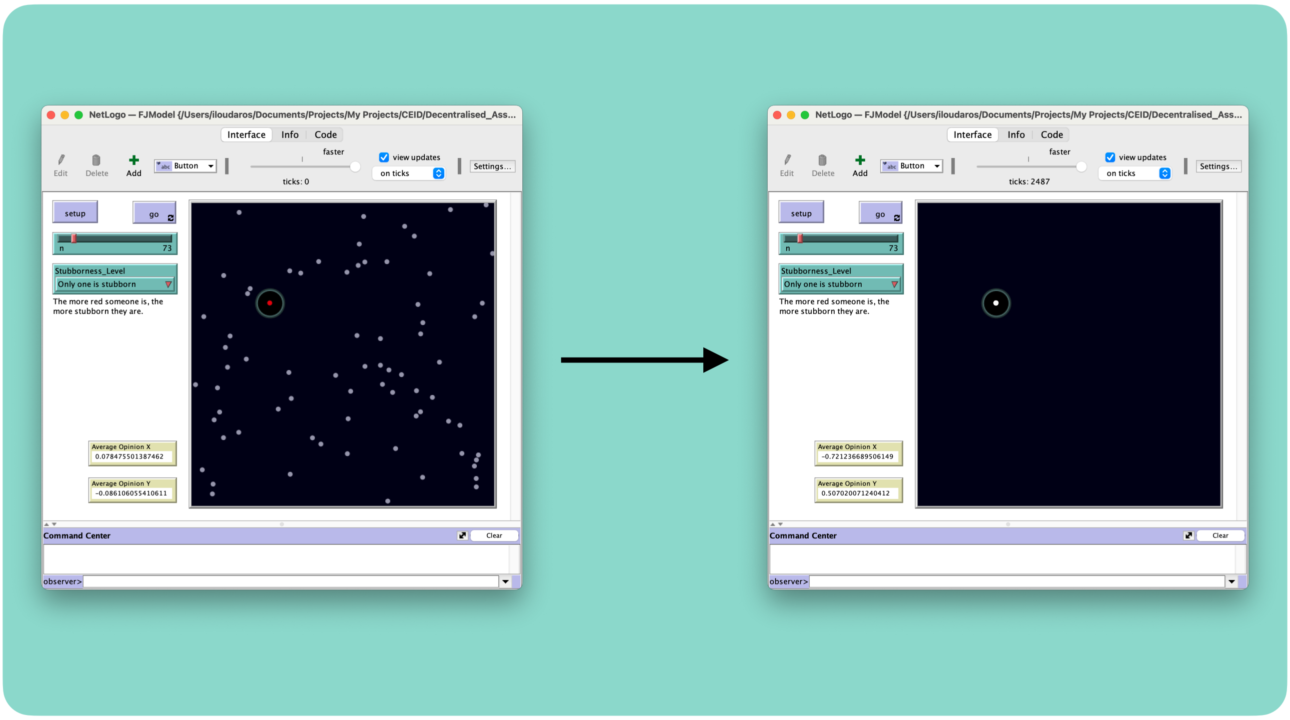
Όλοι επιμένουν στις αρχικές τους γνώμες

Επιλέγοντας την ρύθμιση “Everyone is Stubborn”, ο αλγόριθμος τερματίζει κατευθείαν. Αυτό είναι επόμενο, αφού κανένας πράκτορας δεν είναι διατεθειμένος να κουνηθεί. Προφανώς, με μεγάλη πιθανότητα, δεν φτάνουμε σε ομοφωνία.



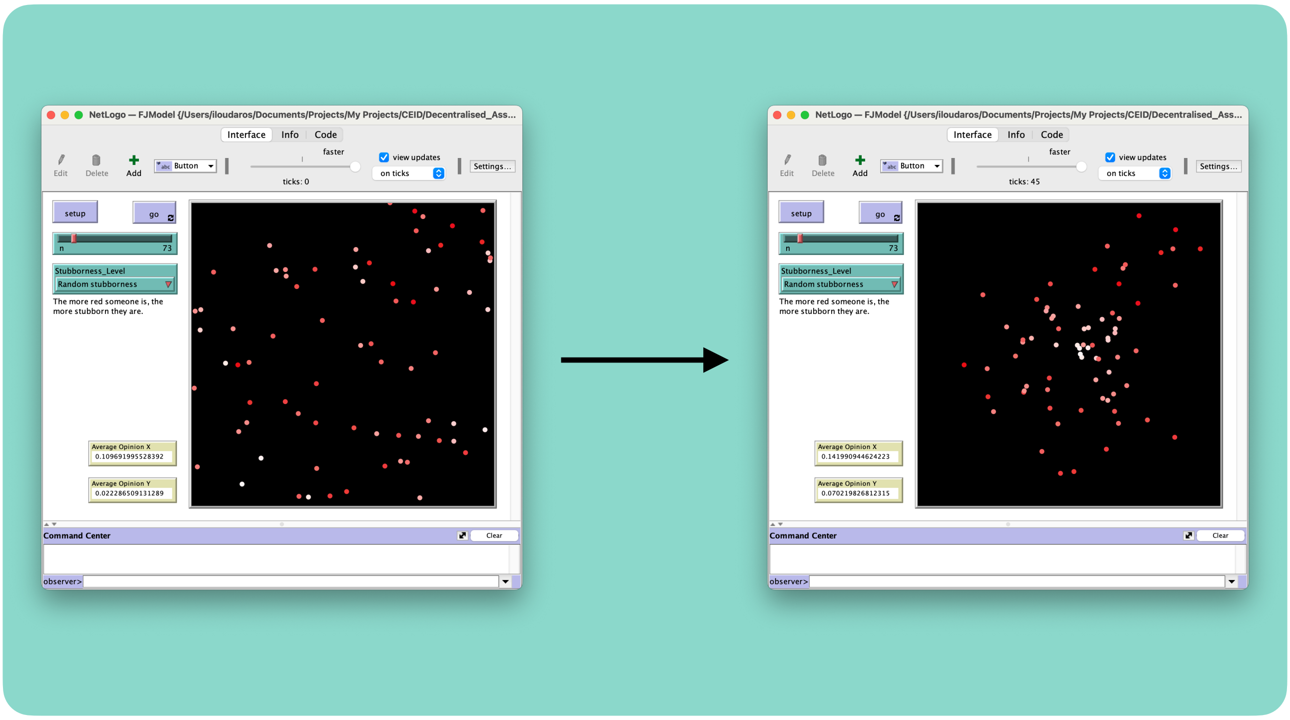
Υπάρχει ένας μοναδικός που επιμένει

Αν υπάρχει ένας μοναδικός πράκτορας που επιμένει, θα τραβήξει αργά αλλά σταθερά, όλους τους άλλους προς το μέρος του. Σίγουρα θα επιτευχθεί ομοφωνία.



