

Simulation Model for Crowd Behaviour

Lorenzo Betti, Francesco Bodria, and Lorenzo Chicchi

Università degli studi di Torino
<http://gamacrowdcontrol.com/>

1 Introduzione

Eventi che prevedono l'affluenza di un gran numero di persone possono risultare problematici da un punto di vista della sicurezza. Quando viene percepita una situazione di pericolo le persone tendono a scappare velocemente riportando danni, a volte letali. Risulta quindi interessante lo studio del comportamento di una folla in tali situazioni per poter poi organizzare al meglio la gestione degli spazi e delle relative uscite di emergenza. La pericolosità di queste dinamiche rende impossibile svolgere degli esperimenti in situazioni reali e per questo diventa particolarmente utile riuscire a simulare adeguatamente questi fenomeni, guidati dalle informazioni che è possibile ricavare da immagini, video o testimonianze relative ad eventi realmente accaduti [4].

Diversi lavori hanno cercato di modellizzare il movimento di una folla analogamente al moto di un fluido, descrivendo il comportamento di una folla da un punto di vista macroscopico [2]. In questi termini ciascuna persona è sottoposta alle leggi della fisica che governano il moto dei fluidi e il loro insieme viene considerato come un'unica entità. Questi modelli si concentrano principalmente sulla pianificazione di uscite di emergenza e non prendono in considerazione l'interazione tra singoli individui, ciascuno dei quali potrebbe essere considerato come un'entità autonoma.

Un'altra opzione consiste nel tentativo di descrivere il comportamento della folla a partire dal singolo individuo, modellizzando il processo decisionale del singolo come risposta a stimoli che arrivano dall'ambiente e dagli altri individui [3]. In questa ottica, la formazione ed il moto di una folla diventa un fenomeno emergente che ha origine dalla definizione di regole che determinano il comportamento dei singoli componenti del sistema. Seguendo questo schema è possibile indagare il ruolo che ha il comportamento umano nella formazione e nel moto delle folle in situazioni di panico. La metodologia computazionale che permette di integrare questi aspetti è la Simulazione a Multi Agenti (MAS) con la quale è possibile definire un certo numero di agenti autonomi con delle regole che permettono di farli interagire.

Il modello che sfrutta l'analogia con fenomeni di idrodinamica è più adatto a descrivere situazioni in cui la densità della folla è estremamente alta ma risulta più debole nel momento in cui si vogliono riprodurre situazioni che si osservano in condizioni reali. Ad esempio, in presenza di due uscite questo modello prevede che entrambe le uscite vengano ugualmente sfruttate invece in situazioni reali si osserva che un'uscita risulta congestionata mentre l'altra viene usata di meno. Questo può essere dovuto al fatto che un individuo in una situazione di panico si limita a seguire le persone che ha intorno, non rendendosi conto che l'uscita alternativa possa essere libera. Oltre a questo il modello si basa sulle equazioni dell'idrodinamica che sono più difficili da maneggiare. Al contrario modelli MAS si basano su regole più semplici e possono tener conto di forze sociali in grado di riprodurre il comportamento relativo all'utilizzo delle due uscite di emergenza descritto sopra. Questi modelli sono tuttavia molto dispendiosi dal punto di vista computazionale perché richiedono di elaborare simultaneamente il comportamento di tutti gli agenti presenti nella simulazione.

Nel presente lavoro abbiamo riprodotto un modello MAS [6][7] in cui le persone sono modellizzate come agenti reattivi che interpretano dei segnali provenienti dall'ambiente, relativi alla loro posizione o alla presenza di altri agenti nelle loro vicinanze, per decidere dove spostarsi su una mappa discreta in due dimensioni. Oltre all'informazione che ha l'agente sulla posizione relativa all'uscita più vicina, il modello cerca di integrare altri due comportamenti che risultano importanti nel fenomeno che cerchiamo di riprodurre: forze sociali e forze di spinta. Le prime si riferiscono all'influenza che può essere attribuita a persone vicine che si stanno muovendo, modellizzata prendendo ispirazione dal fenomeno della chemiotassi (fenomeno secondo il quale organismi biologici seguono delle sostanze chimiche che fungono da traccia per direzionare i loro movimenti). Le seconde invece si riferiscono a forze che si generano nel momento in cui due persone entrano in conflitto per occupare una posizione sulla mappa che risulta già occupata, cosicché la persona che vuole occupare una posizione già occupata esercita una forza sulla persona che occupa tale posizione.

Lo scopo del lavoro consiste nello studiare il comportamento del modello, tenendo traccia del tempo necessario a far sì che tutti gli agenti escano dalla mappa, al variare dei parametri che lo definiscono, introducendo inizialmente le forze sociali ed infine le forze di spinta.

2 Modello

2.1 L'ambiente

Per il nostro modello abbiamo utilizzato la piattaforma GAMA [1]. Tale piattaforma infatti garantisce un ambiente di sviluppo per la modellazione e la simulazione ad agenti particolarmente ottimale per i progetti che coinvolgono la dimensione spaziale, una tipologia di progetto nella quale rientra anche il nostro lavoro. Il modello è visualizzabile e utilizzabile all'indirizzo riportato all'inizio del presente articolo.

Nel nostro modello l'ambiente è rappresentato da una griglia di dimensione 30×80 celle. Sul perimetro della griglia sono presenti sei uscite di lunghezza pari a 5 celle. La forma della griglia è ispirata alla planimetria di piazza San Carlo a Torino, sito in cui si è verificato un grave incidente causato dal movimento di una grande massa di persone [5]. Per il nostro modello abbiamo deciso di lavorare su una griglia di dimensioni minori, mantenendo comunque invariati i rapporti tra le varie lunghezze (Fig.1).

