

Da insiemi di agenti a sistemi swarm adattativi

Progetto

Il progetto complessivo consta in una serie di realizzazioni di (ed esperimenti su) un modello multiagente; ogni nuova versione del modello è più raffinata della precedente. La versione iniziale non è un sistema collettivo; la versione finale realizza un sistema collettivo i cui elementi sono capaci di evolvere e di adattarsi all'ambiente.

1. Il progetto base (termiti)

Le termiti, muovendosi in modo pseudo-casuale, prendono degli oggetti e li spostano verso un altro punto, creando dei magazzini.

Ambiente

Nell'ambiente toroidale sono posizionati in modo casuale degli oggetti (cibo); in base alla variabile *food_density* la presenza degli oggetti nell'ambiente sarà più o meno intensa.

Agenti

Ogni agente può portare o meno del cibo e muovendosi in modo casuale all'interno dell'ambiente esegue due semplici azioni: se passa sopra del cibo lo prende con una certa probabilità (*laydown_perc*) e si allontana dalla zona di un certo numero di passi (*rg_inerzia*), se invece l'agente è carico (colore arancione) e il punto in cui si trova è libero, esso lascia il cibo in quel punto e si allontana di un certo numero di passi (*rg_libera*). In quest'ultimo comportamento è presente anche una caratteristica un po' meno esplicita, quando l'agente ha appena lasciato il cibo esso si allontana controllando di non essersi posizionato nuovamente in un punto con del cibo. Questa dinamica può essere paragonata ad una sorta di "memoria", poiché se l'agente ha appena lasciato del cibo è meglio che esplori un po' l'ambiente prima di raccoglierne dell'altro. Un ulteriore parametro dell'agente è l'angolo di virata (*turn_angle*) ovvero l'angolo massimo col quale si può girare ad ogni passo.

Dinamica del sistema

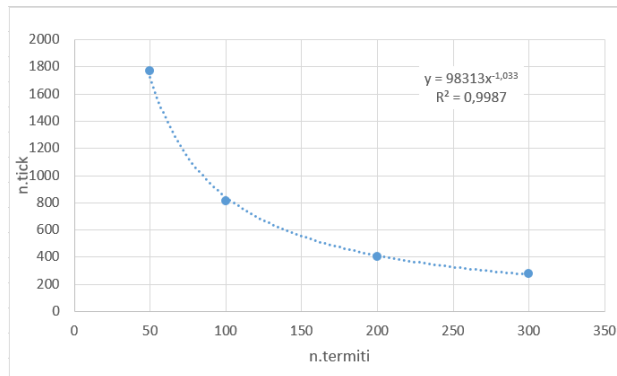
Il comportamento del sistema favorisce la formazione di magazzini, questo a prescindere da come sono impostati i valori dei parametri anche se valori più "estremi" portano ad una formazione di magazzini più blanda in cui è meno chiaro il loro profilo e la loro dimensione. In questo sistema non esiste una esplicita comunicazione tra gli agenti, come anche una definizione precisa di "memoria".

Feedback del sistema

Il tutto si può sintetizzare nei parametri *rg_libera* e *rg_inerzia*; analizzando il primo si può dire che un agente allontanandosi dal punto in cui ha appena lasciato il cibo, presumibilmente un magazzino, vada in una zona in cui cercare del nuovo cibo. Similmente per il secondo se l'agente si allontana da una zona in cui ha colto del cibo finirà in una zona differente, cioè probabilmente un magazzino

Esperimenti

Le termiti non sono un esempio di “sistema collettivo”: il lavoro di 1000 termiti in un tempo T, può essere fatto da 10 termiti in un tempo 100 volte T.



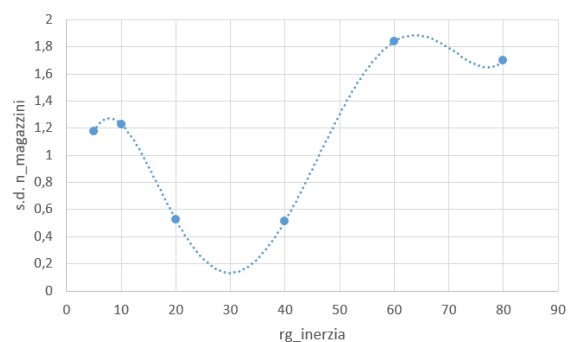
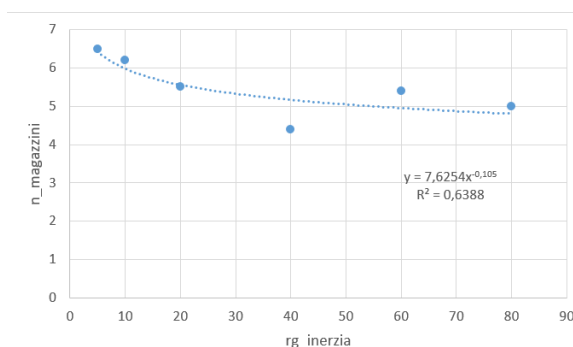
Si può evidentemente notare la relazione proporzionalmente inversa tra numero di agenti e tempo per realizzare 6 magazzini, con 50 termiti sono necessari 1800 tick mentre con 100 poco meno della metà, 800 tick.

