

ALMA MATER STUDIORUM
UNIVERSITÀ DEGLI STUDI DI BOLOGNA

Seconda Facoltà di Ingegneria
con Sede a Cesena
Corso di Laurea Specialistica in Ingegneria Informatica

CROWD MODELING

Elaborata nel corso di: Sistemi Multi Agente LS

Relazione di Progetto di:
DENIS BRIGHI

Relatore:
Prof. ANDREA OMICINI

ANNO ACCADEMICO 2010–2011
SESSIONE III

PAROLE CHIAVE

Crowd Modeling

ABM - Agent Based Modeling

Particle Systems

Autonomous Character

Leader Follower Model

Indice

Struttura della relazione	vii
1 Il Crowd Modeling Problem	1
2 Modellare E Simulare	5
2.1 Caratteristiche dei modelli	5
2.2 Crowd Modeling and Simulation Problem	6
2.3 Aspetti di Scienze Sociali	8
2.4 La folla: i comportamenti, i movimenti e le collisioni	9
3 Tecniche di Modellazione delle Masse	13
3.1 I Sistemi Fluidodinamici	13
3.2 Flocking Systems	14
3.3 I Sistemi a Particelle (Particle Systems)	16
3.4 Autonomous Animated Character	17
3.4.1 Perception Modeling	17
3.4.2 Intention Generator	18
3.4.3 Behavior Planning	18
3.4.4 Comparazione con Agenti BDI	18
3.5 Sistemi a gruppi e a Leader	19
3.5.1 Dynamic Grouping Algorithm	20
4 Opere in Letteratura	21
5 Caso di studio: Crowd Steering Model	27
5.1 Il caso di studio	27
5.2 Il modello Alchemist	28
5.3 Il modello RePAST	30

Struttura della relazione

Questa relazione sarà così strutturata: nel primo capitolo verrà introdotto il problema del Crowd Modeling, puntualizzando le caratteristiche, i vantaggi e le applicazioni; in seguito accenneremo il problema della modellazione e simulazione applicato al problema; nel terzo capitolo verranno esposte le tecniche più utilizzate nell'ambito della modellazione di folle e a seguire verranno elencati i lavori di ricerca maggiormente significativi. L'ultimo capitolo è parte del progetto SAPERE, in cui un simulatore ad hoc (Alchemist) è stato confrontato con un simulatore opensource ad agenti di riferimento (Repast) con riferimento ad un modello comune di Crowd Steering.

Capitolo 1

Il Crowd Modeling Problem

Controllare una folla di persone in aeroporti, stazioni di treni, in occasione di eventi sportivi, etc., è un problema complesso. Infatti bisogna gestire un'enorme quantità di informazioni e di interazioni scambiate fra gli agenti e tra questi e gli ambienti circostanti. Il fattore di sca-

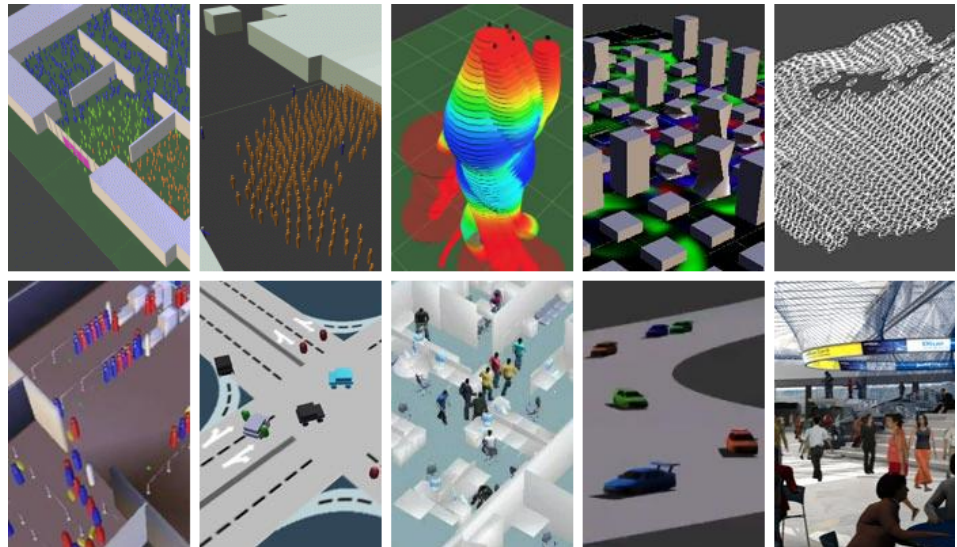


Figura 1.1: Snapshots di Crowd Simulations

labilità è intrinseco al problema, in quanto di per sé la folla, sia essa di esseri umani, animali, oggetti virtuali animati o robot, è compo-

sta da un numero elevato di individui interagenti, rendendo il sistema complesso e caotico. Inoltre l'osservazione del sistema è migliore se è possibile seguirne il suo comportamento attraverso una visualizzazione grafica.



Figura 1.2: scena tratta da da: Lord of the Rings

Questa inoltre è imprescindibile nel caso in cui la simulazione della folla sia applicata a contesti legati a film, cortometraggi o videogame: è proprio qui che il fattore visivo è particolarmente accentuato per immergere lo spettatore dentro al momento che si vuole riprodurre. Questa tecnica è utilizzata molto spesso in film, quali 'Il Signore degli Anelli', per sollevare il lavoro di scripting delle azioni dei personaggi. Questo infatti emergerà dal comportamento simulato dell'individuo associato.

Possiamo dunque suddividere i problemi di crowd simulation in categorie:

- *Simulazioni Real-Time*, simulazioni nelle quali è fondamentale che il tempo di risposta della simulazione sia il più breve possibile o in tempo reale, per l'osservare l'evoluzione del modello e prendere decisioni;
- *Simulazioni Grafiche*, si contrappongono alle real-time, pongono come punto fondamentale l'aspetto visivo del modello;



Figura 1.3: scena tratta da: *Flags of our Father*

- *Simulazioni Ibride*, ove si compongono i due approcci già descritti, simulando il modello in tempo reale, ma con sfondo una rappresentazione grafica dettagliata.

Per ognuna di queste categorie possiamo individuare uno specifico ambito applicativo:

- Le *simulazioni in real-time* vengono applicate ad ambiti di video sorveglianza, addestramento ed in ogni occasione in cui si preferisce la precisione delle informazioni riguardanti i movimenti e le traiettorie in tempo reale alla visualizzazione grafica. L'aspetto visivo della simulazione è spesso piuttosto spartano, ove gli individui sono stilizzati e identici tra loro. Questa tipologia infatti, astrae il livello visivo in modo sostanziale, tanto da migliorare le performance e ridurre i tempi di attesa della simulazione. Esempi tipici sono: controllo del traffico stradale, flussi di persone in aeroporti, stazione di treni, comportamenti sociali in eventi sportivi, manifestazioni, proteste, meeting, i quali possono contribuire sostanzialmente allo sviluppo di nuove tecniche di addestramento per polizia ed esercito.
- Le *simulazioni con rendering grafico* vengono utilizzate soprattutto in ambito cinematografico, dove l'attenzione ai dettagli, ai movimenti di ciascun individuo è altissima e il tempo di calcolo

impiegato è influente e per il quale può essere speso un maggior quantitativo per ottenere migliori risultati. Il lavoro più importante viene realizzato in fase di disegno, quando ogni individuo avrà una propria particolarità e sarà differente da ciascun altro, ponendo grande attenzione al rendering dei movimenti.

- *Un approccio ibrido* viene utilizzato prevalentemente in ambito videoludico. Esso infatti combina le performance di una simulazione real-time, con un livello di dettagli superiore, cercando un trade-off che renda il videogioco complessivamente fruibile e sufficientemente ricco di dettagli grafici.