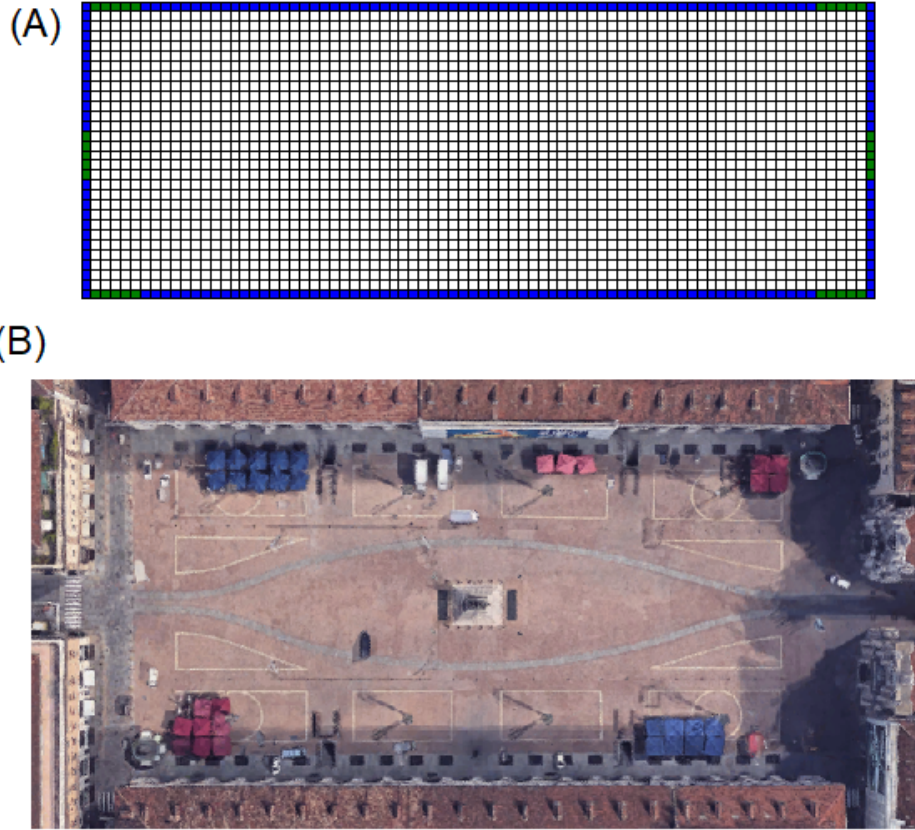


Fig. 1: La griglia del nostro modello (A) e una vista dall'alto della piazza a cui è ispirata (B) .

2.2 Regole di movimento

Una volta definito l'ambiente, lo scopo del nostro modello è quello di simulare il movimento di una grande massa di persone che cercano di raggiungere una delle uscite. Le persone, rappresentate da agenti dinamici che dialogano attraverso l'ambiente, si muovono ad ogni step temporale della simulazione scegliendo una delle celle vicine (ciascuna cella ha quattro celle prime vicine) in base a due parametri: la distanza di ciascuna cella candidata dall'uscita e la traccia di ferormone presente su di essa rilasciata dagli agenti. In particolare, la probabilità con cui una cella viene scelta è pari a:

$$\begin{aligned}
 P_{ij} &= 0 && \text{se la cella } j\text{-esima è un muro,} \\
 P_{ij} &\propto \exp \left[- \left(1 - \frac{S_i}{2S_{max}} \right) k_s S_j + \frac{S_i}{2S_{max}} k_d D_j \right] && \text{se } j \text{ è libera} \\
 P_{ij} &\propto \beta \exp \left[- \left(1 - \frac{S_i}{2S_{max}} \right) k_s S_j + \frac{S_i}{2S_{max}} k_d D_j \right] && \text{altrimenti.}
 \end{aligned} \tag{1}$$

dove l'indice i corrisponde alla cella in cui si trova l'agente e l'indice j alla possibile cella di arrivo. I parametri S e D rappresentano rispettivamente l'attributo *static* e *dinamic* delle celle, mentre k_s e k_d sono due parametri arbitrari che definiscono il peso delle due variabili. Inoltre, rispetto alla forma presentata nell'articolo [6], sono stati aggiunti i due fattori $(1 - S_i/2S_{max})$ e $S_i/2S_{max}$ che vanno a moltiplicare rispettivamente il contributo dato dal valore statico e il contributo dato dal valore dinamico. Tale modifica fa sì che il contributo associato al valore statico abbia un peso maggiore quando si è vicini ad una delle uscite e minore quando si è lontano. Infine, il parametro β , detto *bet*, è un parametro che quantifica la scommessa che ciascun agente effettua sul fatto che una cella occupata venga liberata. Se $\beta = 0$ gli agenti non sceglieranno mai di spostarsi in una cella occupata nel momento della scelta, al contrario, se $\beta = 1$ gli agenti non faranno distinzioni tra celle occupate e celle non occupate. Il parametro β è quindi una stima che gli agenti danno alla probabilità che, se un agente si trova sulla cella di arrivo, esso si sposti prima che l'agente in esame si muova. Una conseguenza del parametro β è quella di controllare la densità della folla: per valori bassi la folla tenderà a essere meno densa, per valori alti la folla risulterà invece più densa. Ovviamente, la probabilità (1) viene opportunamente normalizzata dividendola per la somma delle quattro probabilità associate alle quattro celle prime vicine.

La probabilità (1) è composta essenzialmente da due pezzi: il primo legato all'attributo *static* S_j e pesato dal parametro k_s , mentre il secondo è legato all'attributo *dinamic* D_j e pesato da k_d . Il primo termine favorisce le celle che distano meno dalle uscite. S_j rappresenta infatti proprio la distanza euclidea della cella di arrivo dall'uscita più vicina. Dunque, più tale attributo sarà grande più la probabilità di scegliere quella cella sarà piccola (e viceversa). Il secondo termine della (1) serve invece a favorire le celle in cui sono passati molti agenti. L'attributo D_j è infatti legato al seguente processo diffusivo: quando un agente si sposta lascia nella cella di partenza una quantità di ferormone pari a uno 1, il quale diffonde nelle celle vicine ad ogni step temporale successivo. Viene definito inoltre un tempo di evaporazione durante il quale la traccia svanisce. D_j è la somma di tutti i processi diffusivi di questo tipo che finiscono sulla cella j -esima. Pertanto, se una cella ha un attributo D_j grande significa che da quella cella o nelle celle vicine sono passati o stanno passando molti agenti. Riassumendo, il primo termine rappresenta la volontà delle persone di muoversi verso l'uscita più vicina, mentre il secondo rappresenta la tendenza delle stesse a seguire la folla. A seconda del valore dei parametri k_s e k_d si può passare da una situazione in cui viene enfatizzato uno o l'altro comportamento.

2.3 Forze

Il modello prevede anche l'eventuale presenza di un campo di forze. Quando un agente sceglie di spostarsi in una cella che, nel momento in cui si muove, risulta essere occupata da un altro agente, esso rimane nella cella in cui si trova ed esercita un'unità di forza nei confronti dell'agente che occupa la cella in cui voleva spostarsi. Ciascun agente risente dunque di un campo di forza rappresentato da un vettore di quattro elementi, ciascuno dei quali associato alla spinta ricevuta in una delle quattro direzioni: nord, est, sud, ovest.

Il calcolo della forza che ciascun agente risente è così fatto: partendo dalla cella in cui l'agente si trova, si scorrono le celle nella quattro direzioni nord, est, sud e ovest. Prendendo ad esempio la direzione est (per le altre direzioni vale un discorso analogo), si valuta se nella prima cella in tale direzione c'è un agente che vorrebbe spostarsi verso ovest, ovvero nella direzione opposta, e in tal caso si aggiunge un valore uno all'elemento corrispondente del vettore forza. Quando si trova una cella vuota o occupata da un agente che non vuole spostarsi nella direzione opposta ci si ferma. Un esempio del calcolo è riportato in figura (2). La forza che spinge un agente nella direzione est compete con la forza che lo spinge nella direzione ovest e analogamente per la coppia nord-sud. Se una delle due differenze tra i valori delle due coppie supera una certa soglia F_{max} l'agente non si sposterà più nella cella che ha scelto ma verrà 'trascinato' dalla forza.