Di seguito si riportano diversi esperimenti sui singoli parametri, variandoli uno alla volta per vedere l'effetto sull'intero sistema. Per tutti gli esperimenti si eseguono 10 simulazioni del sistema contando il numero di magazzini a tick uguale 1000, i valori di default sono i seguenti:

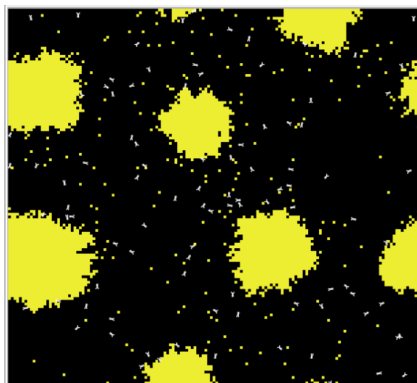
- Numero termiti: 100
- *food_density*: 20%
- *rg_libera*: 20
- *rg_inerzia*: 20
- *turn_angle*: 20

Gli stessi valori sono utilizzati nel primo esperimento, tranne che per il numero di termiti che è il parametro che si vuole analizzare in quel caso. Per ogni esperimento si riporta anche il grafico della deviazione standard.

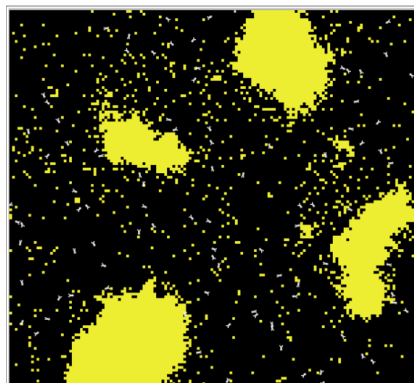
Esperimento su *rg_inerzia*



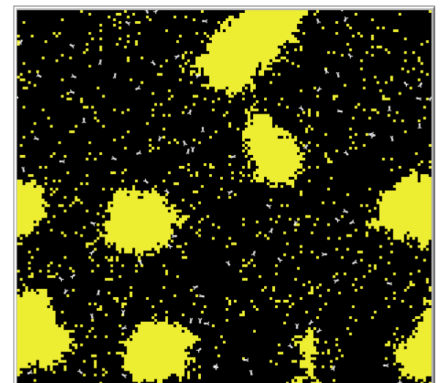
Il parametro *rg_inerzia* corrisponde al numero di passi compiuti dall'agente dopo aver raccolto del cibo. Al variare del parametro non si nota una netta diminuzione nel numero di magazzini, inoltre la deviazione standard aumenta all'aumentare del parametro.



rg_inerzia = 20



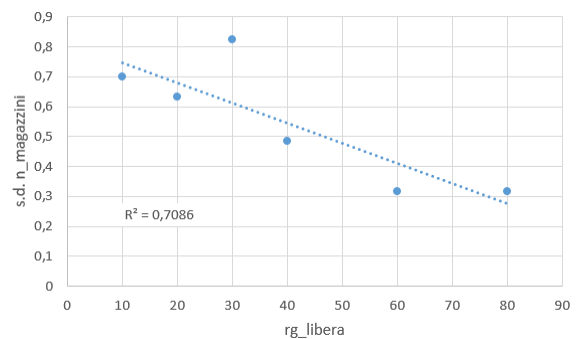
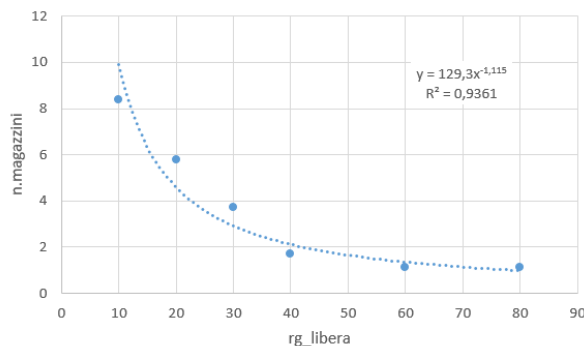
rg_inerzia = 40



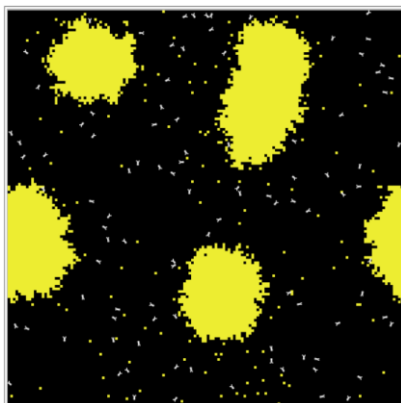
rg_inerzia = 60

Si può fare anche un'ulteriore osservazione più empirica; all'aumentare del valore i rimasugli sparsi di cibo sembrano aumentare, portando anche alla formazione di magazzini molto piccoli o in alcuni casi con una forma non ben definita.

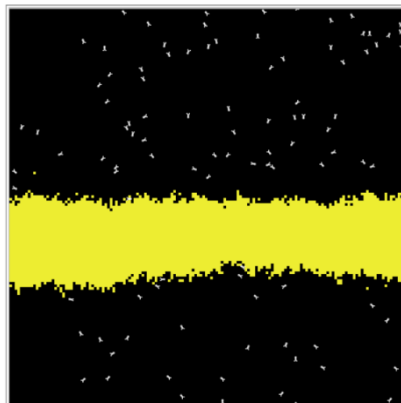
Esperimento su *rg_libera*



Il parametro *rg_libera* sembra invece molto più adeguato nel portare alla stabilità il sistema, per valori maggiori o uguali a 40 si crea quasi sempre un solo magazzino e il fenomeno dei rimasugli sparsi è molto limitato se non del tutto assente già per valori superiori a 20.