Figura 1.4: scena tratta da: Happy Feet

Capitolo 2

Modellare E Simulare

Il compito della simulazione a computer è risolvere uno dei problemi chiave nell'industria moderna, sempre alla ricerca di diminuzione dei costi, dal design, sviluppo, test, fino all'addestramento del personale. La modellazione infatti prevede di rappresentare il mondo reale in forma virtuale, permettendo di sviluppare tecniche che poi verranno applicate di nuovo al sistema esistente come frutto dei risultati provenienti dalla simulazione.

2.1 Caratteristiche dei modelli

I modelli possono essere prevalentemente di tre tipi:

- *Reduction*: il modello del sistema reale è costruito riducendo il numero delle variabili o parametri, per ottenere una simulazione più veloce e semplice da controllare;
- *Abstraction*: l'astrazione del mondo verso un modello espresso in un modo diverso, ad esempio tramite l'uso di equazioni differenziali, etc.;
- *Simplification*: si semplifica la situazione reale, eliminando i dettagli di meno rilievo, esaltando quelli centrali nella dinamica del sistema.

Un “buon modello” lo si ha quando verifica le seguenti proprietà:

1. la riduzione conserva le proprietà essenziali del sistema
2. la semplificazione non altera la precisione dei risultati, i quali dovrebbero essere controllati sperimentalmente
3. i parametri di simulazioni sono semplici da comprendere e da controllare
4. il modello è sufficientemente robusto da garantire previsioni affidabili
5. il modello è capace di adattarsi alle modifiche del sistema

Esistono diversi approcci alla modellazione e si possono dividere essenzialmente in due categorie: i modelli fisici e quelli astratti. I modelli fisici hanno un grado di astrazione molto basso e tendono a riportare molti particolari del sistema dato; molto spesso sono spiegati con lo stesso linguaggio in cui sono costituiti. Non c'è però una netta separazione tra i modelli fisici e quelli astratti in quanto un stesso modello può essere fisico per una certa applicazione, mentre un'astrazione per un'altra. Caso tipico sono i modelli fluidodinamici: se applicati al campo in cui sono interessati fluidi (ad esempio liquidi, o calore...) questi sono considerati come un modello fisico; se applicati alla movimentazione di merci o traffico, allora rappresentano un modello astratto.

2.2 Crowd Modeling and Simulation Problem

Il Crowd Modeling è il processo di modellazione e simulazione del movimento di un vasto numero di oggetti o persone in un dato contesto. In fase di modellazione, gli individui vengono plasmati in funzione del tipo di comportamento che si vuole che essi adottino. Nella fase di simulazione, avviene la verifica di tale dinamica, ma soprattutto si osserva il comportamento emergente del sistema, quello risultante dall'interazione fra individui e che non può essere dedotto dalla semplice analisi dei singoli attori. Il bisogno di simulare popolazioni iniziò quando gli scenari in considerazione richiesero un'attenzione ad un

numero maggiore di caratteristiche che non potevano essere osservate attraverso l'uso dei sistemi convenzionali. L'utilizzo della simulazione virtuale offre il vantaggio di offrire una visione d'insieme del sistema, facendo previsioni sulle possibili dinamiche che possono accadere, essere più conveniente dal punto di vista economico e permette il totale controllo di ogni individuo coinvolto.

Il Crowd Problem inoltre, può fare riferimento a simulazioni basate su dinamiche di gruppo e psicologia delle masse, soprattutto nella pianificazione di pubblica sicurezza. In questo caso ci si focalizza soltanto sui comportamenti della folla. Quest'ultima è diventata di interesse scientifico all'inizio del XIX secolo, focalizzando la ricerca sui comportamenti sociali collettivi di persone ad eventi quali assemblee, incontri sociali, proteste, ribellioni, concerti, eventi sportivi e cerimonie religiose. Attraverso lo studio dei comportamenti umani in diversi tipi di situazioni portate all'estremo, si possono creare modelli avanzati (sviluppando strategie di sorveglianza e controllo della folla) utilizzati in seguito anche da polizia, militari e volontari. Le tecniche di modellazione della folla possono variare da approcci olistici, a quelli a rete; ognuno punta a comprendere aspetti individualistici o comportamentali di ciascun agente. L'uso di modelli multi agente semplifica lo studio dei complessi comportamenti ed interazioni che si sviluppano tra le parti, rendendo le dinamiche emergenti molto più comprensibili. Grazie all'uso di questi software inoltre, i sistemi possono essere testati in condizioni estreme, diventando perfettamente replicabili, e lunghe simulazioni diventano problemi il cui tempo di risoluzione si riduce notevolmente.

In letteratura il problema del Crowd Modeling è un campo ben noto e particolarmente interessante, per il quale sono state proposte diverse soluzioni, ognuna delle quali punta ad una maggiore precisione dei movimenti del gruppo e al generale miglioramento delle prestazioni, in quanto il processo di simulazione è computazionalmente esigente. Inoltre un elevato numero di individui comporta numerose parti interagenti, tra loro e col sistema in cui sono immersi, appesantendo le performance. Ogni decisione, ogni movimento, ogni interazione comporta complesse elaborazioni che difficilmente limitate risorse computazionali potrebbero gestire. Vediamo i primi lavori su questo tema a partire dai primi anni '80: tra i pionieri possiamo ricordare Amkraut,

Girard and Karl[2] ma soprattutto Reynolds[14] che hanno steso la base per successivi studi e lavori in questo campo. Non a caso gli ambienti videoludico e cinematografico hanno dato un notevole incentivo a questi campi di ricerca. Come vedremo, questi saranno di notevole importanza per la creazione di scene sia di film che di videogiochi.

2.3 Aspetti di Scienze Sociali

Una definizione largamente accettata di folla è quella di un folto gruppo di individui nello stesso ambiente fisico, che condividono un obiettivo comune. Gli individui in una folla possono agire in modi diversi a differenza di quelli che si trovano da soli o in piccoli gruppi. Grazie all'utilizzo di questa tecnica si possono osservare e verificare i comportamenti teorizzati dagli studi di sociologia delle masse, senza coinvolgere costosi esperimenti. Inoltre dato l'elevato grado di libertà nel formulare il modello, è possibile replicare identiche situazioni variando solamente alcuni parametri, consentendo agli utenti di comprendere meglio quali siano i comportamenti emergenti nelle varie sperimentazioni.

Ci sono tuttavia, altri tipi di effetti di gruppo rilevanti che sono degni di menzione. La *Polarizzazione* avviene all'interno della massa quando due o più sottogruppi adottano attitudini, opinioni o comportamenti differenti, fino al punto che possono avvenire discussioni che potranno diventare lotte. In certi casi i gruppi possono vedersi come veri e propri avversari. L'effetto *Condivisione* è il risultato dell'influenza da parte di altrui azioni a livello individuale. Questo è possibile perché l'ambiente in cui è immerso il sistema è comune ed osservabile da parte degli individui: le azioni di uno si possono ripercuotere su un'altro. L'*Addizione* è il nome dato allo stesso effetto quando applicato però al gruppo. La *Predominazione* avviene quando uno o più *Leader* nella massa influenzano il comportamento dell'intero gruppo.

L'obiettivo della simulazione è rappresentare questi fenomeni in maniera virtuale, rendendo così possibile accostare il modello umano-reale al modello umano-virtuale, da cui estrapolare dati per una successiva elaborazione.

2.4 La folla: i comportamenti, i movimenti e le collisioni

Sebbene la modellazione possa prevedere diversi gradi di astrazione, quando si parla di folla è quasi impossibile astrarre il livello grafico. Per quanto possiamo semplificare la visualizzazione, gli individui occupano un loro spazio fisico nell'ambiente, si muovono e sono a contatto con altri individui pertanto a questo punto sorgono diversi problemi quali il movimento degli individui e il rilevamento o la prevenzione delle collisioni. Esiste un unico caso in cui questo può essere evitato, cioè il modello fluidodinamico (questa eccezione verrà poi riformulata più avanti perché possiede un meta-modello di fondo completamente differente).