2.4 Agenti

Nel nostro modello, come detto, ci sono due tipi di agenti: gli agenti cella e gli agenti persona. Presentiamo qui brevemente i due tipi di agenti e i loro principali attributi. A ciascuna cella della griglia è associato un agente statico i cui principali attributi sono:

- **grid.x**: coordinata x della cella.
- **grid.y**: coordinata y della cella.
- **neighbors**: lista dei primi vicini.
- **is.wall**: variabile booleana che è vera se la cella rappresenta un muro o un ostacolo e in cui gli agenti persona non possono andare.
- **is.exit**: variabile booleana che è vera se la cella rappresenta un'uscita.
- **is.free**: variabile booleana che è vera se nella cella è presente un agente persona o falsa altrimenti.
- **static**: distanza euclidea dal centro dell'uscita più vicina.
- **dinamic**: quantità di ferormone presente nella cella.

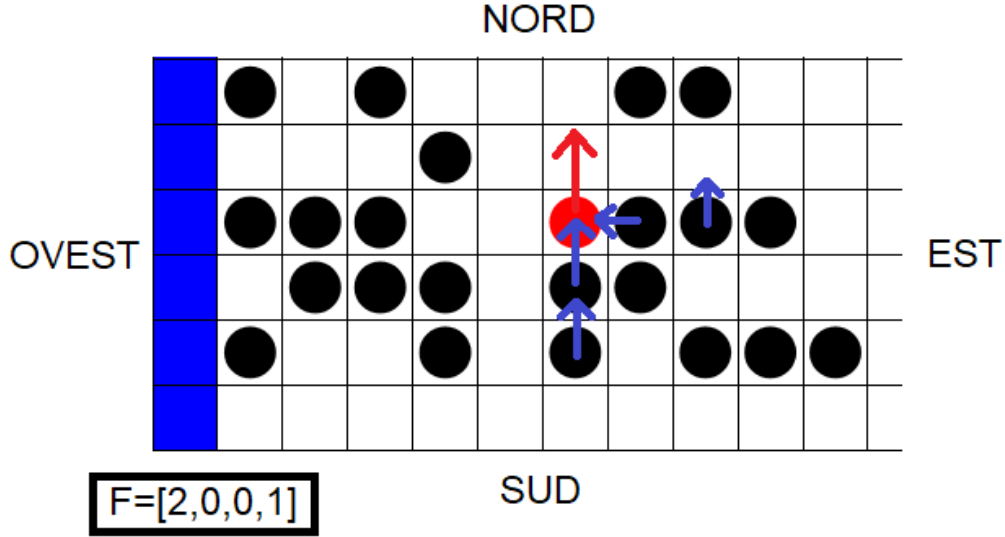


Fig. 2: Esempio di campo di forza percepito da un agente (rosso). Se si suppone $F_{max} = 1$, l'agente si sposterà verso nord.

Gli agenti cella non hanno alcun obbiettivo ma hanno la funzione di creare un ambiente in grado di fornire informazioni (distanza dalle uscite) e mediare gli scambi di informazioni (ferormone) tra gli agenti persona.

Gli agenti persona invece sono agenti liberi di muoversi all'interno della griglia. Essi possiedono i seguenti attributi:

- **current_cell**: cella in cui si trova.
- **possible_cell**: cella scelta per il prossimo movimento.
- **forza**: vettore di 4 elementi che rappresentano ciascuno la forza di cui l'agente risente rispettivamente verso nord, est, sud e ovest.

Gli agenti persona sono agenti reattivi le cui decisioni dipendono soltanto dallo stato dell'ambiente che li circonda. L'unica forma di intelligenza che possono presentare è un'intelligenza emergente e di tipo collettivo, in quanto i singoli agenti persona non sono dotati di una propria intelligenza.

2.5 Organizzazione della simulazione

Durante la simulazione gli agenti eseguono ciclicamente e in un ben preciso ordine tre distinte azioni, ciascuna delle quali viene eseguita da tutti gli agenti prima di passare alla successiva. Questa gerarchia nelle azioni viene gestita dalle tre variabili booleane *think*, *force* e *run*, tali che in ogni momento della simulazione una sola delle tre è vera. La prima azione eseguita dagli agenti è scegliere la cella su cui spostarsi. Ovvero, ciascun agente aggiorna il proprio attributo *possible_cell* secondo le regole presentate nella sezione (2.2). La seconda azione eseguita dagli agenti è calcolare la forza a cui sono sottoposti. Ciascun agente aggiorna dunque il vettore *forza* e, nel caso in cui venga superata la forza soglia, modifica il proprio attributo *possible_cell* con la cella in cui viene spinto dalle forze. La terza ed ultima azione corrisponde al movimento dell'agente nella cella associata all'attributo *possible_cell*. Operativamente, durante la fase di movimento l'agente aggiorna l'attributo *current_cell* assegnandogli il valore posto in *possible_cell*. Prima di spostarsi l'agente libera la cella in cui si trova, ovvero l'agente cella imposta l'attributo booleano *is_free* come *true*. Se un agente si trova su una delle uscite, libera la cella e muore. Per quanto riguarda le prime due azioni, non è importante l'ordine con cui gli agenti vengono scelti per eseguirle. Al contrario, l'ordine con cui si aggiornano gli agenti risulta essere fondamentale nell'ultima azione. GAMA aggiorna in automatico gli agenti in modo random e nel nostro modello si è scelto di mantenere tale regola in quanto ci è sembrata la più appropriata.

3 Risultati

Si sono effettuate diverse simulazioni per analizzare i parametri del modello, in particolar modo si è analizzato il tempo di uscita impiegato dagli agenti a uscire dalla mappa.

3.1 Tempo Di Evaporazione

Il primo parametro analizzato è stato il tempo di evaporazione dei crumbs rilasciato dall'agente. Si sono quindi effettuate varie simulazioni al variare dell'evaporazione e di k_d . Il numero di agenti è stato fissato a 1000 e si sono effettuate 12 simulazioni per ogni set di $[k_d, \text{evaporazione}]$, infine si è presa la media e la varianza di questi.