Τυχαίο stubbornness

Επιλέγοντας την ρύθμιση “Random Stubbornness”, βλέπουμε ότι ανάλογα το stubbornness του κάθε πράκτορα, προσπαθεί να πάει προς το κεντροειδές των αμετάβλητων γνωμών. Πάλι, με μεγάλη πιθανότητα, δεν επιτυγχάνεται ομοφωνία.

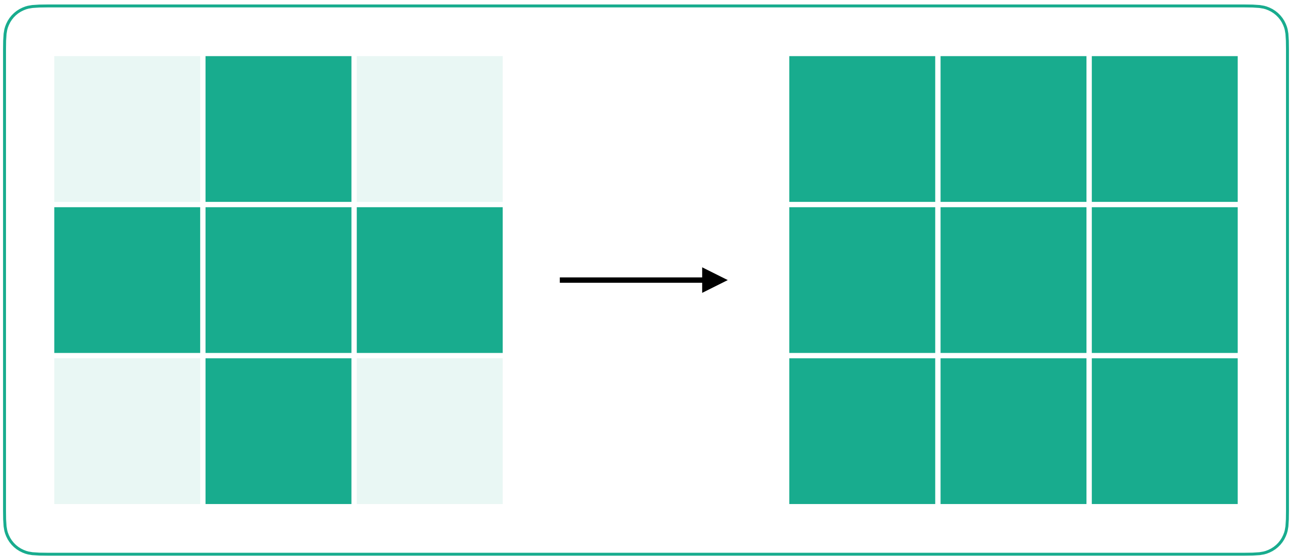


Άσκηση 2

Ξεκινήσαμε την ανάπτυξη του μοντέλου μας με βάση το μοντέλο της NETLOGO “Fire Simple Extension 2”. Ύστερα πραγματοποιήσαμε τις αλλαγές που ζητούνται, όπως φαίνεται παρακάτω.

Μπορείτε να τρέξετε το μοντέλο μας, ανοίγοντας το αρχείο “Fire.nlogo”.

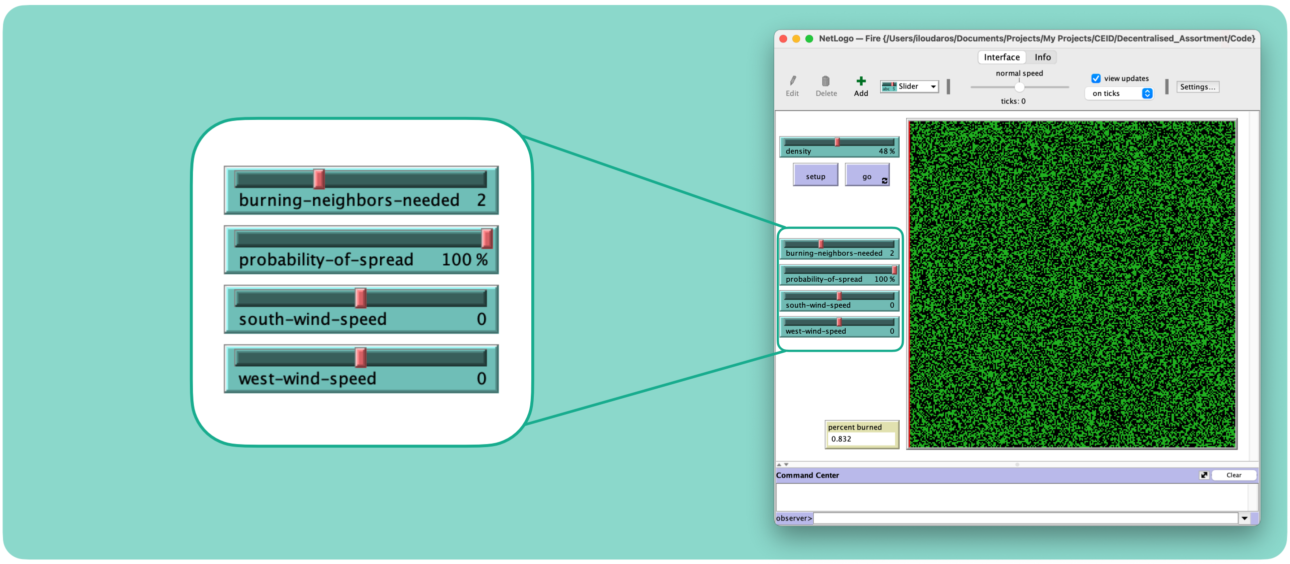
Αλλαγή σε γειτονιά Moore



Χρειάστηκε να αντικαταστήσουμε το “`neighbors4`”, με το “`neighbors`” και είμαστε έτοιμοι! Η αλλαγή σε γειτονιά Moore, μείωσε δραστικά την κρίσιμη πυκνότητα στο 40%! Αποτέλεσμα αναμενόμενο, αφού δώσαμε στην φωτιά περισσότερους τρόπους να εξαπλωθεί.