Possiamo definire generalmente un individuo a livello fisico come un oggetto che possiede le seguenti caratteristiche: una posizione, una massa, una velocità, e un'orientamento. A più alto livello avrà un goal da realizzare e tutto un insieme di logiche, di tecniche per realizzarlo.

Detto ciò definiamo quali siano i tipi di comportamenti che un individuo può compiere:

- *Steering*: l'atto primitivo del movimento, il cambio di direzione. E' generalmente calcolato geometricamente, data la posizione attuale, il vettore velocità corrente e la direzione da seguire.
- *Seek*: l'atto di sterzare l'individuo verso una specifica posizione nello spazio globale. Questo comportamento deve regolare la velocità allineandola verso il target ogni volta, per arrivare alla posizione senza superarla.
- *Flee*: comportamento simile ma duale rispetto a *Seek*: la direzione sarà opposta al target prefissato.
- *Pursuit*: l'atto di inseguimento dell'obiettivo, differisce da *Seek* in quanto il target ora è mobile e la direzione e velocità da mantenere saranno soggette a maggiori variabili. Solitamente si fa uso di un semplice algoritmo di previsione della traiettoria della target per meglio impostare il movimento.

- *Evasion*: l'atto inverso all'inseguimento, e fa uso del comportamento Flee per fuggire da un certo individuo in movimento.
- *Obstacle Avoidance*: comportamento che consente all'individuo di seguire una traiettoria priva di ostacoli presenti nell'ambiente circostante
- *Wander*: il vagare, un movimento casuale nello spazio. I comportamenti di *Explore* e *Foraging* sono spesso collegati al Wandering, il primo alla scoperta dell'ambiente circostante, il secondo al movimento casuale alla ricerca di risorse.
- *Path Follower*: abilita l'individuo al perseguimento di un percorso predeterminato come possono essere strade, corridoi o tunnel. Non è inteso come un percorso rigido(es. un treno su di una rotaia) ma piuttosto come un movimento prodotto da persone che si muovono in un corridoio: i percorsi individuali rimangono vicini e spesso paralleli alla linea centrale del corridoi ma liberi di deviare.
- *Separation*: la capacità di un individuo, facente parte di un gruppo, di mantenere una distanza di separazione dagli altri individui evitando il collasso del modello in un'unico punto.
- *Cohesion*: l'abilità di un membro di un gruppo di stare vicino agli altri membri, senza allontanarsi eccessivamente.
- *Alignment*: dà all'individuo la capacità di allinearsi nella stessa direzione degli altri individui del gruppo.
- *Leader Following*: il comportamento emergente quando uno o più individui ne seguono un altro designato come leader. Generalmente il comportamento di ciascun individuo è quello di stare vicino al leader, mantenendo una certa distanza per non provocare sovraffollamento, avendo cura di non trovarsi sul suo percorso. Nel caso in cui i seguaci siano più di uno, si dovranno evitare collisioni tra di essi.
- *Combining Behaviours*: i comportamenti finora descritti possono essere visti come blocchi fondamentali per comportamenti

più complessi. Per comporre nuovi comportamenti, più simili a quelli reali, dobbiamo selezionarne alcuni e miscelarli insieme. Esistono principalmente due modi di combinare questi diversi comportamenti: 1) variare da un comportamento all'altro in modo sequenziale, eseguendone uno per poi passare a quello successivo; 2) miscelare due o più comportamenti contemporaneamente, eseguendo una traiettoria ibrida: il classico esempio è un'evasione evitando gli ostacoli.

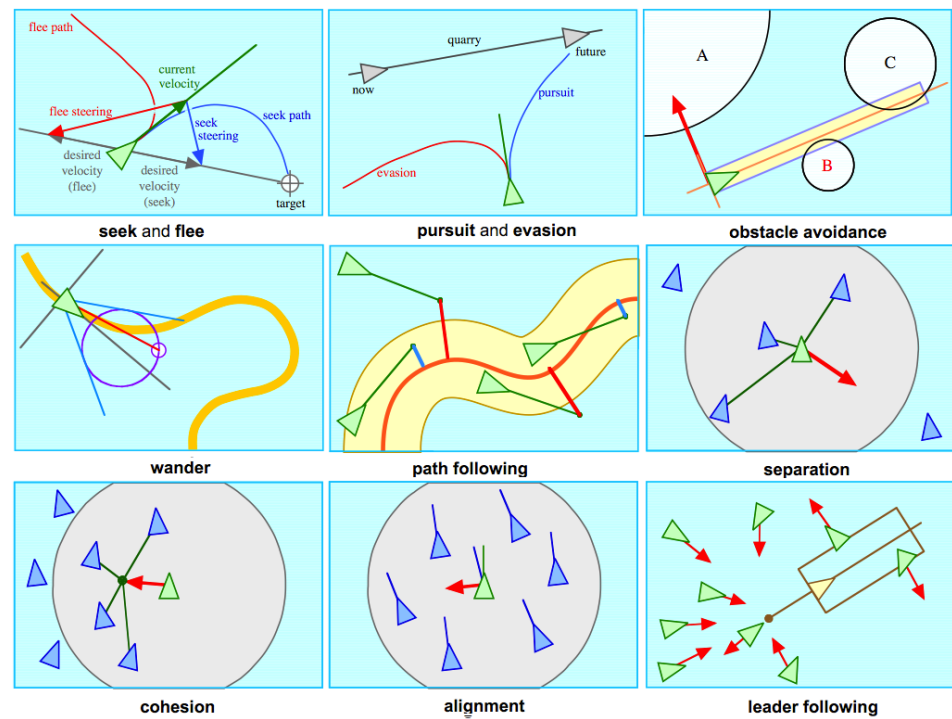


Figura 2.1: Possibili comportamenti di un individuo

Capitolo 3

Tecniche di Modellazione delle Masse

Diverse sono le tecniche che si possono ritrovare in letteratura per quanto riguarda il problema in esame. Anche se tutte differiscono tra loro, in comune hanno spesso un approccio base su cui poi sviluppare l'algoritmo specifico. Possiamo riassumere le tecniche in quattro categorie:

1. *Sistemi Fluidodinamici*
2. *Flocking Systems*
3. *Particle Systems*
4. *Behavioural Model o Autonomous Character*

Vediamo ora in particolare questi approcci, riportando per ognuno la bibliografia associata.

3.1 I Sistemi Fluidodinamici

Questi tipi di modelli rappresentano un modello continuo di una massa usandone una sola specifica caratteristica, come ad esempio la densità o la velocità, distribuendola su di una griglia corrispondente all'ambiente in cui è virtualmente disposta e dove essa può muoversi. Si

astrae così dal modello fisico, rappresentando ad una ad una le caratteristiche della folla con una metafora naturale quale è il movimento di un fluido su di una superficie o lungo tubi. Non possiamo dunque integrare sul comportamento di ciascun individuo ma ci si interfaccia con l'intera massa, modellandone i comportamenti risultanti in termini, sostanzialmente, di flusso.