rg_libera = 30



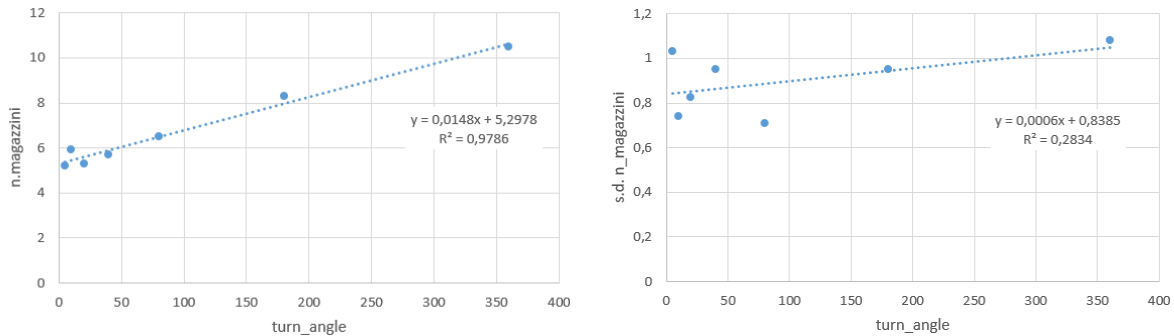
rg_libera = 60



rg_libera = 80

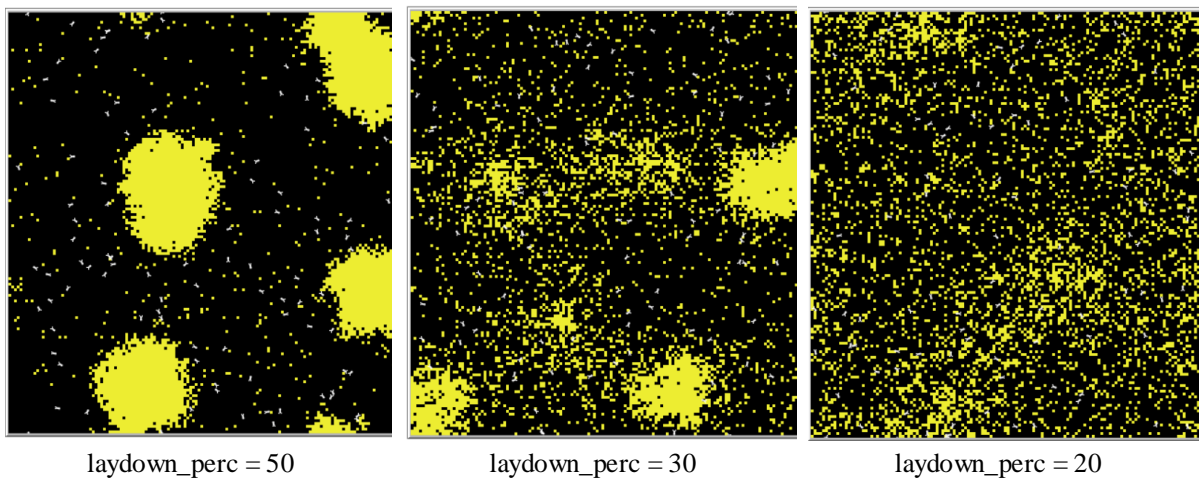
Si può notare come al crescere del valore il magazzino sia unico e senza cibo sparso; inoltre per valori maggior o uguali a 60 in molti casi si creano dei magazzini a forma allungata, verticali o orizzontali.

Esperimento su *turn_angle*

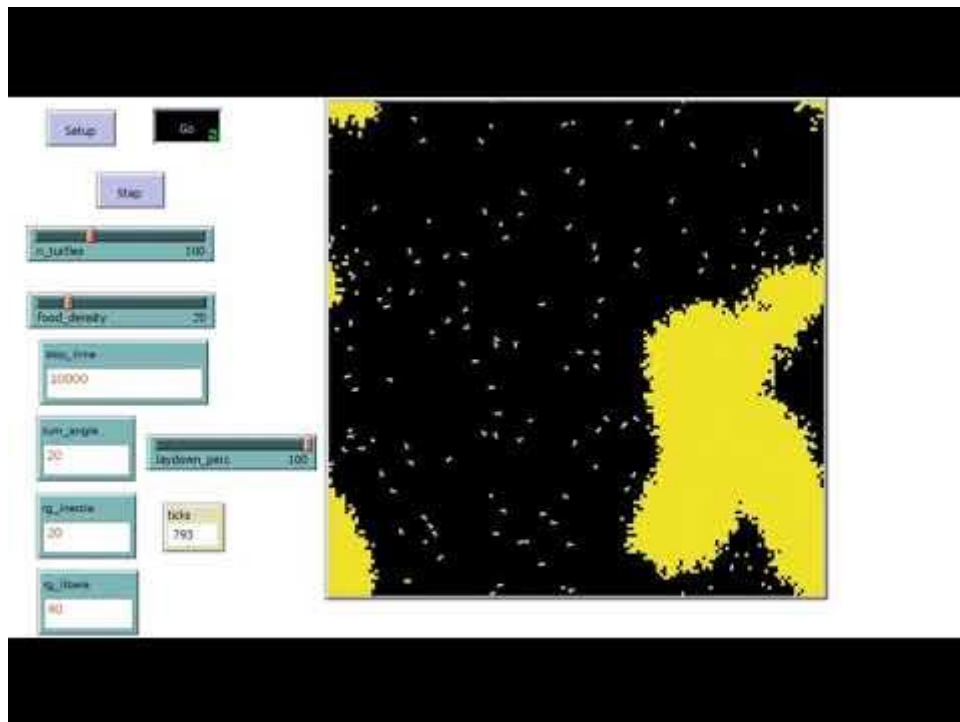


Il parametro *turn_angle* è l'unico che aumentando fa aumentare anche il numero di magazzini, esso però ha una leggera differenza rispetto ai parametri analizzati prima. Mentre *rg_libera* e *rg_inerzia* rappresentano il numero esatto di passi che vengono compiuti, *turn_angle* è più paragonabile ad un range di valori; fissato un valore x si sceglie in modo random un valore nell'intervallo $]0, x]$.