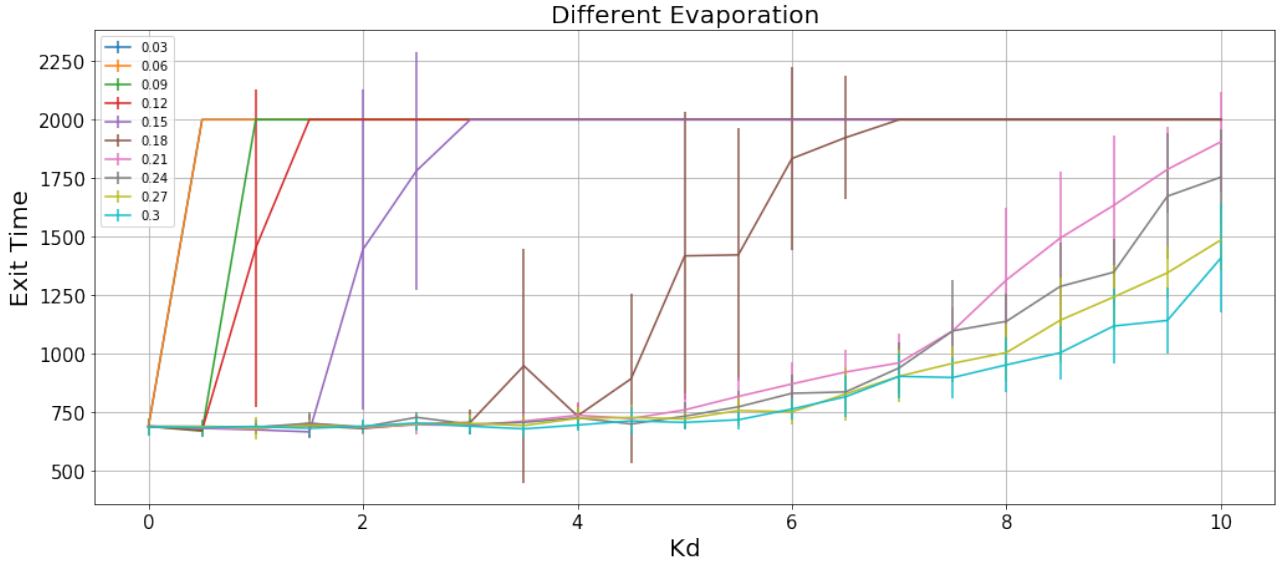


Fig. 3: Tempo medio di uscita al variare di k_d con $k_s = 1$ per differenti valori del parametro che regola l'evaporazione del campo dinamico, rappresentati in figura con i diversi colori.

Come si vede dalla figura per ogni valore di evaporazione esiste un valore critico di k_d oltre il quale si hanno delle persone intrappolate in artefatti della simulazione. Esempio: se l'evaporazione è troppo bassa rispetto a k_d il campo dinamico di due agenti vicini finisce per auto-alimentarsi intrappolando i due agenti in un minimo da loro stessi creato.

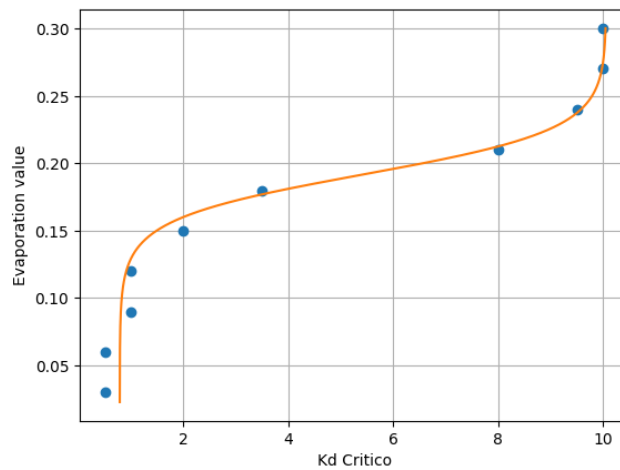


Fig. 4: In figura sono riportati, al variare di k_d , i valori del parametro che regola l'evaporazione tali per cui non tutti gli agenti riescono a raggiungere le uscite in 2000 cicli. Si è considerato k_s costante e pari a 1.

Per le simulazioni successive si è scelto di usare 0.21 come valore per l'evaporazione in modo tale da avere completa evacuazione della mappa fino a valori alti di k_d .

3.2 Analisi dei Parametri Ottimali

Una volta fissata la costante di evaporazione, abbiamo raccolto i tempi di uscita di ciascuna simulazione andando a variare i parametri che regolano l'accoppiamento con il campo statico e con il campo dinamico nell'intervallo $[0.5, 9.5]$ a passi di 1 per k_s e nell'intervallo $[0, 10]$ a passi di 0.5 per k_d , e β nell'intervallo $[0, 1]$ a passi di 0.25. Nelle figure sono presenti i tempi di uscita mediati su 12 simulazioni per ogni configurazione di parametri al variare di k_s , tenendo k_d fissato su alcuni valori rappresentativi.