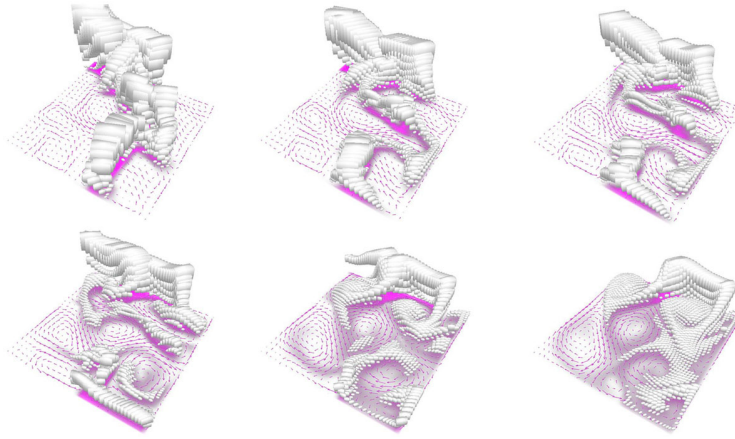


Figura 3.1: Esempio di un sistema fluidodinamico

3.2 Flocking Systems

Questo modello fu introdotto per la prima volta da Reynolds [14] e mostrato con un video dimostrativo al Electronic Theater al SIGGRAPH '87. Reynolds ha sviluppato un modello coordinato per il movimento di animali, come uno stormo di uccelli. Chiamò genericamente “boids” le creature simulate: da qui il termine di Flocking Boids, richiamato anche in opere di altri autori. I movimenti implementati da questo algoritmo furono soltanto tre: la separazione, l'allineamento e la coesione. Grazie a questi tre pattern comportamentali riuscì nell'intento di simulare un intero gruppo di uccelli. Se andiamo più nel dettaglio possiamo vedere che nonostante ogni boid abbia accesso diretto all'intera descrizione geometrica della scena, il movimento in formazione

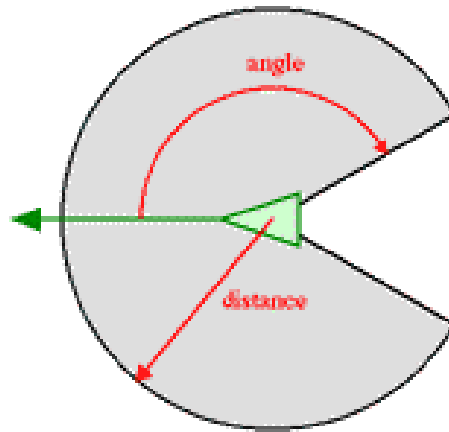


Figura 3.2: Visuale di un Boid

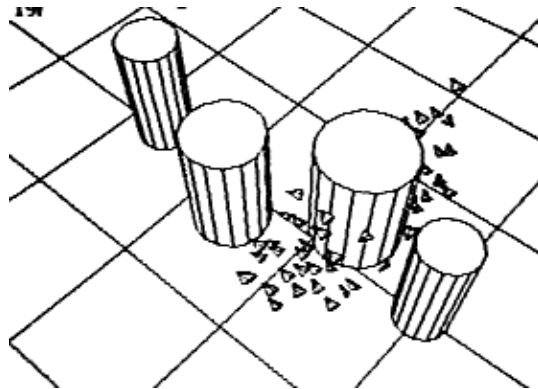


Figura 3.3: Esempio di simulazione di un insieme di Boid

necessita solamente che le creature reagiscano ad un numero ristretto di stimoli, come ad esempio l'osservazione dei propri vicini in uno spazio limitato dalla singola capacità visiva. Il vicinato dunque è caratterizzato da una distanza, misurata tra i centri dei boid, e da un angolo di visuale, misurato dalla propria direzione di volo: gli individui che stanno al di fuori di queste sono ignorati.

3.3 I Sistemi a Particelle (Particle Systems)

In questo approccio ogni entità nella folla è rappresentata come una singola particella interagente con le altre particelle attraverso determinate regole di attrazione/reazione e di collisione. Un modello sostanzialmente a Virtual Forces, ogni particella viene attratta da forze positive (come ad esempio un obiettivo o un leader), e respinta da forze negative (come possono essere ostacoli o predatori). Le particelle interagiscono inoltre con le superfici che le circondano; possono essere soggette a forze esterne e accoppiate con un medium che le ingloba (si ipotizzi un liquido). Un particle system è definito da: una descrizione del tipo di particelle presenti, le sorgenti principali per la loro generazione, la geometria (2D/3D) dell'ambiente e di eventuali ostacoli, l'evoluzione delle particelle nel sistema. La descrizione delle particelle ne deve definire puntualmente le caratteristiche: attributi spesso utilizzati sono la massa (che può essere soggetta a forza gravitazionale), il suo tempo di vita (deterministico o stocastico), le proprietà di diffusione (perturbazioni casuali in velocità o in accelerazione), la sua carica (es: in campi elettromagnetici), la probabilità di cambiare tipologia (in casi in cui ci si possa trovare nelle vicinanze di una maggioranza di particelle di un altro gruppo), etc... Le particelle seguono un intero ciclo di vita che va dalla generazione alla morte, passando per l'evoluzione e il movimento. La generazione avviene attraverso sorgenti che ne definiscono le peculiarità di partenza, come la loro posizione spaziale, la loro dimensione, la geometria, la direzione, la velocità e l'accelerazione iniziale. Esempi di evoluzione di particelle possono essere: trasformazioni spontanee, il movimento dato da attrazioni provocate da campi (gravitazionali, elettromagnetiche, etc...),

collisioni elastiche/anelastiche, trasformazioni dovute all'incontro con particolari particelle nelle vicinanze. Infine le particelle sono destinate a morire quando un apposito parametro di lifetime raggiunge il tempo di simulazione previsto, simulando l'invecchiamento della particella stessa.

La difficoltà maggiore di questo approccio è causata dall'elevato numero di particelle presenti ed interagenti nel sistema per modellare realtà complesse. Nonostante il parallelismo intrinseco al problema, le complicate interazioni tra particelle non favoriscono l'implementazione diretta. Questo comunque non impedisce di suddividere il problema nel suo insieme in parti, che poi possono essere eseguite in maniera concorrente: invece di distribuire le particelle nel cluster si potrà dividere lo spazio (bidimensionale o tridimensionale) in porzioni e simulare le dinamiche indipendentemente (fatta eccezione per le zone di frontiera) riunendo poi i risultati ottenuti.

3.4 Autonomous Animated Character

Autonomous Animated Character (in seguito AC) è la definizione che danno Liu Feng e Ronghua Liang[7] al modello di individui da loro proposto. Si tratta di un tecnica per la modellazione di singoli attori del sistema, ognuno dei quali modellato come un agente autonomo che segue tre funzioni primarie: percezione, pianificazione del comportamento e generazione del moto.

3.4.1 Perception Modeling

Le percezioni consentono agli AC la raccolta di informazioni riguardanti l'ambiente ad essi circostanti. Tuttavia è difficile modellare il meccanismo con cui un individuo recepisce informazioni provenienti dall'esterno ed ancora più ardua è la rappresentazione delle procedure informative. Mentre un robot infatti, osserva il mondo esterno tramite i propri sensori, ad un agente autonomo è concessa l'intera osservazione del mondo, modificando così il metamodello sottinteso. Per quanto riguarda la percezione visiva, ad esempio, un agente dovrebbe poter osservare solamente gli oggetti presenti in un certo spettro visivo, determinato da una distanza massima e da un angolo di osservazione,

e che non siano completamente/parzialmente coperti da altri oggetti, pertanto questo meccanismo deve essere quindi implementato in modo tale da avvicinare il modello di percezione a quello reale.

3.4.2 Intention Generator

Ogni autonomous character genera intenzioni coerenti attraverso un “intention generator”. Sostanzialmente esistono 2 tipi di intenzioni applicabili: quelle ricevute dagli animatori, i quali specificano percorsi o comportamenti da seguire, oppure quelle generate autonomamente. Il secondo caso è sicuramente quello più interessante dal punto di vista ingegneristico in quanto esprime la capacità da parte di un agente software di prendere decisioni autonome. Queste intenzioni sono dunque generate a partire dalle informazioni collezionate dal modulo percettivo ed ora elaborate in base al proprio stato e ai goal di ciascuno.