Προϋπόθεση στο πλήθος των γειτονικών δέντρων.

Προστέθηκε ένα slider (`burning-neighbours-needed`), ώστε να γίνεται να ρυθμιστεί το πλήθος των γειτονικών δέντρων που απαιτείται να καίγονται, πριν πάρει φωτιά ένα δέντρο.



Ο τρόπος που υλοποιήθηκε το συγκεκριμένο ερώτημα είναι ιδιαίτερα απλός. Ουσιαστικά, προστέθηκε μια συνθήκη επίτρεψης στην αρχή του υπολογισμού της πιθανότητας ανάφλεξης ενός δέντρου. Η συνθήκη αυτή, διασφαλίζει ότι οι υπολογισμοί θα γίνουν, μόνο αν οι γείτονες είναι τουλάχιστον όσοι αυτοί που έχουμε βάλει στο slider.

Όπως περιμέναμε αυτό επηρεάζει το μοντέλο σημαντικά! Ήδη, από τους δύο αναγκαίους γείτονες βλέπουμε ότι το δάσος μας γίνεται εξαιρετικά πιο ανθεκτικό.

Στους τρεις, τα μέτωπα της φωτιάς όλο και λεπτένουν. Είναι ιδιαίτερα ενδιαφέρον να δει κανείς τι συμβαίνει στο 100% της πυκνότητας. Η φωτιά όλο και λεπταίνει, δημιουργώντας ένα σχήμα βέλους. Αυτό συμβαίνει γιατί τα οριακά δέντρα πάνω και κάτω στο t=0 έχουν μόνο έναν αριστερό και έναν διαγώνιο γείτονα που καίγεται και έτσι δεν ανάβουν, έτσι προστατεύουν και όλα τα δέντρα της σειράς τους. Έπειτα, τα καινούργια “οριακά δέντρα”, έχουν πάλι μόνο έναν αριστερό και έναν διαγώνιο καιόμενο γείτονα κ.ο.κ.. Αυτό το φαινόμενο μπορεί να εξαλειφθεί αν επιτρέψουμε την κάθετη αναδίπλωση του χώρου.

Στους 4, είναι αδύνατον να εξαπλωθεί η φωτιά ακόμη και για πυκνότητα 100%, αφού στην αρχή έχουμε μόνο μια στήλη φωτιάς, και έτσι οποιοδήποτε μη καμένο δέντρο, έχει κατά μέγιστο, 3 γειτονικά δέντρα που καίγονται.

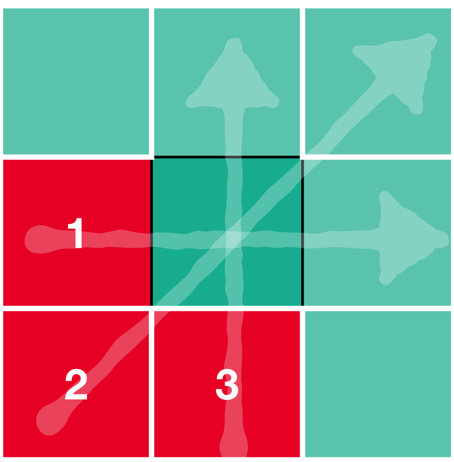
Προσοχή, για να χρησιμοποιηθεί αυτό το slider όπως ζητείται στην εκφώνηση, πρέπει τα υπόλοιπα sliders να είναι σε συγκεκριμένες θέσεις, όπως φαίνεται στην εικόνα.

Η επίδραση του ανέμου

Το μοντέλο “Fire Simple Extension 2” ήδη υπολόγιζε την επίδραση του ανέμου για την γειτονιά Neumann. Για να το κάνει αυτό, όριζε την πιθανότητα “probability of spread”. Αυτή είχε μια αρχική τιμή ορισμένη από τον χρήστη, και αυξανόταν, ή μειωνόταν για το κάθε δέντρο, ανάλογα με την κατεύθυνση του ανέμου και την σχετική θέση του δέντρου με τον καιόμενο γείτονα του. Υποστηριζόταν μόνο 2 κατευθύνσεις ανέμων, κάθετη και οριζόντια.

Για να επεκτείνουμε αυτή τη συμπεριφορά σε γειτονιά Moore, χρειάζεται να υπολογίζουμε την συνισταμένη του κάθετου και του οριζόντιου ανέμου. Στην προσπάθεια μας αυτή, προκύπτει ένα ενδιαφέρον φαινόμενο. Όταν υπάρχουν πολλαπλά καιόμενα δέντρα γειτονικά σε ένα μη καιόμενο, το κάθε ένα από αυτά, ανάλογα τον άνεμο, θα ανεβάσει ξεχωριστά την πιθανότητα να πάρει φωτιά το μη καιόμενο δέντρο.

Για παράδειγμα:

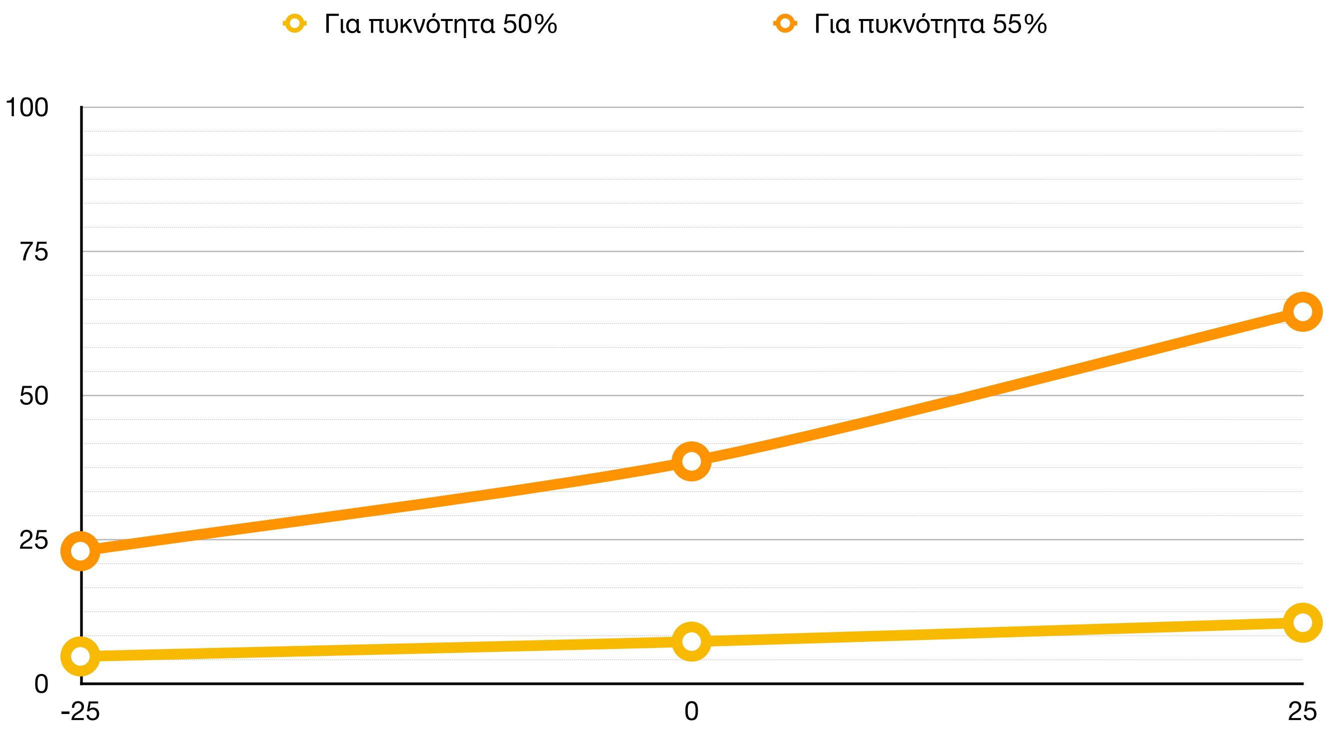


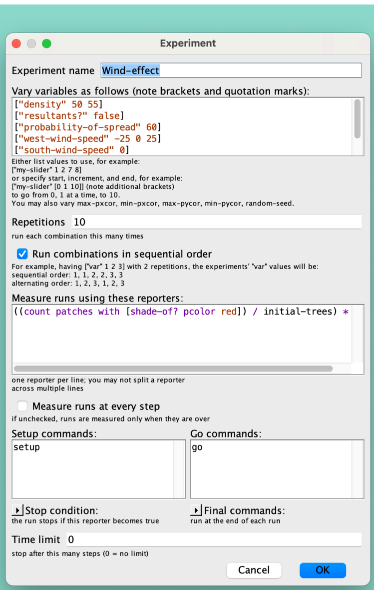
Ας εξετάσουμε τι θα συμβεί στο κεντρικό δέντρο στο σχήμα αριστερά, αν έχουμε και δυτικό και νότιο άνεμο. Η πιθανότητα του θα αυξηθεί και από το δέντρο 1 και από το δέντρο 2 και από το δέντρο 3!

Τρόπος να γίνει το παραπάνω πιο ακριβές, θα ήταν να υπολογίζεται ακριβώς η γωνία της συνισταμένης των ανέμων, ώστε να προσεγγίζεται η συνεισφορά του κάθε γειτονικού δέντρου, αφού πρώτα πολλαπλασιαστεί με έναν ημιτονοειδή παράγοντα. Θεωρήσαμε ότι κάτι τέτοιο ξεφεύγει από τα ζητούμενα της άσκησης, οπότε δεν το υλοποιήσαμε. Αντ'αυτού, η συνισταμένη θεωρούμε ότι είναι πάντα σε γωνία πολλαπλάσια των 45 μοιρών. Αυτό κάνει το δάσος μας πιο εύφλεκτο!

Για να ενεργοποιήσετε/απενεργοποιήσετε τον υπολογισμό των συνισταμένων, μπορείτε να χρησιμοποιήσετε το αντίστοιχο switch “`resultants?`”

Για να εξετάσουμε την επίδραση του ανέμου προετοιμάσαμε το πείραμα “Wind-effect" που μπορείτε να βρείτε στο εργαλείο BehaviorSpace. Για αυτό το πείραμα, θεωρούμε την βασική πιθανότητα ανάφλεξης 60% και θεωρούμε την μέγιστη ένταση του ανέμου 25 (Άρα μπορεί να προσθέσει ή να αφαιρέσει μέχρι και 0.25 στην αρχική πιθανότητα ανάφλεξης). Παραθέτουμε τα αποτελέσματα παρακάτω, κάθε τιμή της γραφικής παράστασης είναι στην πραγματικότητα μέσος όρος μεταξύ 10 εκτελέσεων των συγκεκριμένων ρυθμίσεων.





Παραθέτουμε τις ρυθμίσεις του πειράματος. Μπορείτε να δείτε τα αποτελέσματα του στο TSV που υπάρχει αναρτημένο στο repository του project μας, πατώντας το παρακάτω κουμπί.