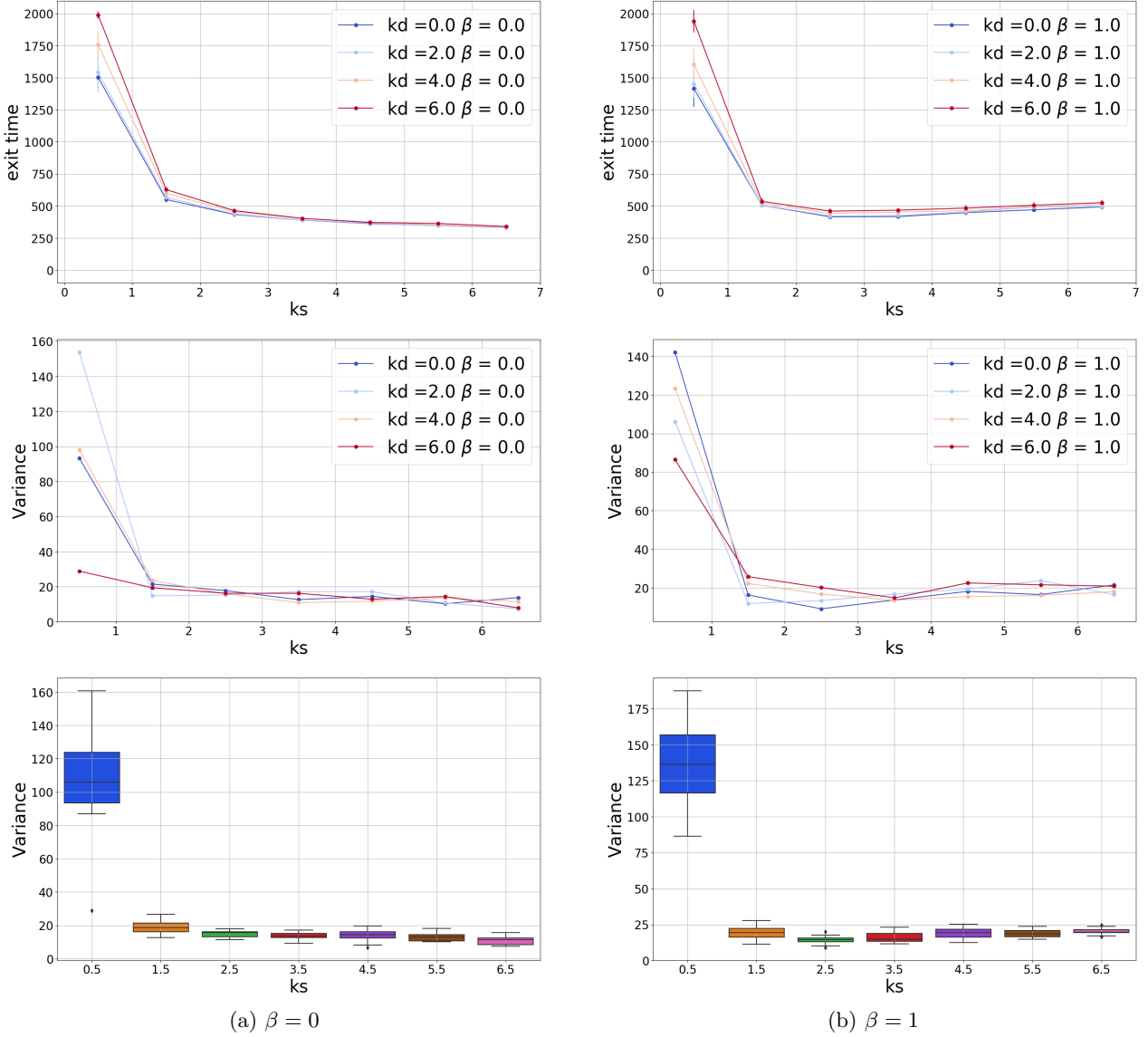


Fig. 5: Media e varianza del tempo di uscita in funzione di k_s per diversi valori di k_d (figure in alto ed in mezzo) e boxplot relativo alla distribuzione delle varianze al variare di k_s . A sinistra per $\beta = 0$ mentre a destra per $\beta = 1$

In Fig. 5 in alto vediamo il tempo medio di uscita in funzione di k_s . È possibile vedere che per valori bassi di k_s i tempi di uscita raggiungono valori prossimi al limite massimo che abbiamo impostato a 2000 cicli, questo è dovuto al fatto che in tale configurazione gli agenti danno poca importanza alla posizione dell'uscita a loro più vicina e si muovono casualmente, tanto più k_d è alto. Aumentando k_s si osserva che i tempi di uscita si riducono drasticamente fintanto che l'accoppiamento con il campo dinamico rimane contenuto, invece quando questo domina si assiste ad un incremento del tempo di uscita e della relativa varianza. Infine, una volta che l'accoppiamento con il campo statico supera $k_s = 2$, il tempo medio di uscita si stabilizza su un valore costante indipendentemente dall'intensità dell'accoppiamento con il campo dinamico, dovuto al fatto che in tale regime gli agenti hanno piena conoscenza della posizione dell'uscita e la decisione sul loro moto è determinata unicamente da questa informazione.

Questo comportamento globale risulta praticamente insensibile ai diversi valori del parametro β e quindi possiamo dire che la densità della folla non ha un ruolo determinante a questo livello. Nel momento in cui β assume valore diverso da zero, si osserva invece un minimo nel tempo medio di uscita nei pressi di $k_s = 2.5$, indipendentemente dal valore assunto da k_d .

Si può osservare in Fig. 6 che i diversi valori di β hanno un effetto opposto ai due estremi dei valori sui quali facciamo variare l'accoppiamento con il campo statico, infatti in figura si vede come per valori bassi di k_s , una minor densità della folla comporti un aumento del tempo medio di uscita mentre per valori alti di k_s sia la maggior densità ad aumentare il tempo medio di uscita. Questo comportamento può essere attribuito al fatto che una maggior uniformità tra gli agenti nel muoversi verso la direzione relativa all'uscita più vicina penalizza il tempo di uscita nel caso di maggior densità perché la folla rimane congestionata a causa della volontà di più agenti di seguire lo stesso percorso. Di contro, se l'accoppiamento con il campo statico è debole si ha un beneficio dalla maggior densità perché, non verificandosi congestionamento, le persone si avvicinano all'uscita più velocemente.