3.4.3 Behavior Planning

Quando la lista delle intenzioni è completata, l’AC pianifica una serie di comportamenti razionali per soddisfare le intenzioni. Alternando fasi di determinazione dei comportamenti dominanti a fasi di integrazione, si riuscirà a creare una roadmap di movimenti da utilizzare, basata sulla pianificazione, per il soddisfacimento delle intenzioni, ottimizzandone al tempo stesso il percorso. I comportamenti di base sono composti da movimenti fondamentali, ma spesso non sono solo la semplice concatenazione di questi; ogni movimento dovrebbe infatti essere aggiustato attraverso la conoscenza del percorso così da rendere le transizioni tra movimenti continue e “lisce”.

3.4.4 Comparazione con Agenti BDI

Il core del metamodello degli Autonomous Animated Character è in qualche modo analogo a quello degli agenti BDI (Belief-Desire-Intention): anche se con termini diversi ognuno di essi presenta un suo stato interno, un sistema percettivo dell’ambiente, una base di conoscenza interna, una lista di obiettivi da assolvere. Tuttavia il

modo con cui esprimere la conoscenza e gli obbiettivi e sostanzialmente diversa. E' sicuramente di interesse futuro l'applicazione di agenti BDI in un contesto di Crowd Simulation.

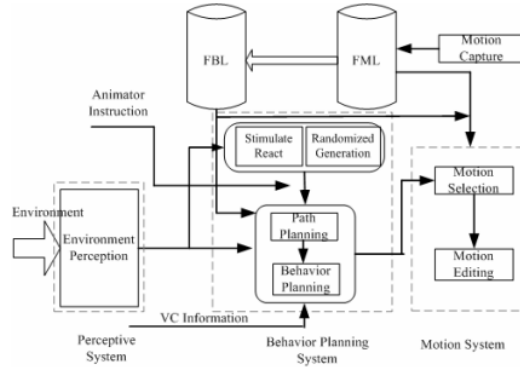


Figura 3.4: Struttura di un Autonomous Character

3.5 Sistemi a gruppi e a Leader

I sistemi con dinamiche di gruppo sono stati studiati in quanto è un comportamento molto ricorrente sia nel mondo animale che in quello umano. L'atto del raggruppamento rende ciascun individuo parte di una specie più forte, adattabile e sicura della semplice somma degli individui. Altre volte sono stati studiati in quanto il modello da simulare sottintende intrinsecamente uno o più gruppi. In tutte queste situazioni è spesso utilizzata la modellazione della folla attraverso un Leader-Follower Model, nel quale un individuo guida il gruppo per tutta la durata della simulazione, incentrando i ragionamenti collettivi in un'unico punto, rendendo più efficienti le stesse. Questa tipologia di modelli ha avuto un notevole riscontro in ambiti di simulazioni di situazioni di pericolo/emergenza. Infatti è stato possibile simulare diversi scenari in cui grandi masse di individui in spazi chiusi vengono guidati attraverso le uscite seguendo dei capogruppi. Questi fungevano da persone preparate alle situazioni di emergenza predisposti dalla dirigenza del locale. Questo ha dato un notevole contributo allo svi-

luppo di tecniche di evacuazioni di ambienti o di prese di decisioni in situazioni critiche, ottimizzando, in aggiunta, il numero e le modalità d'uso dei leader. Inoltre sono spesso utilizzati approcci ibridi, come integrazione di gruppi di partecipanti ed agenti individuali, e le dinamiche, in questo caso, diventano ancora più complesse in quanto individui possono: integrarsi in gruppi esistenti; lasciare il gruppo attuale e unirsi ad un'altro; formare un proprio gruppo.

3.5.1 Dynamic Grouping Algorithm

Nelle simulazioni di folla basate sul modello ad inseguimento di un leader, è molto importante definire come vengano raggruppate le persone. Per ogni gruppo bisogna anche assicurarsi che sia presente uno ed un solo capogruppo. Questi problemi si possono risolvere in due modi: un raggruppamento statico definito a priori dal progettista della simulazione, oppure mediante un algoritmo automatico per l'aggregazione degli individui. Certamente la situazione più interessante da osservare è la creazione dinamica di gruppi e l'elezione del proprio leader da parte dei partecipanti. Un algoritmo proposto in letteratura è il Dynamic Grouping Algorithm, il quale prevede la caratterizzazione di ciascun individuo con un parametro di "dominazione". Il partecipante confronterà i livelli di dominazione tra gli agenti circostanti, inseguendo quello con parametro maggiore, ma se a sua volta presenta un livello maggiore si staccherà dal gruppo a cui appartiene, iniziando a percorrere la propria strada, con la possibilità di diventare a sua volta un capogruppo o riaggregarsi ad un gruppo avente un leader più forte.

Capitolo 4

Opere in Letteratura

Ricercando opere sull'argomento, possiamo notare come i primi studi pubblicati siano riguardanti l'ambiente cinematografico. Con il progredire delle tecnologie e delle risorse computazionali, le grandi case cinematografiche hanno avuto un interesse sempre maggiore nel realizzare scene con l'utilizzo di tecniche di rendering ogni volta più avanzate. Avere un modello di riferimento per controllare i movimenti dei personaggi in modo automatico fu la necessità immediatamente successiva, al fine di sollevare gli sviluppatori dal dirigere manualmente le scene. A questo proposito, il primo lavoro che possiamo citare è un cortometraggio degli autori Amkraut, Girard e Karl[2] chiamato "Eurhythmy" che fu presentato al SIGGRAPH 1985 e successivamente concluso nel 1989. Il movimento in formazione degli uccelli è stato ottenuto mediante un vettore di forza globale che guida il flusso dello stormo.

Tuttavia l'opera pionieristica in questo campo è stata quella di Reynolds[14] del 1987, in cui descrive un modello comportamentale per simulare un movimento aggregato di uno stormo di uccelli. Il paper tecnico in cui descriveva il modello fu accompagnato da un cortometraggio chiamato "Stanley and Stella in: Breaking the Ice" e mostrato all'Electronic Theater, SIGGRAPH 1987. L'idea rivoluzionaria fu che un comportamento complesso di un gruppo di attori può essere ottenuto attraverso semplici regole locali applicate ai membri del gruppo invece di sostanziali condizionamenti globali. Il modello sviluppato è un complesso Particle System ove ogni uccello (chiamato

to boid) è una particella. Ogni boid è implementato come un agente indipendente che naviga in accordo con la sua percezione locale dell'ambiente, le leggi che simulano fisica e il suo set di comportamenti integrato. Reynold estese ulteriormente il suo lavoro includendo vari tipi di comportamenti quali: raggiungere obbiettivi, evitare ostacoli, seguire un percorso.

Terzopoulos e Tu[16] proposero un framework per l'animazione di pesci virtuali. Oltre a complessi comportamenti individuali basati sulle percezioni dell'ambiente, i pesci virtuali esibiscono movimenti collettivi inaspettati come il nuoto in formazione e l'evasione dai predatori.



Figura 4.1: Terzopoulos e Tu, simulazione di pesci in formazione(in alto) ed evasione dal predatore

Brogan e Hodgins[5] simularono comportamenti di gruppo per sistemi con dinamiche significanti. Comparandolo ai precedenti lavori, è stato realizzato un movimento più realistico attraverso la considerazione delle leggi fisiche e delle proprietà del moto degli individui, come i momenti d'inerzia, il bilanciamento e l'equilibrio. Il loro algoritmo per controllare i movimenti consiste in due fasi: nella prima un modello percettivo determina quali creature e ostacoli sono visibili per ogni individuo; successivamente un algoritmo collocativo determina la posizione desiderata per ciascun attore data l'attuale posizione e velocità delle creature ed ostacoli percepiti.