Esperimento su *laydown_perc*



Riguardo al parametro *laydown_perc* si può fare un'osservazione, sempre abbastanza empirica, sulla struttura dei magazzini che si vengono a creare. Al diminuire di questa percentuale i magazzini iniziano ad avere una forma che presenta alcuni buchi al loro interno, questo fenomeno si inizia a notare per valori inferiori a 50, fino ad arrivare intorno al 20% valore per il quale il sistema non forma praticamente nessuna struttura paragonabile ad un magazzino.

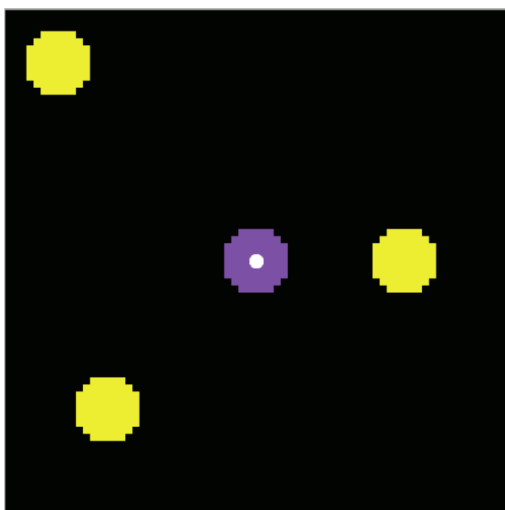


Come si vede nell'esperimento sopra, modificando il valore di *laydown_perc* si può portare il sistema dall'ordine al disordine e viceversa.

2. Stigmergia

Nido e termiti rilasciano ciascuno un proprio odore: il modello diventa un sistema "formicaio". Il formicaio è il primo esempio di sistema collettivo (sono attivi feedback fra gli agenti tramite l'ambiente, ed il comportamento collettivo è quindi diverso dal lavoro di un singolo agente), in grado di sfruttare diverse sorgenti di materiale.

Ambiente



Nell'ambiente, di tipo toroidale, sono posizionati tre cumuli di cibo in tre punti diversi, al centro è invece presente il nido, il punto da cui le termiti inizieranno la loro ricerca per il cibo e anche

il punto in cui esse torneranno una volta averlo raccolto. L'ambiente, inoltre, accoglie il feromone rilasciato dalle termiti diffondendolo (*diffusion_rate*) e col passare del tempo lo fa evaporare con una certa intensità (*evaporation_rate*). Il nido emette anch'esso una traccia olfattiva, via via minore all'aumentare della distanza da esso, questo "odore" servirà alle termiti per dirigersi verso la posizione del nido.

Agenti

Se una termite è scarica si muove in modo casuale in cerca di cibo; una volta trovato ha l'obiettivo di portarlo al nido. Essendo che il nido rilascia del feromone, le termiti cariche, ad ogni passo, individuano il punto in cui la traccia è più forte e si dirigono verso quella posizione. Nel riportare il cibo al nido le termiti rilasciano anche esse del feromone, che servirà agli altri agenti per individuare la scorta di cibo. Le termiti scariche "annusano" l'ambiente circostante e si girano di 45° nella direzione in cui la traccia è più intensa.

Dinamica del sistema

Il sistema converge in qualsiasi circostanza, i parametri di evaporazione e diffusione influiscono solo sul tempo impiegato dalle termiti ad esaurire tutte le scorte di cibo. Quando un'agente carico ritorna al nido rilascia del feromone creando una sorta di percorso dal nido verso la scorta di cibo, la dimensione e la permanenza nell'ambiente di questo percorso dipendono dai parametri *evaporation_rate* e *diffusion_rate*. Benché si possa pensare che, intuitivamente, l'evaporazione sia nociva nei confronti degli agenti mentre la diffusione del feromone li aiuti nella creazione dei percorsi, entrambi questi parametri possono migliorare o peggiorare la velocità di convergenza del sistema in funzione del loro valore. Una diffusione molto intensa del feromone può creare dei percorsi troppo "vasti" e molto distanti dalle risorse di cibo, in queste circostanze gli agenti potrebbero seguire delle tracce che non conducono al cibo. L'evaporazione invece può aiutare ad "eliminare" i percorsi che non sono più utili, facendo sì che il sistema si focalizzi solo sulle risorse ancora presenti; di contro un'evaporazione troppo insistente impedisce completamente la creazione dei percorsi.

Feedback del sistema

L'interazione tra gli agenti non è diretta, essi comunicano tramite il feromone rilasciato nell'ambiente dalle termiti cariche di cibo, per questo parliamo di stigmergia. Nell'eseguire la simulazione si nota come, solitamente, la prima scorta a essere consumata è quella nella parte "destra/est" dell'ambiente; questo è dovuto al fatto che essendo la più vicina al nido, le termiti, muovendosi casualmente, hanno una maggiore probabilità di incontrarla per prima e di crearci di conseguenza un percorso.