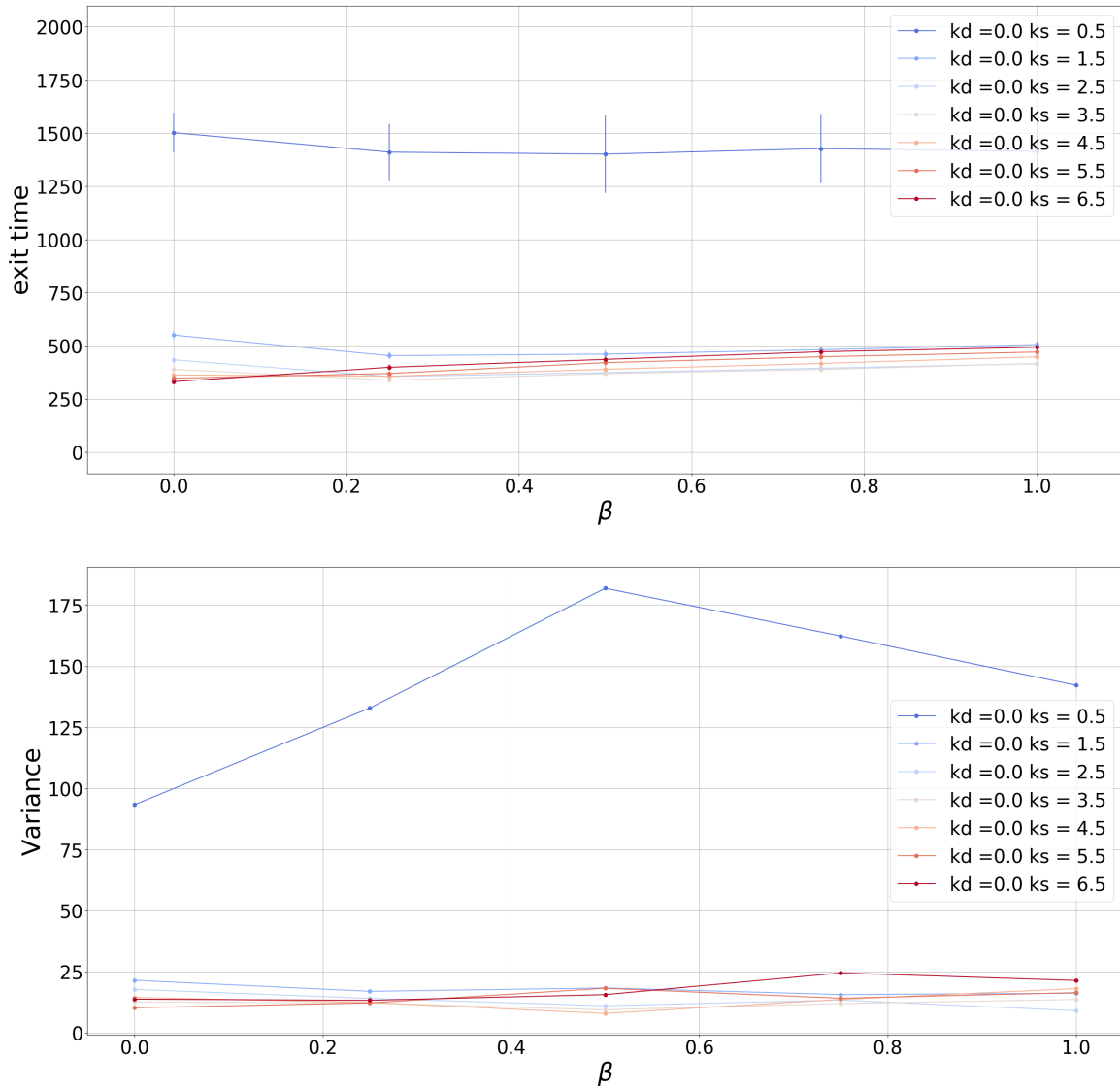


Fig. 6: diminuire β favorisce l'uscita degli agenti per k_s elevati

Complessivamente, a questo livello la dinamica è governata da k_s il quale ha un ruolo fondamentale nel passaggio da un regime di moto casuale, nel momento in cui k_s è basso, ad un regime di moto ordinato, quando aumenta e ne consegue una riduzione drastica della varianza del tempo di uscita. Invece k_d ha il ruolo di rendere il moto degli agenti casuale quando è alto rispetto k_s . Tra questi due regimi, è possibile osservarne un altro in cui si osserva la minimizzazione del tempo medio di uscita intorno a $k_s = 2.5$ e per valori di β superiori a zero. Nonostante che la dinamica sia determinata principalmente da k_s , esiste una configurazione di parametri che ottimizza il tempo medio di uscita.

3.3 Forze

Nelle figure (7) e (8) è riportato l'andamento del tempo medio di uscita e la rispettiva varianza, confrontato il caso in cui non siano presenti forze (sinistra) al caso in cui invece siano presenti (destra). In particolare, in Fig.7 si varia sia k_s che k_d mantenendo il parametro $\beta = 0$. Nel caso in cui siano presenti le forze si osserva un effetto simile a quando le forze sono disattivate. In particolare, k_s appare sempre essere il parametro d'ordine principale del sistema. Infatti per k_s che si discosta da valori prossimi allo zero il tempo di uscita diminuisce drasticamente e anche la varianza mostra lo stesso comportamento: sintomo che il sistema ha una sorta di transizione di fase. Nel caso in cui siano presenti le forze, si osserva un fenomeno nuovo che consiste in un brusco incremento del tempo di uscita per valori grandi di k_s . Infatti, per quei valori di k_s gli agenti tengono conto esclusivamente della posizione delle uscite e finiscono per accalcarsi nei pressi delle stesse. La presenza delle forze fa sì che si creino degli ingorghi dai quali gli agenti escano con difficoltà.