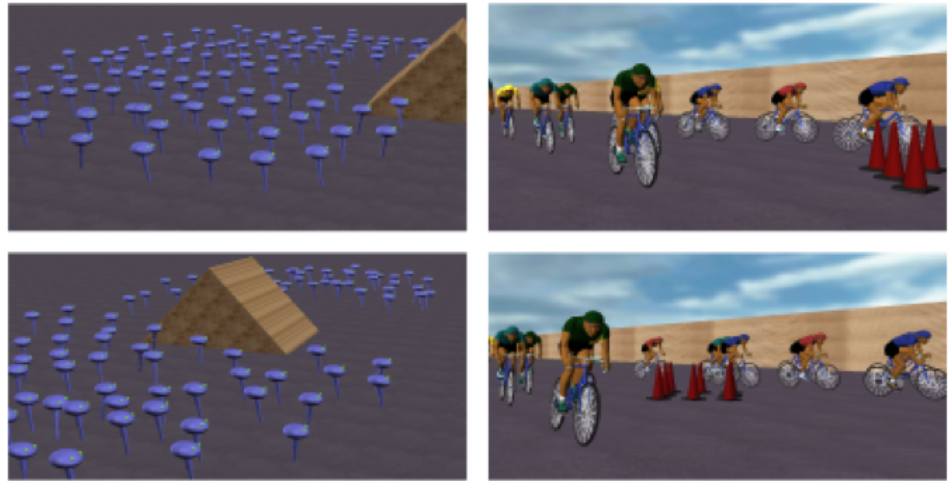


Figura 4.2: Brogan e Hodgins, simulazione di robot ad una gamba(sinistra) e ciclisti (destra) nell'atto di evitare un ostacolo

Bouvier[4] ha utilizzato un approccio simile ai boid di Reynolds per simulare folle umane. Usando una combinazione di particle systems e transition network riuscì a modellare uno spazio urbano con successo. A più basso livello, forze attrattive(gli obbiettivi) e repulsive(gli ostacoli), come metafora delle forze elettromagnetiche, hanno permesso alle persone di muoversi attraverso l'ambiente in maniera fluida e coerente. Ad un livello più alto sono state utilizzate reti di transizioni per modellare il raggiungimento di goal successivi, cambiamenti di densità di popolazione locali ed eventi globali.

Musse e Thalmann in [11] e [12] presentarono un modello gerarchico per una simulazione real-time rappresentante una folla umana virtuale. Il loro modello è basato a gruppi (dette anche “comunità”) invece che a singole persone: questi infatti rappresentano strutture più intelligenti e meno numerosi dei singoli individui. Le comunità possono essere controllate con diversi livelli di autonomia: gruppi che seguono ordini (andare in una certa posizione, eseguire animazioni, ...) dettati dagli utenti a run-time; comportamenti predeterminati da un copione; gruppi autonomi che usano eventi e reazioni per creare comportamenti più complessi.

Più recentemente Ulicny e Thalmann[17] in [18] hanno presentato un simulatore comportamentale di folle di persone con un’architettura modulare per sistemi multiagente che supporta sia comportamenti autonomi che predeterminati. Nel loro sistema il comportamento è elaborato a livelli, ove le decisioni sono prese da regole comportamentali e l’esecuzione è gestita da una macchina a stati finiti.

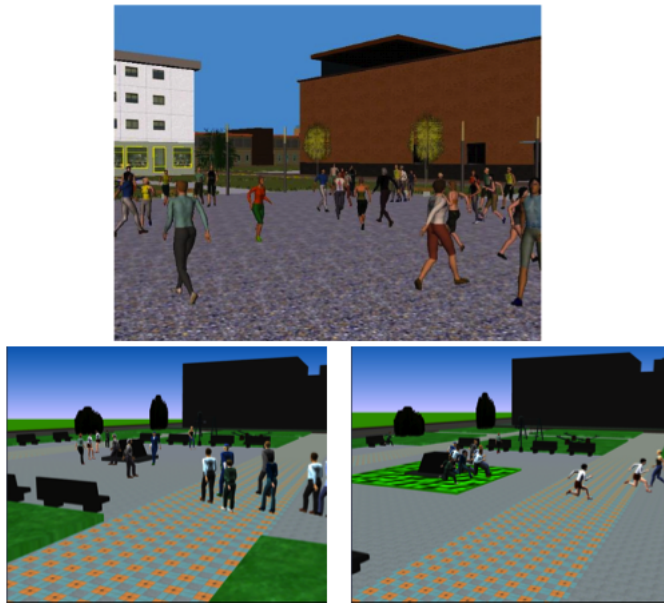


Figura 4.3: Ulicny e Thalmann: Simulazione della folla in modalità autonoma(in alto), situazione prima di una fuga di gas(in basso a sinistra) e dopo(in basso a destra)

Capitolo 5

Caso di studio: Crowd Steering Model

5.1 Il caso di studio

Il modello proposto è uno scenario Crowd Steering, ambientato in un museo, con una serie di stanze connesse tra loro e sui cui pavimenti è presente una rete di dispositivi (o sensori/nodi) che possono scambiarsi informazioni l'un l'altro in base alla prossimità degli stessi: in particolare possono avvertire la presenza di visitatori nella loro vicinanza e mantenere informazioni sulle esposizioni in essere nel museo. Ogni stanza ha quattro uscite e sono interconnesse attraverso corridoi esterni. I visitatori sono forniti di un dispositivo (uno smartphone, PDA o un device dedicato) che, attraverso l'interazione con i sensori sul pavimento, li guideranno verso l'obiettivo suggerendo dinamicamente la direzione da seguire. Il percorso dovrà essere il più breve possibile ma evitando le zone sovraffollate al tempo stesso. Si utilizzerà come metafora fondante quella di “field”, o campo, centrato ed emesso da ogni attrazione del museo che perde potenza all'aumentare della distanza dal centro dello stesso, pertanto ogni attrazione è assimilabile ad un dispositivo in grado di emettere segnali captati dai sensori a terra. Questi sono in grado di recepirne la potenza: ad intensità maggiore corrisponde una minore distanza, infatti, come nella metafora di onda elettromagnetica, si modellerà il campo come decrescente proporzionalmente alla distanza dalla fonte.

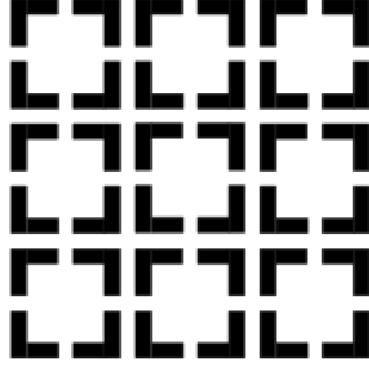


Figura 5.1: Crowd Steering Museum's Scenario

5.2 Il modello Alchemist

Il modello è fondamentalmente costituito da un'infrastruttura di nodi. I dispositivi mobili sono modellati come agenti che dinamicamente si collegano ai sensori a terra immediatamente vicini, entro un certo raggio "r", e dai quali possono recuperare dati e misure di lontananza. Elaborando queste informazioni suggeriscono ai visitatori il percorso migliore verso la loro destinazione. Anche i visitatori sono modellati

