

ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA
SEDE DI CESENA

FACOLTA' DI INGEGNERIA

CORSO DI LAUREA IN INGEGNERIA INFORMATICA

Sistemi Multi-Agente LS

Swarm-bot: colonie di robot auto-organizzati

STUDENTE
Danilo D'Aversa

PROFESSORE
Andrea Omicini

Anno Accademico 2009/10

Indice generale

1 Introduzione.....	2
1.1 Swarm-bot.....	2
2 Hardware.....	4
3 Comportamenti sociali di uno Swarm-bot.....	6
3.1 Self-organization.....	6
3.2 Self-assembling.....	6
3.2.1 Self-assembling : trasporto.....	7
3.2.2 Self-assembling : avversità del terreno.....	8
3.3 Movimento coordinato.....	9
3.3.1 Movimento coordinato: trasporto.....	9
3.3.2 Movimento coordinato: avversità del terreno.....	10
3.4 Task allocation.....	11
4 Foraging scenario.....	12
5 Swarmanoid.....	14
Bibliografia.....	15

1 Introduzione

Nel corso di Sistemi Multi-Agente è stato introdotto un nuovo concetto di sistema incentrato appunto sull'idea di agente. Nel particolare caso del sistema *Swarm-bot* presentato di seguito si può notare come ogni suo elemento (*s-bot*) sia un agente (robotico), il che fa dello *Swarm-bot* un sistema multi-agente robotico. A prova di ciò possiamo vedere come molte delle caratteristiche proprie di un sistema multi-agente siano qui presenti, infatti:

- il sistema è composto da agenti (robotici), quindi decentralizzato nel controllo e anche nell'hardware;
- l'interazione tra gli elementi del sistema porta all'emergere di comportamenti sociali;
- il sistema è aperto, ossia nessuna assunzione sul numero dei suoi componenti;
- il sistema è “situato” nel senso che c'è una stretta relazione con l'ambiente nel quale esso sarà inserito.

Inoltre ogni *s-bot* ha tutte le caratteristiche dell'agente, poichè:

- è autonomo nel controllo, dato che ognuno ha un *controller* indipendente;
- possiede un comportamento non-deterministico;
- è proattivo poichè tenta di raggiungere un goal (completare un proprio task);
- è anche reattivo ai cambiamenti ambientali e sociali;
- è strettamente legato all'ambiente nel quale è immerso (sistema di sensori e di attuatori adatti all'ambiente nel quale saranno inseriti);
- è sociale, quindi capace di interagire con altri *s-bot* in modo visuale (LED), fisico (*gripper*) e anche sonoro;
- può comunicare solo in modo locale, infatti la telecamera può percepire i LED degli altri *s-bot* entro un raggio limitato e il *gripper* può essere usato solo con gli *s-bot* vicini;
- può sentire in modo locale, poichè i sensori di cui è fornito sono limitati nell'intorno dello *s-bot*;
- è mobile nel senso stretto del termine in quanto ha possibilità di spostarsi sul terreno;
- è “intelligente” dato che i parametri del proprio *controller* sono impostati tramite algoritmi evolutivi.

L'idea di copiare soluzioni ispirate dalla natura per risolvere problemi l'ho sempre ritenuta una scelta tanto intelligente quanto complessa. Quindi per me è stata semplice la scelta di questo particolare argomento, in aggiunta al fatto che la mia curiosità nei confronti degli insetti sociali e dell'organizzazione del loro complesso sistema è sempre stata fonte di grande interesse.

1.1 Swarm-bot

Il termine *swarm intelligence* fu coniato da Beni e Wang [1] per descrivere un nuovo approccio al controllo di sistemi robotici cellulari distribuiti. In seguito, Bonabeau et al. [2] estesero questa definizione includendo “ogni tentativo di progettare algoritmi o dispositivi distribuiti per risolvere dei problemi ispirati al comportamento collettivo delle colonie di insetti sociali o altre società di animali”. Questa nuova definizione promosse la *swarm intelligence* come nuovo paradigma computazionale per risolvere una grande varietà di problemi.

La *swarm robotic* applica la *swarm intelligence* al controllo di “colonie di robot” (*robotic swarm*), enfatizzando la decentralizzazione del controllo, le limitate capacità di comunicazione tra i robot, l'uso di informazioni locali, l'emergere di un comportamento globale e la robustezza ai malfunzionamenti [3].

All'interno di tale campo si inserisce il progetto *Swarm-bot* (<http://www.swarm-bots.org/>), sponsorizzato dal programma *Future and Emerging Technologies* appartenente alla Comunità Europea. Tale progetto si propone di creare sistemi affidabili, robusti e flessibili da applicare in esplorazioni spaziali semi-automatiche, salvataggi, esplorazioni subacquee, ecc.