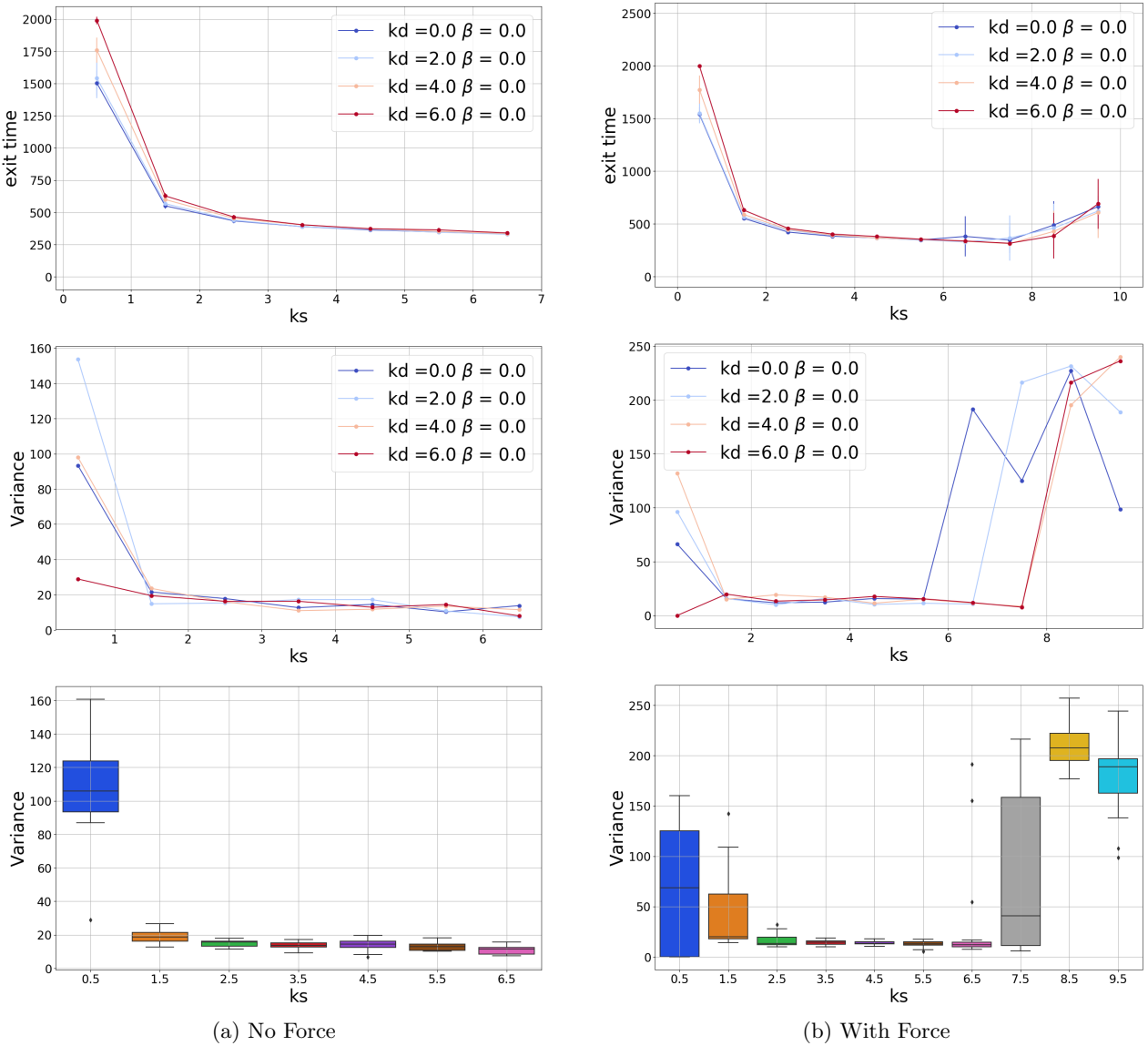
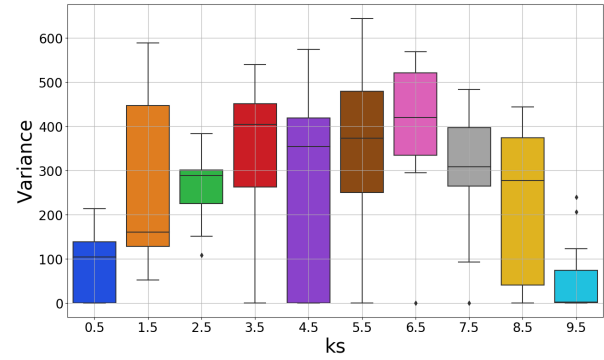
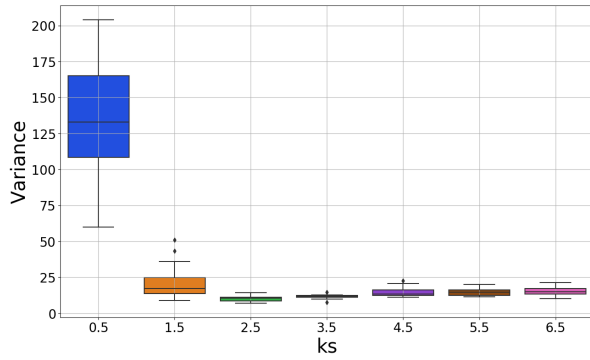
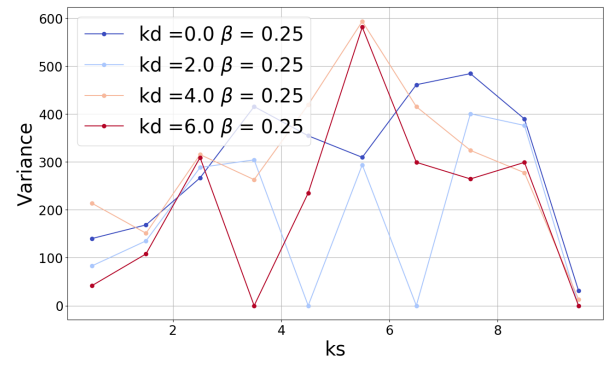
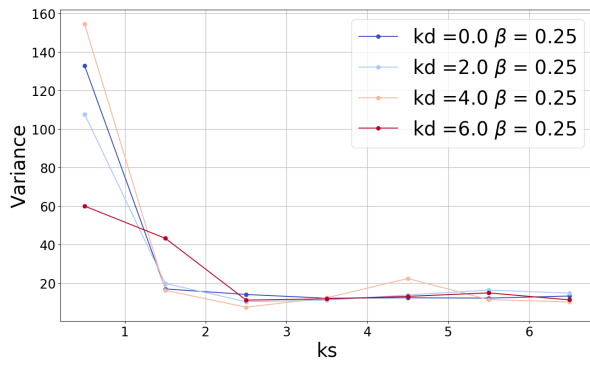
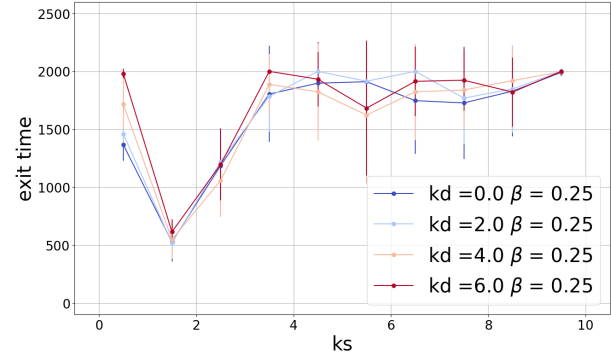
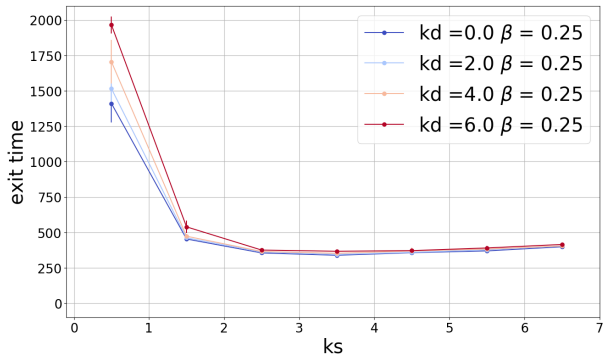


Fig. 7: L'introduzione delle forze non comporta sostanziali cambiamenti, fatta eccezione per i valori più estremi, associati alla formazione di ingorghi.

Risultati diversi si osservano in Fig.8, dove il parametro β è fissato ad un valore diverso da zero (0.25). In generale, i tempi di uscita risultano essere molto più grandi in presenza delle forze. Ricordiamo che il parametro β quantifica la densità della folla, la quale è strettamente legata all'effetto delle forze. In una folla più densa gli agenti hanno più probabilità di risentire della forza e il loro movimento risulterà essere governato principalmente da quella. Dai grafici si osserva come il parametro k_s non riesca più ad avere lo stesso effetto che ha nel caso di assenza di forze. Il tempo di uscita rimane sostanzialmente alto per ogni valore di k_s , fatta eccezione per



(a) No Force

(b) With Force

Fig. 8: Per valori diversi da zero di β l'effetto delle forze diventa molto importante.

il caso $k_s = 1.5$ al quale però è associato una varianza molto alta. Inoltre anche la varianza non mostra più l'andamento che mostrava nel caso senza forze e rimane alta per i diversi valori di k_s (è necessario notare che a quei valori in cui la varianza è pari a 0 sono associati tempi di uscita costantemente superiori al limite dei 2000 cicli e che vengono considerati tutti uguali a 2000).

Dunque, sembra che il parametro β sia quello fondamentale per avere un effetto o meno dalla comparsa della forza nel sistema. Per valutare meglio l'effetto di β , si è riportato in Fig.(9) il tempo di uscita e la varianza al variare proprio del parametro β , nei due casi di assenza e presenza di forze. Mentre nel primo caso al variare di β non si osserva nessun effetto considerevole, nel caso in cui ci siano le forze il tempo di uscita aumenta bruscamente se β assume un valore superiore a zero. In conclusione, possiamo dire che le forze hanno un effetto consistente solo se β è diverso da zero.