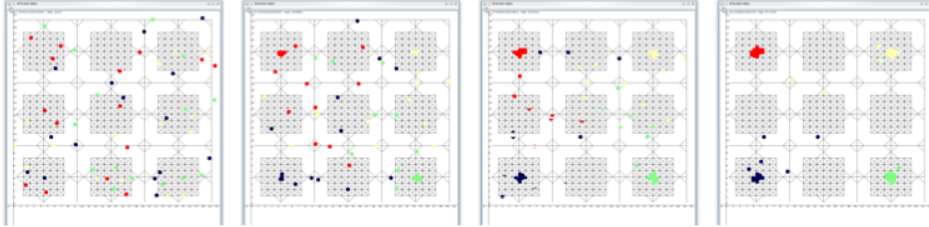


Figura 5.2: Alchemist – Simulation's stages

come agenti: collocati spazialmente nel museo, tendono a seguire le indicazioni fornite dai dispositivi in loro possesso, muovendosi ad ogni turno di una lunghezza discreta verso il loro obiettivo, evitando allo stesso tempo di avvicinarsi agli altri agenti visitatori e mantenendo

una distanza minima per non incorrere in collisioni. Modellando questo problema in un approccio di tipo SAPERE, dobbiamo definire la forma di interazione tra gli attori del sistema: gli obbiettivi, i nodi, i dispositivi mobili e gli agenti. SAPERE si basa sull'approccio a LSA e regole che definiscono la dinamica e l'evoluzione del sistema nel tempo. Una LSA non è altro che una tupla di valori ordinati $\langle v_1, \dots, v_n \rangle$. Le tre forme utilizzate in questo modello sono:

$$\begin{aligned} &\langle \text{source}, id, type, N_{max}, \pi, \mu, type' \rangle \\ &\langle \text{field}, id, type, value, \pi, \mu, type', tstamp \rangle \\ &\langle \text{pre_field}, id, type, value, \pi, \mu, type', tstamp \rangle \end{aligned}$$

Figura 5.3: Crowd Steering LSA

Ogni attrazione del museo viene modellata con una tupla LSA di tipo source, mentre il campo generato da questa attraverso la sequenza field e quella pre-field. Ad ogni tupla vengono assegnati anche valori aggiuntivi quali: un'etichetta (id), il tipo di campo (type), il massimo valore del campo (Nmax) e due valori usati per elaborare il nuovo valore del campo che deve essere propagato o trasformato (π e μ).

Le eco-laws utilizzate dal modello invece, possono essere riassunte in fig. 5.4.

$$\begin{aligned} \langle \text{source}, id, type, N_{max}, \pi, \mu, type' \rangle &\mapsto \langle \text{source}, id, type, N_{max}, \pi, \mu, type' \rangle, \\ &\quad \langle \text{field}, id, type, N_{max}, \pi, \mu, type', \#T \rangle \\ \langle \text{field}, id, type, N, \pi, \mu, type', t \rangle &\mapsto \langle \text{field}, id, type, N, \pi, \mu, type', t \rangle, \\ &\quad + \langle \text{pre_field}, id, type, \pi(N, \#D), \pi, \mu, type', t \rangle \\ \langle \text{pre_field}, id, type, N, \pi, \mu, type', t \rangle, \langle \text{field}, id', type', M, \pi', \mu', Type, t' \rangle &\mapsto \langle \text{field}, id, type, \mu(N, M), \pi, \mu, type', t \rangle, \\ &\quad \langle \text{field}, id', type', M, \pi', \mu', Type, t' \rangle \\ \langle \text{field}, id, type, N, \pi, \mu, type', t \rangle, \langle \text{field}, id, type, M, \pi, \mu, type', t' + t \rangle &\mapsto \langle \text{field}, id, type, M, \pi, \mu, type', t' + t \rangle \\ \langle \text{field}, id, type, N, \pi, \mu, type', t \rangle, \langle \text{field}, id, type, M, \pi, \mu, type', t \rangle &\mapsto \langle \text{field}, id, type, \max(M, N), \pi, \mu, type', t \rangle \\ \langle \text{field}, id, type, N, \pi, \mu, type', t \rangle, \langle \text{field}, id', type, N + M, \pi, \mu, type', t' \rangle &\mapsto \langle \text{field}, id', type, N + M, \pi, \mu, type', t' \rangle \end{aligned}$$

Figura 5.4: Eco Law

Quando un'attrazione viene posta nel museo emette una LSA di tipo source, ed appena questa viene recepita da un nodo, quest'ultimo

emette a sua volta una tupla di tipo field, assegnando come valore iniziale del campo il valore massimo specificato dalla sorgente (regola 1). Quando un nodo contiene una tupla di tipo field, esso diffonde ad un nodo vicino scelto casualmente una tupla di tipo pre-field modificando il valore del campo, computandone il suo nuovo valore come funzione della distanza tra i due nodi (regola 2) grazie alla funzione π . Alla presenza di una tupla di tipo field e pre-field nello stesso nodo ed ove i valori type dipendono tra loro, si attiva la terza eco-law, che modifica il valore del campo attraverso la funzione μ . Questa è appositamente creata per diminuire il campo nelle direzioni in cui sono presenti zone sovraffollate, penalizzandoli nella scelta. Ogni tupla è inoltre identificata da un tempo di creazione t , permettendo di riconoscere quali siano le informazioni più aggiornate nel caso di LSA apparentemente simili. Le regole 4 e 5 sono atte a capire quali siano le informazioni corrette in caso di LSA multiple: tuple con tempi di creazione diversi indicano aggiornamenti del campo, oppure tuple con ugual tempo di creazione ma valori di campo differente indicano che l'informazione ha compiuto dei cicli attraverso i sensori, ritornando indietro al mittente. Nel primo caso si mantiene l'LSA a tempo maggiore, cioè più recente; nel secondo caso è opportuno mantenere quella con il valore di campo maggiore, indicando così la via più breve verso l'obiettivo ed evitando inutili percorsi circolari. Le persone sono modellate come agenti che decidono in maniera autonoma quale traiettoria seguire: questa verrà scelta stocasticamente, ma con probabilità proporzionale al valore del campo nelle direzioni percorribili (cioè non limitate da muri o altre persone). L'atto del movimento dei visitatori invece, è modellato come una reazione chimica, con preconditione nulla e propensity rate costante: questa scatena nell'agente il meccanismo di movimento appena descritto. -rate di attivazione-

5.3 Il modello RePAST

Il modello ha il compito di replicare fedelmente i comportamenti descritti e implementati da quello realizzato in Alchemist, ma con una diversa metodologia di programmazione, ovvero l'ABM: Repast infatti, è uno dei principali simulatori che propone l'Agent Based Modeling