Esperimenti

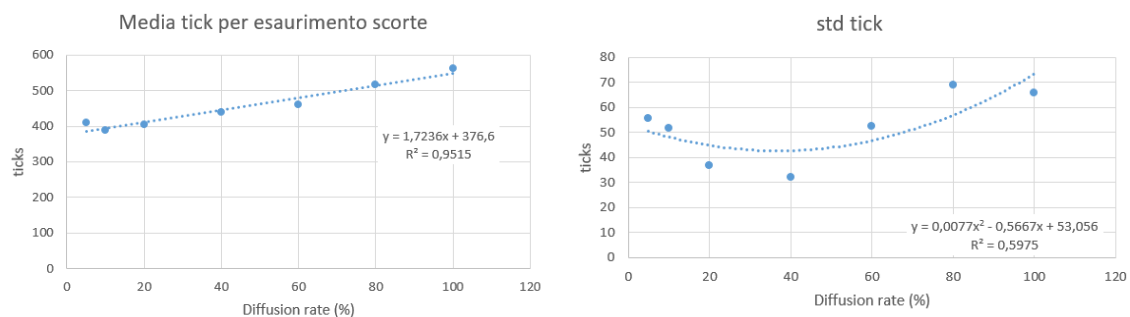
Di seguito si propongono degli esperimenti per cercare di individuare i valori ottimali dei parametri *evaporation_rate* e *diffusion_rate*. Per fare questo si analizza in quanti tick le termiti esauriscono tutte le scorte di cibo; per ridurre al minimo l'errore umano, nella simulazione è stato inserito un controllo che riporta il valore dei tick nel momento in cui l'ultima patch di cibo è stata presa da un'agente. Per ogni esperimento si effettuano 10 lanci della simulazione e si riporta anche il grafico della deviazione standard, si osserverà che questa ha valori molto alti. Nella simulazione si può notare che le termiti impiegano molto tempo nel trovare gli ultimi frammenti di cibo delle scorte, poiché questi restano isolati e con nessuna scia di feromone

vicina ad essi. In via del tutto empirica possiamo dire che in corrispondenza dei valori che si discostano molto dalla media, la maggior parte di quel tempo “in più” è utilizzato per cercare i frammenti di cibo rimasti isolati.

I valori di default della simulazione sono i seguenti:

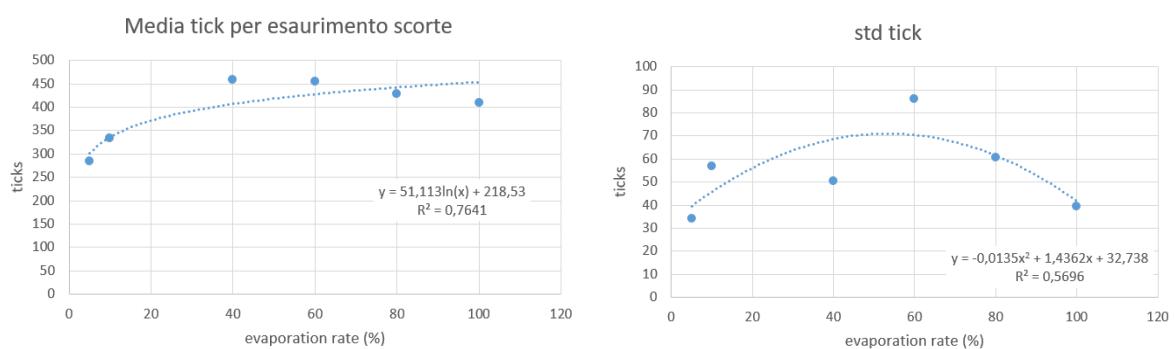
- Numero termiti = 125
- Evaporation rate = 20%
- Diffusion rate = 20%

Esperimento su *diffusion rate*



Si può notare come l'andamento sia nettamente lineare, all'aumentare del *diffusion rate* aumenta anche il tempo impiegato dalle termiti a esaurire tutte le scorte di cibo. Il valore ottimale di *diffusion rate* è 10%.

Esperimento su *evaporation rate*



Nel caso dell'*evaporation rate* non si può identificare una forte linearità, ad ogni modo è stato impiegato meno tempo in corrispondenza del valore più basso: *evaporation rate* al 5%.

3. Selezione

Ogni formica ha proprie risorse (quantificate e memorizzate in una variabile apposita) ed un proprio metabolismo (la quantità di risorse che viene consumata ad ogni mossa); ogni formica che torna al nido portando del cibo viene ricaricata. Le formiche che hanno risorse troppo basse (minori di zero nelle unità del programma) vengono sostituite da altre, create a caso (basta re-inizializzare le variabili della formica al valore iniziale – ricordandosi di cambiare i valori delle variabili inizializzare casualmente). L'effetto finale consiste nella scomparsa delle formiche meno adatte ai propri compiti

Ambiente

L'ambiente è uguale al precedente, con l'unica differenza che ogni 70 *tick*, le scorte di cibo vengono rigenerate, sempre nella stessa posizione.