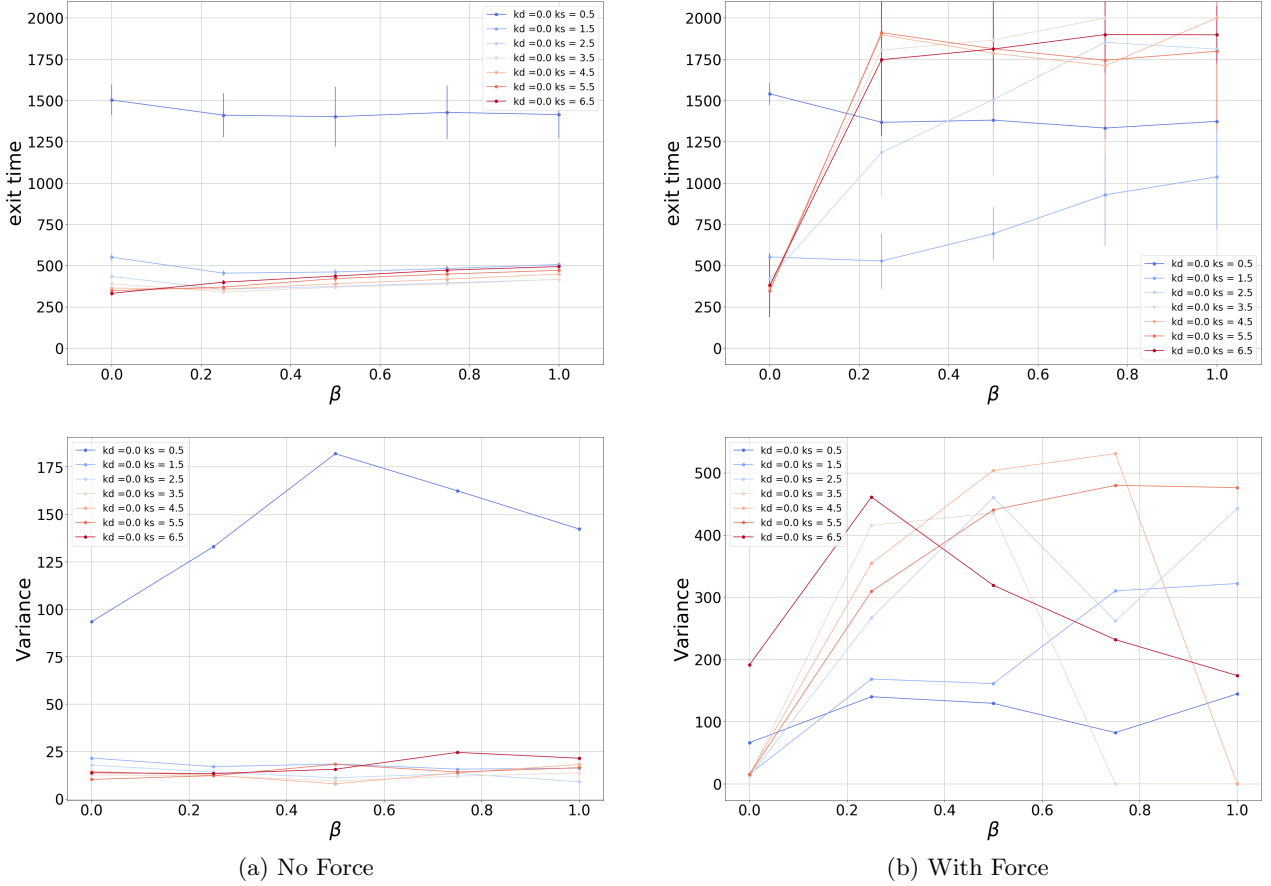


Fig. 9: β è il parametro che accende o meno l'effetto delle forze.

4 Conclusioni

I risultati trovati sembrano mostrare una dipendenza forte del sistema dal parametro k_s . In particolare, l'andamento della varianza del tempo di uscita al variare di k_s sembra indicare la presenza di due differenti stati del sistema (Fig. 5).

A questo primo livello di modellizzazione, in cui non si considera l'effetto delle forze, la densità della folla sembra giocare un ruolo solo marginale. L'unico effetto apprezzabile nel caso di folla molto densa è la creazione di un minimo per il tempo di uscita in corrispondenza di un valore di k_s interno all'intervallo di riferimento, mentre nel caso di folla poco densa il minimo risulta essere nell'estremo destro dell'intervallo.

Al contrario, il parametro k_d non gioca un ruolo fondamentale nel modello, infatti contribuisce ad aumentare il tempo di uscita per k_s piccoli e risulta totalmente ininfluente per k_s grandi. Tale risultato non è in accordo con [6] in cui un aumento di k_d provoca un aumento non monotono del tempo di uscita per valori di k_s bassi, in cui si osserva un minimo locale. In questo regime, gli agenti beneficiano anche della conoscenza di regioni della mappa in cui è presente un alto flusso di agenti in moto.

Questo disaccordo potrebbe essere dovuto alla differente geometria della mappa, infatti in [6] è presente un'unica uscita in una mappa di dimensione 63 x 63, con distanza media dall'uscita pari a 37, mentre noi

abbiamo 6 uscite in una mappa di dimensione 80×30 , con distanza media da un'uscita pari a 17. In queste condizioni, gli agenti potrebbero non aver modo di percepire il campo dinamico.

L'introduzione delle forze nel modello comporta alcuni cambiamenti nella dinamica del sistema. Si ha in particolare un effetto di penalizzazione sui valori grandi di k_s per quanto riguarda il tempo di uscita. L'effetto delle forze è amplificato o ridotto a seconda che la folla sia rispettivamente molto densa o poco densa. Si è quindi riscontrato, come ci aspettavamo, una forte dipendenza dell'effetto delle forze dalla densità della folla (e quindi dal parametro β).

Durante il lavoro sono emersi alcuni spunti per ulteriori sviluppi, tra cui la possibilità di usare differenti mappe con uscite in posizioni ed in numero differente e, al limite, considerare una mappa che corrisponda a Piazza San Carlo in scala 1:1. In queste situazioni, sarebbe possibile verificare se il modello è sensibile alla grandezza della mappa per quanto riguarda l'effetto del campo dinamico sugli agenti. Inoltre, potrebbero essere utilizzati valori negativi dell'accoppiamento con il campo dinamico che rappresenterebbe un'interazione repulsiva rispetto a celle in cui sono passati altri agenti.

References

1. Taillandier P., Vo DA., Amouroux E., Drogoul A.: GAMA: A Simulation Platform That Integrates Geographical Information Data, Agent-Based Modeling and Multi-scale Control. Springer, Berlin, Heidelberg (2012)
2. V. J. Kok, M. K. Lim, C. S. Chan : Crowd behavior analysis: A review where physics meets biology (2016)
3. I. Sakour, H. Hu : Robot-Assisted Crowd Evacuation under Emergency Situations: A Survey (2017)
4. X. Pan1, C. S. Han, K. Dauber, K. H. Law : A Multi-agent Based Framework for the Simulation of Human and Social Behaviors during Emergency Evacuations (2007)
5. Wikipedia, https://it.wikipedia.org/wiki/Tragedia_di_piazza_San_Carlo
6. A. Kirchner, A. Schadschneider : Simulation of evacuation processes using a bionics-inspired cellular automaton model for pedestrian dynamics (2002)
7. Colin M. Henein, Tony White: Agent-Based Modelling of Forces in Crowds (2005)