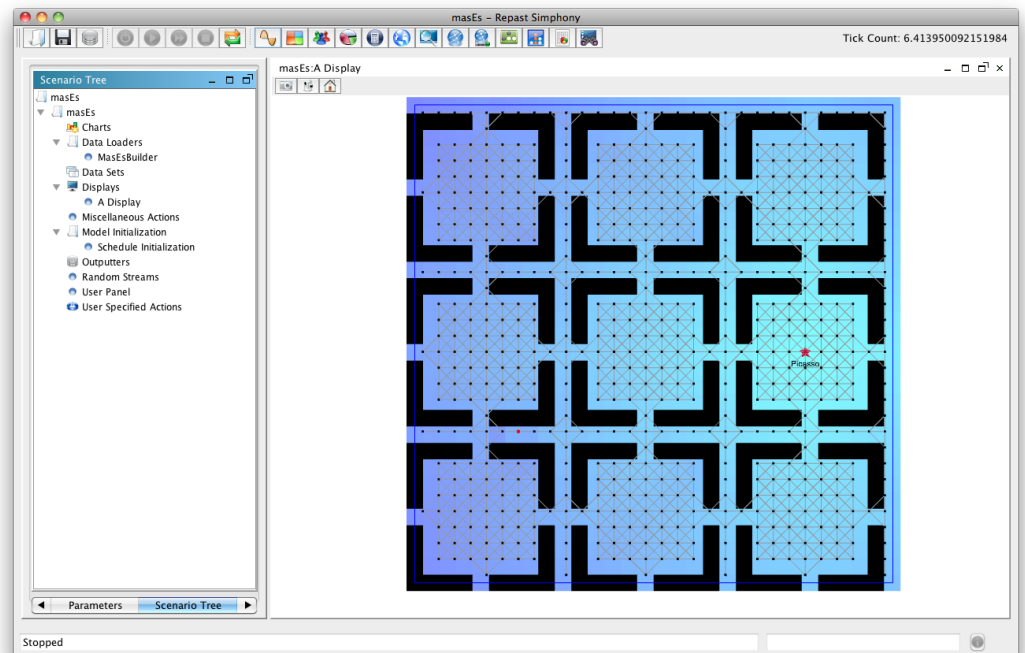


Figura 5.5: Eco Law

come livello di astrazione primario, rispetto alle precedenti strategie di programmazioni (quali quella procedurale, ad oggetti, etc). Si è così identificato come concetto fondamentale primario quella di Agent, ricalcando e adattandolo al modello proposto dal caso di studio: uno scheletro che ha come scopo fondamentale quello di realizzare il sistema di interazione con lo scheduler del simulatore affinché abbia lo stesso metodo di attivazione delle reazioni, come visto precedentemente in Alchemist. All'attivazione di ciascun agente viene schedulato il primo turno con tempo pari al logaritmo di un numero random diviso per il rate associato. Ad ogni turno viene richiamato il metodo che ogni attore del sistema utilizzerà, modificandolo opportunamente per i propri scopi; contestualmente a questa fase viene inoltre programmato il prossimo turno dell'agente. Person, Sensor e Target sono classi derivanti da Agent e per questo ne ereditano il comportamento auto-pianificante.

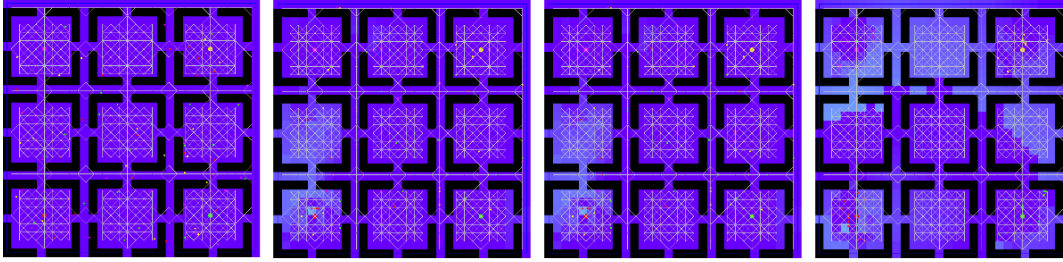


Figura 5.6: Repast – Simulation's Stages

- Person è la classe che ingloba il comportamento del PDA e del visitatore in un'unica astrazione: ad ogni azione infatti esso seleziona le direzioni nelle quali è permesso spostarsi (quelle non ostacolate da altre persone o muri del museo), e si muove in maniera stocastica; la probabilità di ciascun verso è proporzionale al valore del campo segnalato dal sensore a terra.
- Attraverso la classe Target si vogliono modellare le attrazioni presenti nel museo: queste, una volta posizionate, provvedono a scoprire automaticamente i sensori vicini (entro un certo raggio)

e ad ogni turno inviare ad uno di essi in modo casuale, il proprio valore di campo massimo.

- Sensor infine, è la classe di oggetti, che chiameremo anche nodi, in grado di mantenere il valore del campo nel punto in cui esso è posizionato. Inoltre, grazie ai collegamenti fra nodi, possono scambiarsi informazioni relative a campi dello stesso tipo, cioè che fanno riferimento alla stessa attrazione.

In Repast il concetto di network (o anche grafo) è fornito built-in e particolarmente ottimizzato, con la possibilità di realizzare collegamenti sia orientati che non, specificando i vertici e l'eventuale peso (in caso di grafi pesati). Sfruttando questa astrazione si può creare facilmente il network di sensori. Ad ogni turno verrà richiamata la funzione di “spread” (emissione) del valore del campo verso uno dei sensori adiacenti: quest'ultimo, alla ricezione, calcolerà il nuovo valore da mantenere memorizzato sottraendo al quantitativo ricevuto la distanza tra i due sensori. Verrà sottratto inoltre un ulteriore coefficiente per ogni persona che si trova nelle vicinanze dell'agente ricevente, come visto precedentemente. Si penalizzeranno così le direzioni con una maggior concentrazione di agenti.

Bibliografia

- [1] M. Ab Karim, A. Talib, A. Ismail, and B. Belaton. A crowd model for emergency and panic situations: Structure, characteristics and behaviours. *Multi-Agent Systems for Society*, pages 178–191, 2009.
- [2] S. Amkraut, M. Girard, and G. Karl. Motion studies for a work in progress entitled eurythmy. *SIGGRAPH Video Review*, 21, 1985.
- [3] M. Azahar, M. Sunar, D. Daman, and A. Bade. Survey on real-time crowds simulation. *Technologies for E-Learning and Digital Entertainment*, pages 573–580, 2008.
- [4] E. Bouvier, E. Cohen, and L. Najman. From crowd simulation to airbag deployment: particle systems, a new paradigm of simulation. *Journal of Electronic Imaging*, 6(1):94–107, 1997.
- [5] D. Brogan and J. Hodgins. Group behaviors for systems with significant dynamics. *Autonomous Robots*, 4(1):137–153, 1997.
- [6] A. Bruzzone and R. Signorile. Crowd control simulation in java based environment. *SIMULATION SERIES*, 31:23–28, 1999.
- [7] L. Feng and R. Liang. Intelligent crowd simulation. *Computational Science and Its Applications?ICCSA 2003*, pages 962–962, 2003.
- [8] Q. Ji and C. Gao. Simulating crowd evacuation with a leader-follower model. *IJCSES*, 1(4):250, 2007.
- [9] A. Lerner, Y. Chrysanthou, A. Shamir, and D. Cohen-Or. Data driven evaluation of crowds. *Motion in Games*, pages 75–83, 2009.

- [10] T. Li and C. Wang. An evolutionary approach to crowd simulation. *Autonomous Robots and Agents*, pages 119–126, 2007.
- [11] S. Musse and D. Thalmann. A model of human crowd behavior: Group inter-relationship and collision detection analysis. In *Computer Animation and Simulation*, volume 97, pages 39–51. Citeseer, 1997.
- [12] S. Musse and D. Thalmann. Hierarchical model for real time simulation of virtual human crowds. *Visualization and Computer Graphics, IEEE Transactions on*, 7(2):152–164, 2001.
- [13] W. Reeves. Particle systemsma technique for modeling a class of fuzzy objects. *ACM Transactions on graphics*, 2(2):91–108, 1983.
- [14] C. Reynolds. Flocks, herds and schools: A distributed behavioral model. In *ACM SIGGRAPH Computer Graphics*, volume 21, pages 25–34. ACM, 1987.
- [15] C. Reynolds. Steering behaviors for autonomous characters. In *Game Developers Conference*, volume 1999, pages 763–782. Citeseer, 1999.
- [16] D. Terzopoulos, X. Tu, and R. Grzeszczuk. Artificial fishes: Autonomous locomotion, perception, behavior, and learning in a simulated physical world. *Artificial Life*, 1(4):327–351, 1994.
- [17] B. Ulicny and D. Thalmann. Crowd simulation for interactive virtual environments and vr training systems. *Computer Animation and Simulation 2001*, pages 163–170, 2001.
- [18] B. Ulicny and D. Thalmann. Towards interactive real-time crowd behavior simulation. In *Computer Graphics Forum*, volume 21, pages 767–775. Wiley Online Library, 2002.
- [19] B. Zhan, D. Monekosso, P. Remagnino, S. Velastin, and L. Xu. Crowd analysis: a survey. *Machine Vision and Applications*, 19(5):345–357, 2008.