Agenti

Gli agenti si muovono sempre in maniera casuale, in questa simulazione viene impostato un valore di *virata massima*; quando le formiche vengono generate gli si assegna un valore di *virata* nel range $[1, \text{virata massima}]$, di conseguenza ad ogni movimento esse possono scegliere, casualmente, nel range $[1, \text{virata}]$. In questa simulazione gli agenti hanno due proprietà in più rispetto ai precedenti: *scorta* e *metabolismo*. Ad ogni passo ogni formica preleva dalla *scorta* una certa quantità di risorse corrispondente al suo *metabolismo*, la *scorta* viene ricaricata ogni volta che l'agente torna al nido e se scende sotto 0 provoca la morte dell'agente stesso, quando un'agente muore se ne genera un altro. È possibile attivare, per tutte le formiche, una proprietà che lega la velocità (*steps*) al *metabolismo*; se si attiva questo legame gli agenti con velocità 1 potranno avere un metabolismo da 1-3, mentre se la velocità è 2 il metabolismo potrà variare tra 3-5. Questo legame vuole simulare il consumo di risorse in base alle caratteristiche "fisiche" dell'individuo, come uno sportivo ha bisogno di assumere, mediamente, più calorie, allo stesso modo gli agenti dovranno consumare di più se si muovono più velocemente. Il valore di *scorta* è impostato globalmente per tutte le formiche e può essere modificato, inoltre esse hanno una *data di nascita* che corrisponde al valore di tick nel momento in cui vengono generate, questo servirà per calcolare la vita media della popolazione.

Dinamica del sistema

Il comportamento del sistema non differisce molto dal precedente, le formiche riescono sempre a trovare il cibo creandoci dei percorsi più o meno efficienti. Nel sistema è possibile attivare o meno la morte degli agenti, ovvero quando la loro *scorta* scende sotto 0 essi non si rigenerano.

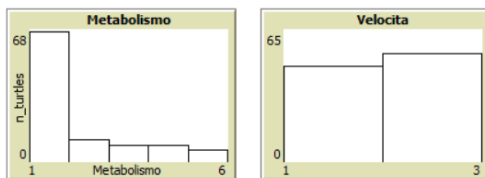
Feedback del sistema

Il sistema non prevede la riproduzione degli individui, infatti se si attiva la morte degli agenti col passare del tempo la popolazione si estingue, questo vuol dire che per quanto una formica possa essere efficiente non sarà mai "immortale".

Esperimenti

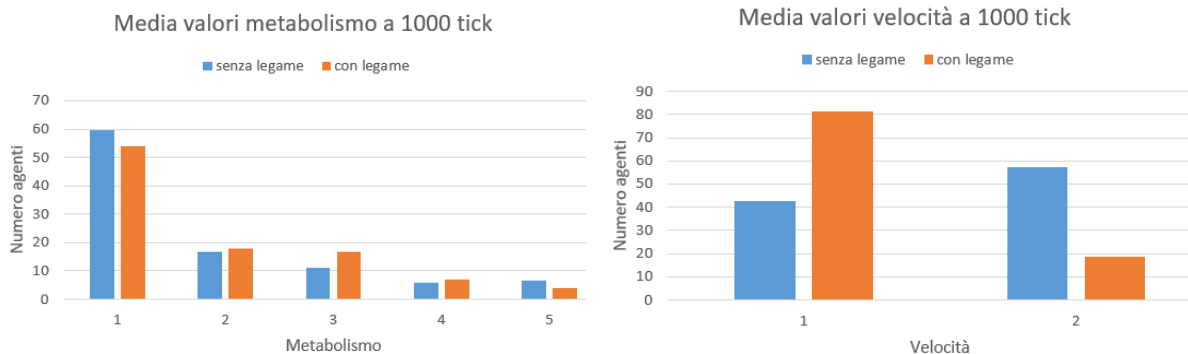
Le impostazioni default del sistema sono le seguenti:

- Numero agenti: 100
- Diffusion rate: 20%
- Evaporation rate: 20%
- Angolo virata massima: 40
- Scorta massima: 200
- Legame metabolismo – velocità: **off**
- Morte degli agenti: **off**

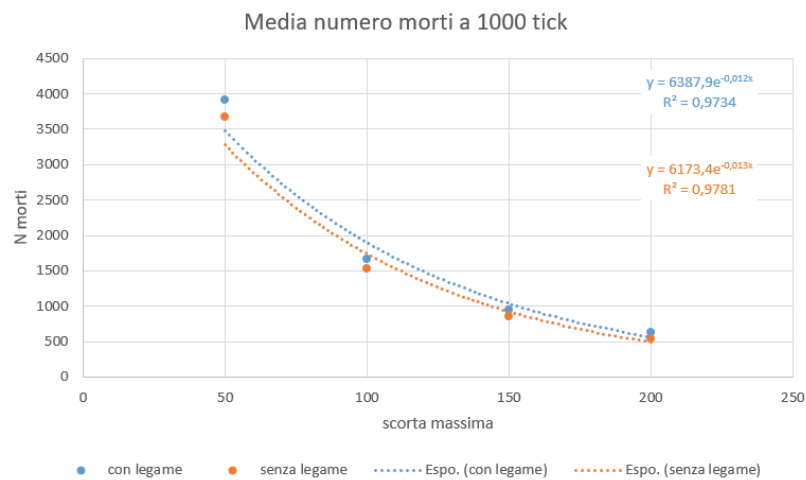


Con queste impostazioni si può osservare che il metabolismo più prediletto è 1, mentre il valore di velocità è equamente distribuito tra 1 e 2, con una leggera propensione per quest'ultimo. La vita media di ogni agente varia tra 120 e 135 tick mentre in media ogni agente ha una scorta intorno a 120.