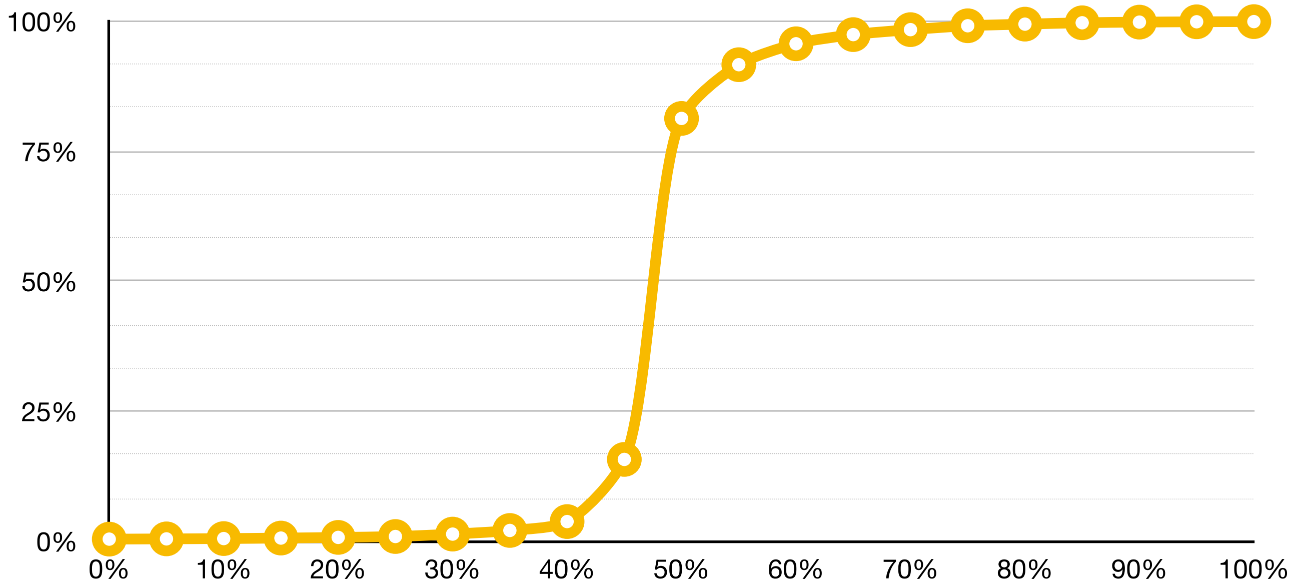
TSV με αποτελέσματα

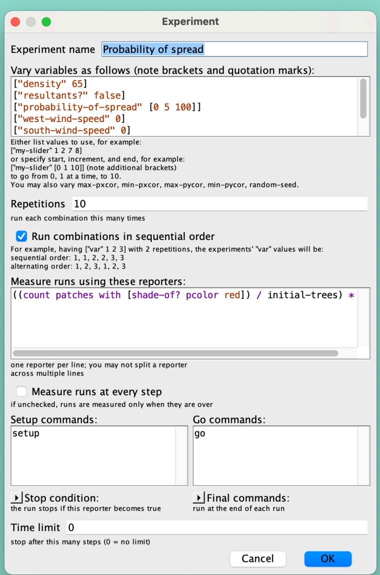
Στοχαστικότητα μοντέλου

Εισάγουμε στο μοντέλο μας την έννοια της πιθανότητας διάδοσης της φωτιάς. Για να μελετήσουμε την επίδραση της, έχουμε στήσει το πείραμα “Probability of spread” το οποίο είναι διαθέσιμο στο εργαλείο BehaviorSpace.

Κατά την διάρκεια εκπόνησης αυτού του ερωτήματος, συνειδητοποιήσαμε ένα λάθος που υπάρχει στο μοντέλο της NETLOGO. Αν τρέξετε το αρχικό μοντέλο με density 90 και πιθανότητα διάδοσης 100, θα δείτε κάτι αξιοπερίεργο. Το percentage burned, υπερβαίνει το 100%! Το λάθος του μοντέλου είναι απλό. Δεν βάζει απλά φωτιά στα δέντρα της πρώτης στήλης. Βάζει φωτιά σε ολόκληρη τη στήλη (δηλαδή ακόμη και σε σημεία που δεν ήταν δέντρα)! Αυτό έχει ως αποτέλεσμα, όταν μετά πάει να μετρήσει τα καμμένα τετράγωνα, να μετράει και όλα αυτά τα επιπλέον της πρώτης στήλης, με αποτέλεσμα το πλήθος των καμμένων patches να είναι μεγαλύτερο του πλήθους των δέντρων. Το σφάλμα διορθώθηκε ώστε να μπορούμε να πάρουμε σωστές μετρήσεις.

Τα αποτελέσματα που μετράμε είναι για πυκνότητα 65%. Στον κάθετο άξονα έχουμε το ποσοστό του δάσους που κάηκε, στον οριζόντιο την πιθανότητα διάδοσης. Κάθε τιμή της γραφικής παράστασης είναι στην πραγματικότητα μέσος όρος μεταξύ 10 εκτελέσεων των συγκεκριμένων ρυθμίσεων.





Παραθέτουμε τις ρυθμίσεις του πειράματος. Μπορείτε να δείτε τα αποτελέσματα του στο TSV που υπάρχει αναρτημένο στο repository του project μας, πατώντας το παρακάτω κουμπί.

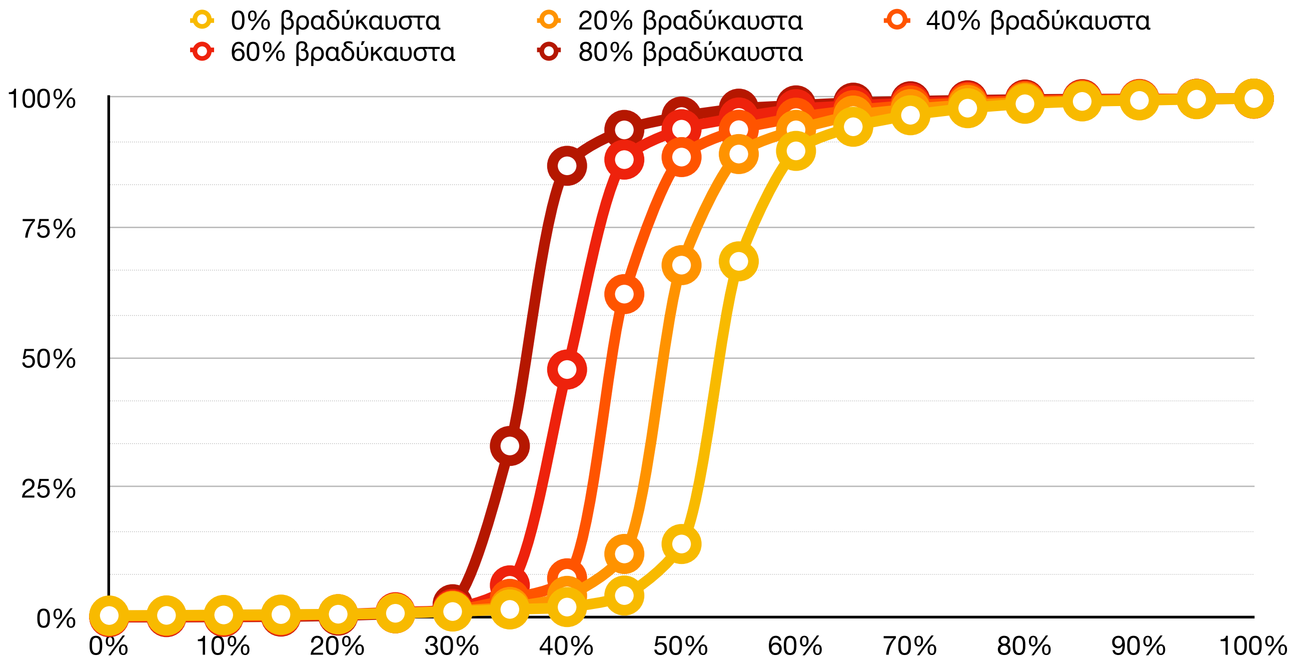
TSV με αποτελέσματα

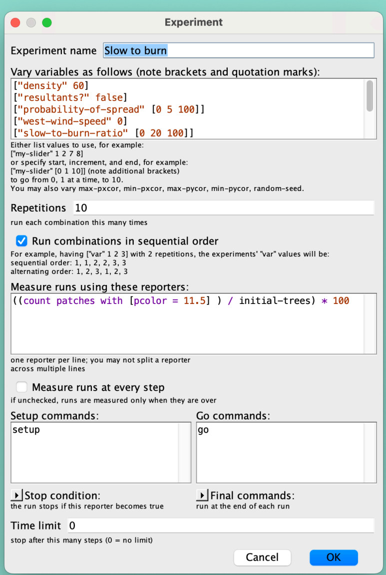
Προσθήκη βραδύκαυστων δένδρων

Η προσθήκη των βραδύκαυστων δέντρων έγινε με την εξής απλή τεχνική. Προστέθηκε ιδιότητα στα patches "`time-to-burn`". Προστέθηκε το slider "`slow-to-burn-ratio`" που ελέγχει το ποσοστό των δέντρων τα οποία είναι βραδύκαυστα. Τα βραδύκαυστα δέντρα έχουν "`time-to-burn` 2" και λίγο πιο σκούρο χρώμα για να ξεχωρίζουν.

Για να εξάγουμε συμπεράσματα σε σχέση με το πως τα βραδύκαυστα δέντρα επηρεάζουν το ποσοστό του δάσους που καίγεται, σχεδιάσαμε το πείραμα "Slow to burn" που μπορείτε να βρείτε στο εργαλείο BehaviorSpace.

Τα αποτελέσματα του πειράματος μας κάνουν ξεκάθαρο ότι τα βραδύκαυστα δέντρα βοηθάνε το δάσος να καεί! Μια διαισθητική ερμηνεία αυτού του αποτελέσματος είναι ότι κάθε βραδύκαυστο δέντρο έχει 2 ευκαιρίες να διαδώσει τη φωτιά. Κάθε τιμή της γραφικής παράστασης είναι στην πραγματικότητα μέσος όρος μεταξύ 10 εκτελέσεων των συγκεκριμένων ρυθμίσεων.





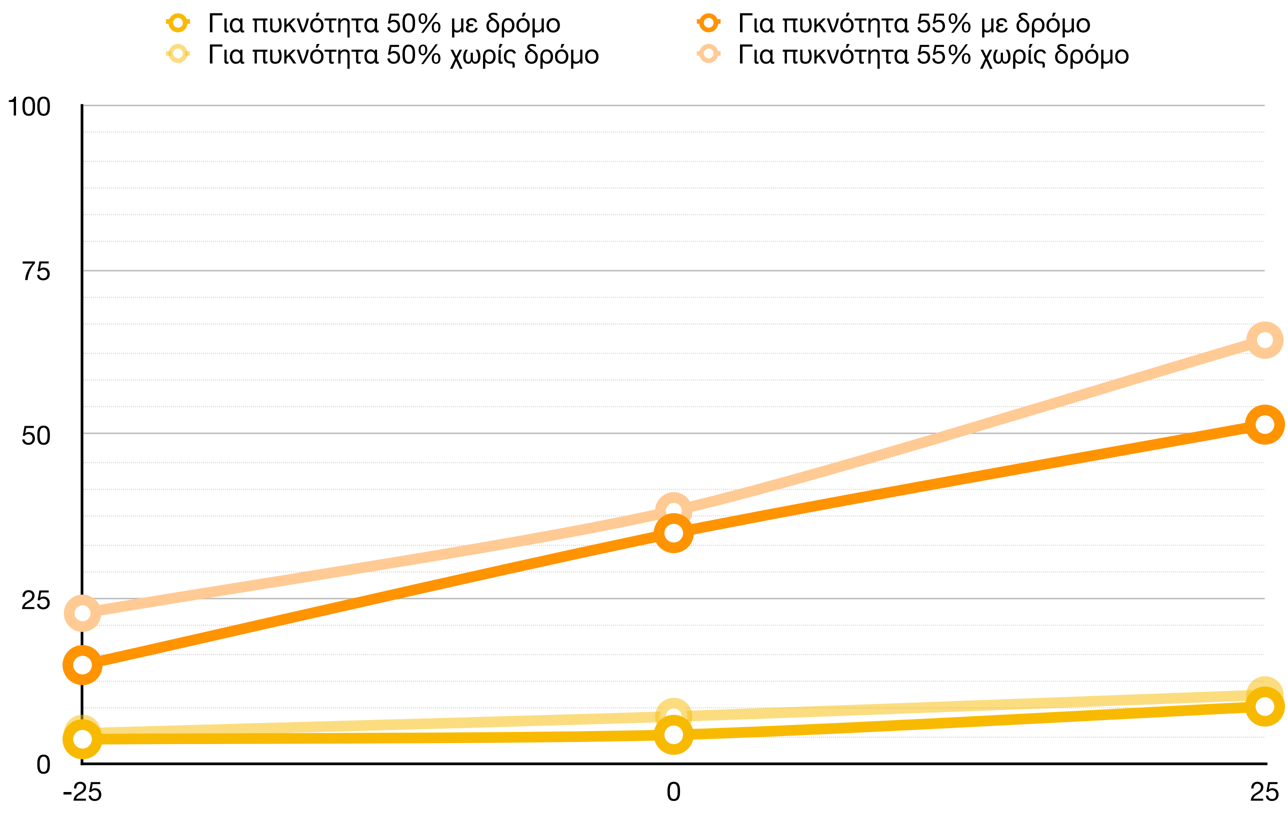
Παραθέτουμε τις ρυθμίσεις του πειράματος. Μπορείτε να δείτε τα αποτελέσματα του στο TSV που υπάρχει αναρτημένο στο repository του project μας, πατώντας το παρακάτω κουμπί.