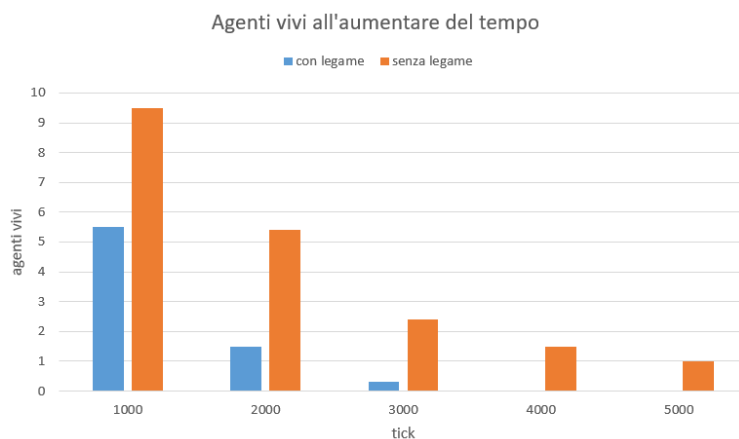
Di seguito si riportano le distribuzioni del *metabolismo* e delle *velocità* quando il legame tra le due proprietà è attivo o meno.



Osservando il *metabolismo*, non ci sono cambiamenti sostanziali, gli unici due di interesse sono in corrispondenza di 1 e 3. Nel primo il numero di agenti diminuisce, probabilmente proprio a favore del secondo poiché la configurazione con *metabolismo* = 3 e *velocità* = 2, è quella in cui si ottiene il massimo risultato con lo sforzo minore. Per quanto riguarda le velocità c'è invece un'inversione di tendenza, nella situazione **con** legame si predilige di molto la velocità 1, indicando che il sistema propende verso un consumo minore anziché verso una maggiore mobilità ed esplorazione.



Il grafico riporta il numero di morti a 1000 tick al diminuire della scorta, per il sistema con e senza legame *metabolismo-velocità*. Entrambe le casistiche hanno, ovviamente, lo stesso andamento, il numero di morti tuttavia è leggermente inferiore nel sistema **senza** legame che sembra risultare meno oppressivo per gli agenti, i quali non devono “pagare” un consumo maggiore per avere più libertà di movimento.



Con il passare del tempo il numero di formiche diminuisce dato che non è in atto nessun meccanismo di riproduzione. Come era intuibile dall'andamento dei morti a 1000 tick, il sistema senza legame ha una sopravvivenza maggiore, infatti quando è presente il legame la popolazione non sopravvive oltre i 3000 tick.

4. Evoluzione

Le formiche di successo (che hanno cibo ed abbastanza risorse) possono cercare intorno a sé altre formiche con abbastanza risorse e creare un'altra formica, che avrà metabolismo e velocità ognuno preso a caso da uno dei due genitori. Il processo è dispendioso, e la formica perde parte delle proprie risorse (1/10 delle iniziali).

Ambiente

Uguale al precedente

Agenti

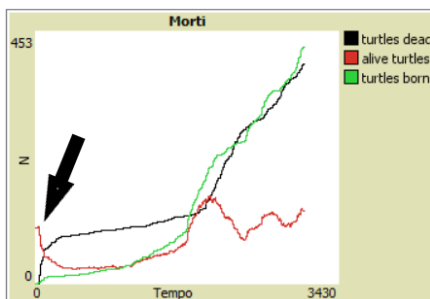
Il comportamento degli agenti è lo stesso, se non per la riproduzione. Una formica può riprodursi solo se sta portando del cibo e se la sua scorta rimanente è maggiore del 60% della sua scorta massima. Sotto queste condizioni se due formiche si incontrano, ne generano un'altra che erediterà casualmente metà patrimonio da entrambi; per riprodursi le formiche consumano il 10% della loro scorta massima. In questo caso quando un'agente ha scorta minore di 0, muore senza generarne un altro. Anche in questa simulazione si può introdurre il legame tra *metabolismo* e *velocità*, se esso è presente i geni saranno distribuiti in modalità diversa all'atto della riproduzione. Per prima cosa il figlio erediterà casualmente la *velocità* da uno dei due genitori e conseguentemente anche il metabolismo sarà ereditato dallo stesso genitore, in pratica, quindi, si sceglie casualmente da quale genitore erediterà tutto il patrimonio.

Dinamica del sistema

Anche in questo caso il sistema si comporta nello stesso modo del precedente.

Feedback del sistema

Nonostante il comportamento sia equivalente al precedente, in questo sistema abbiamo un'oscillazione della popolazione; il rapporto agenti vivi/morti si mantiene costante nel tempo.

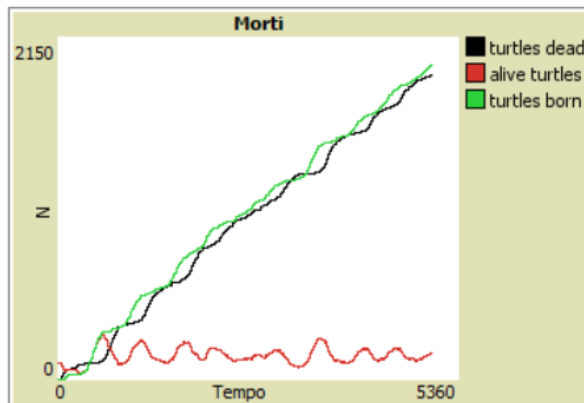


All'inizio il sistema presenta una fase di forte discesa della popolazione, tuttavia quando gli agenti iniziano a riprodursi si innesca una sorta di reazione a catena che permette alla popolazione di riacquisire individui, fino alla stabilizzazione di morti/vivi.