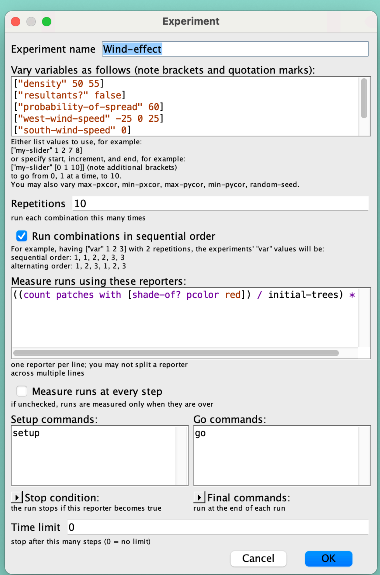
TSV με αποτελέσματα

Προσθήκη αγροτικού δρόμου

Ας εξετάσουμε το ενδεχόμενο να υπάρχει ένας αγροτικός δρόμος που να διασχίζει το δάσος κάθετα, μέχρι τη μέση του. Για να ενεργοποιήσετε τον αγροτικό δρόμο αρκεί να βάλετε το switch “`road?`” στο on. Εκτελούμε λοιπόν το πείραμα “Wind-effect” άλλη μια φορά, με ανοιχτό τον διακόπτη.

Τα αποτελέσματα μας δείχνουν ότι ο δρόμος βοηθάει στο να μειωθεί η διάδοση της φωτιάς. Αυτό δεν μας κάνει έκπληξη αφού δημιουργεί ένα τοίχος που εμποδίζει τα δέντρα στο πάνω αριστερά τεταρτημόριο να διαδώσουν την φωτιά στα δέντρα του πάνω δεξιού τεταρτημόριου. Κάθε τιμή της γραφικής παράστασης είναι στην πραγματικότητα μέσος όρος μεταξύ 10 εκτελέσεων των συγκεκριμένων ρυθμίσεων.



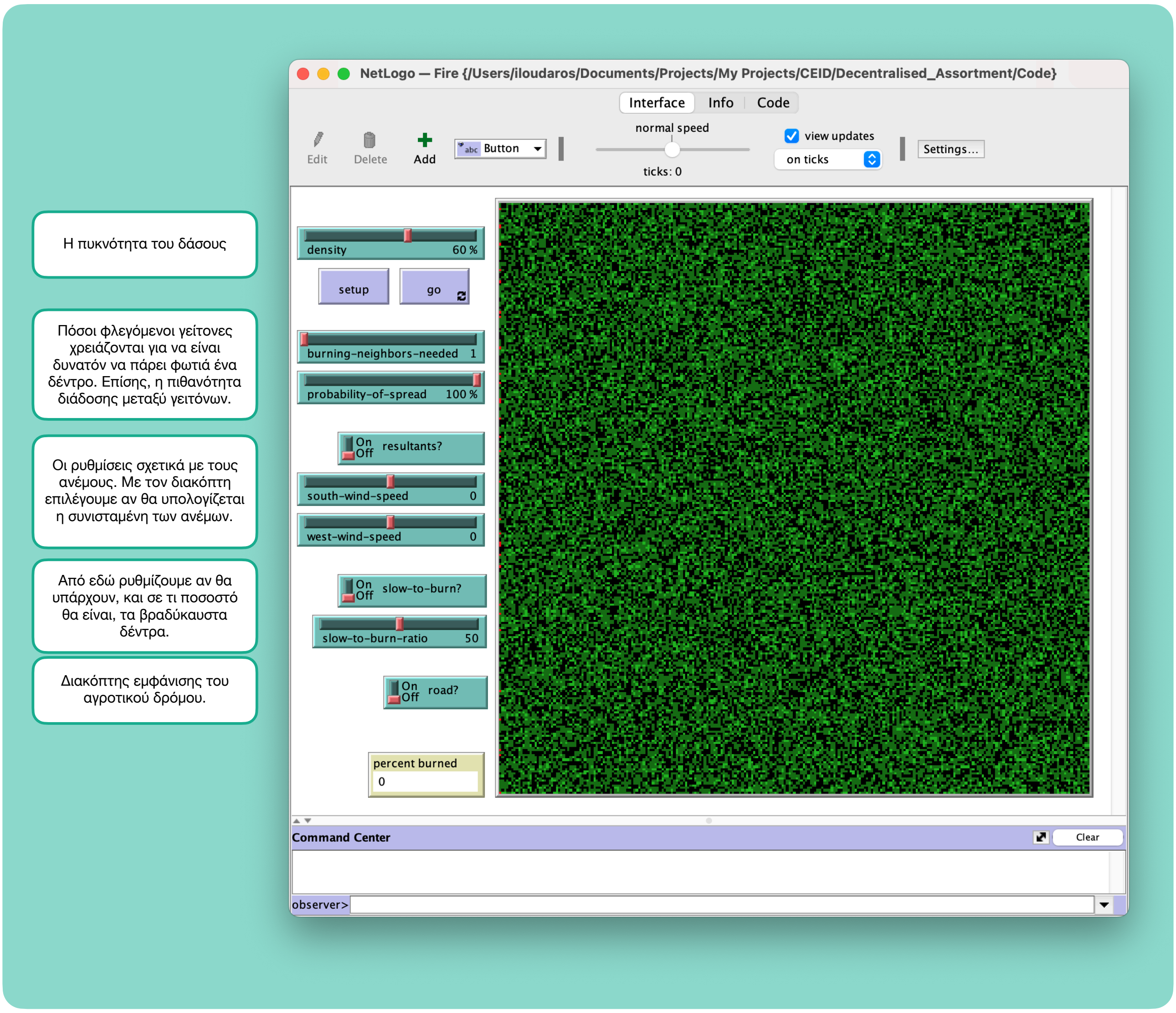


Παραθέτουμε τις ρυθμίσεις του πειράματος. Μπορείτε να δείτε τα αποτελέσματα του στο TSV που υπάρχει αναρτημένο στο repository του project μας, πατώντας το παρακάτω κουμπί.

TSV με αποτελέσματα

Συνδυασμός παραγόντων

Παρακάτω μπορείτε να δείτε το τελικό μοντέλο μας μαζί με επεξηγήσεις των ρυθμίσεων του.



Για να διερευνήσουμε την συμπεριφορά του μοντέλου ετοιμάσαμε το πείραμα “Critical density” που μπορείτε να βρείτε στο εργαλείο BehaviorSpace του μοντέλου. Το πείραμα εξερευνάει τις μεταβολές στην κρίσιμη πυκνότητα του δάσους, ανάλογα τον άνεμο, την πιθανότητα διάδοσης, τις διαφορετικές αναλογίες βραδύκαυστων δέντρων και την ύπαρξη ή όχι του αγροτικού δρόμου. Επίσης ετοιμάσαμε το “Critical density example” που είναι ένα πείραμα που εξερευνά μια ενδεικτική ρύθμιση του μοντέλου.

Βλέπουμε ότι η κρίσιμη πυκνότητα υπάρχει σε κάθε ρύθμιση του μοντέλου. Παρατηρούμε όμως ότι μεταβάλλεται ανάλογα με το πόσο ευνοείται ή όχι η εξάπλωση της φωτιάς. Αν ευνοείται η εξάπλωση, η κρίσιμη πυκνότητα μειώνεται, ενώ αν δεν ευνοείται η εξάπλωση, η κρίσιμη πυκνότητα μεγαλώνει. Για το Critical density example, βλέπουμε ότι η κρίσιμη πυκνότητα με δρόμο είναι κάπου στο 53%, ενώ χωρίς, είναι στο 52%.

Παράγοντες που βοηθούν την εξάπλωση : Μεγάλες πιθανότητες διάδοσης, δυτικός άνεμος, ύπαρξη βραδύκαυστων δέντρων.

Παράγοντες που εμποδίζουν την εξαπλωση : Μικρές πιθανότητες διάδοσης, ανατολικός άνεμος, απουσία βραδύκαυστων δέντρων, ύπαρξη δρόμων.

Εναλλακτικές Ασκήσεις Μοντελοποίησης

Ο Λαβύρινθος

Μπορείτε να τρέξετε το μοντέλο μας, ανοίγοντας το αρχείο “Maze.nlogo”. Για να το χρησιμοποιήσετε:

1. Διαλέξτε το επιθυμητό μέγεθος του λαβυρίνθου από το slider “`wsize`”.
2. Πατήστε setup.
3. Πατήστε το go
4. Μόλις δείτε ότι ο λαβύρινθος έχει σχηματιστεί, ξαναπατήστε το go.
5. Διαλέξτε το ύψος της εισόδου και της εξόδου από τα sliders “`Left-Height`” και “`Right Height`” αντίστοιχα.
6. Πατήστε τον solver που θέλετε να χρησιμοποιήσετε. Η εκτέλεση θα σταματήσει αυτόματα μόλις βρεθεί η έξοδος.

Ντετερμινιστική Στρατηγική

Για την πρώτη στρατηγική, ο πράκτορας ξεκινάει από το πράσινο patch και έχει σκοπό να φτάσει στο κόκκινο. Κάθε γύρο, ο πράκτορας κολλημένος στον τοίχο δεξιά του, κινείται μέσα στον λαβύρινθο, μέχρι να φτάσει στο τέλος του. Ο πράκτορας θα βρίσκει πάντα την λύση, λόγω της μορφής του λαβυρίνθου.

Στην γραφική παράσταση παρακάτω βλέπουμε τον χρόνο που απαιτείται για να λυθεί ο λαβύρινθος σε συνάρτηση με το μέγεθος του.

Στοχαστική Στρατηγική

Για την δεύτερη στρατηγική, ο πράκτορας ξεκινάει και πάλι από το πράσινο patch και έχει σκοπό να φτάσει στο κόκκινο. Κάθε γύρο, ο πράκτορας επιλέγει τυχαία μία κατεύθυνση και κοιτάζει προς αυτήν, και αν δεν υπάρχει τοίχος μπροστά του, κινείται προς την κατεύθυνση που κοιτάζει. Αφού η πορεία του είναι τυχαία, όσο περνάει ο χρόνος, τόσο αυξάνεται η πιθανότητα να φτάσει στο τέλος του λαβυρίνθου.

Παράλληλη Στρατηγική

Για την τρίτη στρατηγική, ο πράκτορας ξεκινάει και πάλι από το πράσινο patch και έχει σκοπό να φτάσει στο κόκκινο. Κάθε γύρο, ο πράκτορας δημιουργεί άλλους πράκτορες, προς όλες τις διαθέσιμες κατευθύνσεις που δεν έχουν ελεγχθεί, κρατώντας τον αριθμό πρακτόρων σε κάθε patch 4. Έτσι, καλύπτεται ολόκληρος ο λαβύρινθος με πράκτορες μέχρι κάποιος από αυτούς να βρει την έξοδο.

Επιδόσεις Αλγορίθμων

Παρακάτω παραθέτουμε τις γραφικές παραστάσεις με την επίδοση του κάθε αλγορίθμου. Η εκτίμηση της συμπεριφοράς του Ντετερμινιστικού και του Παράλληλου αλγορίθμου με την κλιμάκωση του λαβυρίνθου είναι ευκολότερη. Η διακύμανση των τιμών είναι μικρότερη και έτσι μπορέσαμε να βγάλουμε κάποιους αξιόπιστους μέσους όρους. Ο στοχαστικός αλγόριθμος έχει μεγάλη διακύμανση και έτσι η συμπεριφορά του είναι λιγότερο προβλέψιμη. Επιπλέον, η τυχαιότητα του λαβυρίνθου, ακόμη και στην εύκολη δυσκολία, δυσκολεύει τις εκτιμήσεις μας. Παρόλα αυτά, στις γραφικές παραστάσεις προσπαθούμε να δώσουμε μια ενδεικτική εικόνα.

Όπως θα παρατηρήσετε, λόγω του πολύ μεγάλου ρυθμού αύξησης του απαιτούμενου χρόνου για τον Ντετερμινιστικό και τον Στοχαστικό αλγόριθμο, χρειάστηκε να τους αναπαραστήσουμε σε λογαριθμική κλίμακα. Αντίθετα, ο παράλληλος έχει τόσο καλή συμπεριφορά, που δεν μπορούσαμε να τον αναπαραστήσουμε στο ίδιο γράφημα με τους υπόλοιπους δύο.