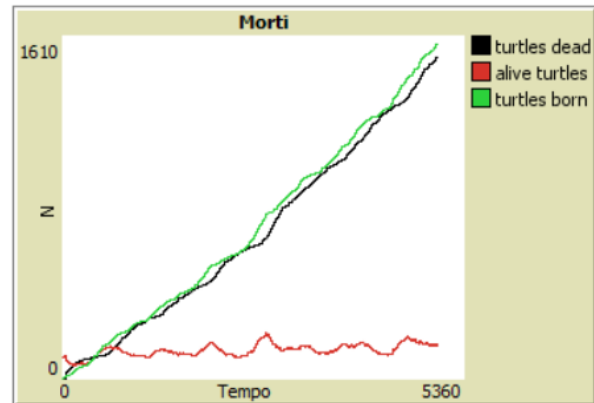
Esperimenti

Le impostazioni di default del sistema sono come le precedenti se non per *evaporation rate* e *diffusion rate* che sono rispettivamente 5% e 10%, questa modifica serve per evidenziare meglio alcuni comportamenti del sistema.

La distribuzione della velocità dipende quasi unicamente dalla configurazione iniziale degli individui, infatti in base a questa tutti gli agenti acquisiscono velocità 1 o 2 e solo in alcuni casi queste due si distribuiscono equamente sulla popolazione. Per quanto riguarda il metabolismo, in tutti i casi la popolazione tende al valore 1. Questo cambia se si attiva il legame *metabolismo-velocità*, infatti in questo caso la configurazione a cui tende la popolazione è unica in tutti i casi, ovvero $\text{metabolismo} = 1$ e $\text{velocità} = 1$.



Legame metabolismo-velocità **On**



Legame metabolismo-velocità **Off**

Si può notare come l'andamento della popolazione sia molto più ondulatorio quando il legame è attivo, questo fenomeno lo si può analizzare meglio osservando l'andamento della popolazione nel tempo.

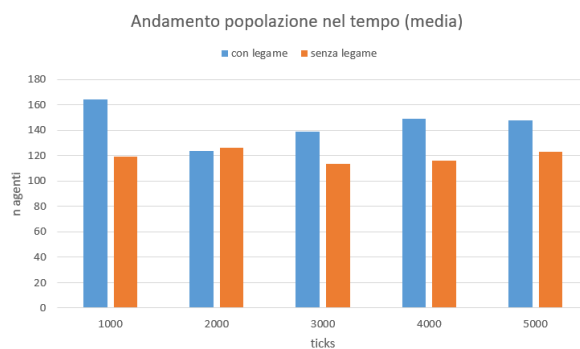


fig.1

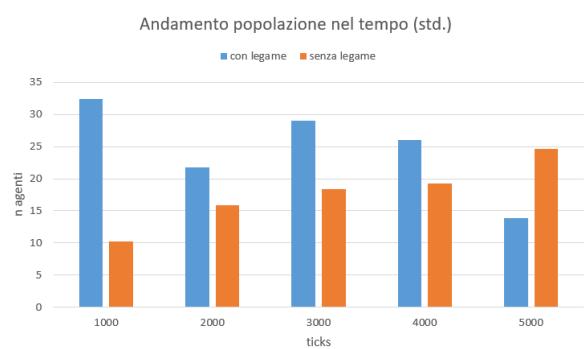


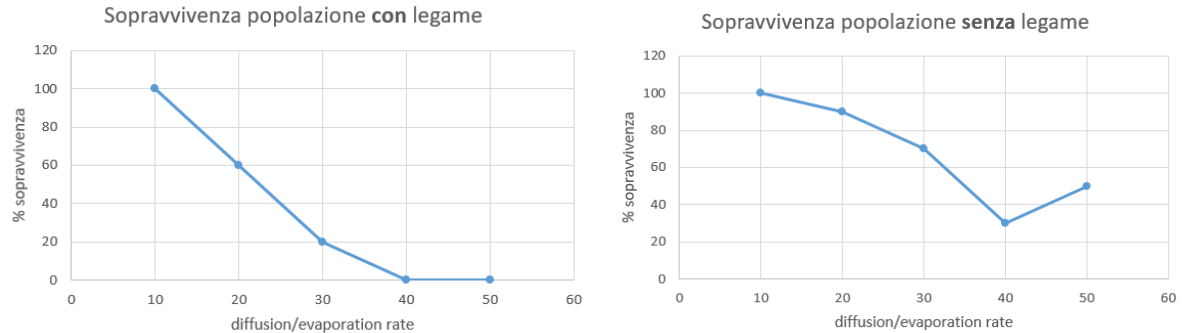
fig.2

L'andamento medio della popolazione (fig.1) è più o meno costante per entrambi i sistemi con e senza legame, con valori in generale maggiori per il primo.

Analizzando invece la deviazione standard (fig.2), all'aumentare dei tick, nel sistema con legame la popolazione sembra "stabilizzarsi" mentre, al contrario, nel sistema senza legame la varianza aumenta costantemente.

In ultima istanza si vuole analizzare in che circostanze la popolazione tende verso l'estinzione, nell'esperimento proposto si aumentano parallelamente *diffusion* e *evaporation* rate per osservare come gli agenti si comportano in un ambiente più "ostile".

Per estinzione si intende un numero di individui non superiore al 10% della popolazione di partenza.



Nella popolazione con legame l'estinzione si raggiunge per rates superiori al 40%, mentre nell'altro caso c'è una buona sopravvivenza (30%-50% della popolazione) anche per valori superiori al 40%.

La poca sopravvivenza della popolazione con legame si potrebbe spiegare nel seguente modo: in un ambiente considerevolmente oppressivo, la popolazione non riesce a passare abbastanza velocemente i caratteri ereditari migliori alla prole, in pratica, riferendosi al primo grafico, la specie non sopravvive al primo picco di discesa della popolazione.