L'elemento basilare di uno *Swarm-bot* è lo *s-bot*, un robot mobile che possiede sensori, motori, capacità computazionali relativamente limitate e abilità di connettersi e disconnettersi dagli altri *s-bot*. I loro collegamenti fisici forniscono la possibilità di assemblarsi insieme, in modo da creare un'entità unica collettiva che permette di risolvere problemi che un singolo *s-bot* non potrebbe, come ad esempio spingere o trascinare un oggetto pesante, attraversare una buca troppo grande, salire o scendere una collina troppo ripida, ecc. Ogni *s-bot* è capace di spostarsi, di sentire e di agire in base a decisioni e a informazioni locali. L'esecuzione di un compito globale è ottenuto senza pianificazione e controllo centralizzati, ma tramite l'interazione robot-robot e robot-ambiente. Tale decentralizzazione dell'hardware, insieme a quella del controllo, garantiscono una ottima robustezza ai malfunzionamenti per il sistema *Swarm-bot*.

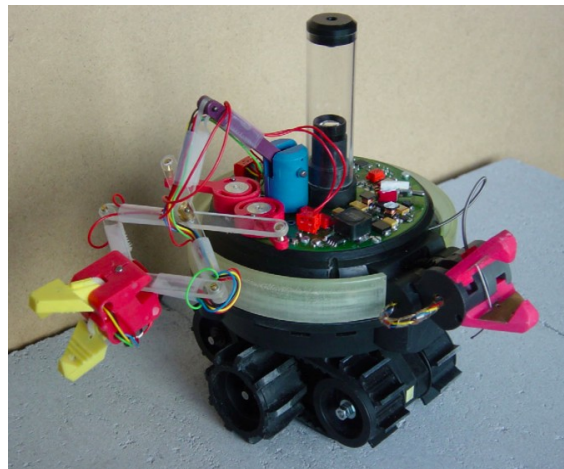
2 Hardware

Gli *s-bot* sono le unità elementari di uno *Swarm-bot*. Ogni *s-bot* è un robot capace di svolgere dei semplici compiti come muoversi autonomamente, percepire l'ambiente e afferrare oggetti. In più ogni *s-bot* è capace di comunicare e collegarsi con altri *s-bot*. Il sistema di trazione è composto da due *Treel* (*TRack* + *whEEL*), cioè una ruota e un cingolo affiancati ed guidati dallo stesso motore. Questa combinazione garantisce un sistema di guida differenziale, chiamato *Differential Treels Drive*, che porta due vantaggi:

- una efficiente rotazione sul posto, grazie all'ampiezza del diametro offerto dai due *Treel*
- una forma cilindrica del sistema di trazione, che si adatta bene alla forma del corpo (torretta) dello *s-bot* migliorandone la mobilità.

Treels e torretta possono ruotare in modo indipendente garantendo una maggiore flessibilità di movimento.

Nella torretta sono presenti diversi sistemi di sensori (di prossimità, di luce, di umidità, accelerometri), due *gripper* per afferrare oggetti o altri *s-bot* e anche dispositivi di comunicazione, come camera omnidirezionale, led colorati intorno alla torretta, emettitore e ricevitore di suoni. In aggiunta a questi ci sono anche sensori per reperire informazioni riguardo ai contatti fisici, allo sforzo e alle reazioni alle interconnessioni con gli altri *s-bot*, come sensori di torsione e di trazione.



Uno degli aspetti più critici di questo progetto è stata la realizzazione della connessione tra gli *s-bot* [4]. Tutti i metodi sviluppati precedentemente si basavano su un modello simile a quello di piedini e buchi (in cui vanno inseriti i piedini). Questa connessione è molto stabile, ma richiede un allineamento molto accurato per permettere la sua instaurazione. Benché garantisca maggiore rigidità meccanica, nel progetto *Swarm-bot* è stato scelto di non usare tale modello. Ciò ha semplificato la procedura di connessione e ha garantito una ampia tolleranza nel posizionamento e nell'allineamento.



Sono state implementate due modalità di interconnessione possibili tra gli *s-bot* [5] :

- rigida : resa possibile tramite un *gripper* (morsa) meccanico posizionato su di un lato della torretta. Questo *gripper* va ad aggrapparsi ad un anello a forma di T che circonda la torretta dello *s-bot*. Questa modalità di collegamento permette la formazioni di catene solide che consentono di passare diversi ostacoli. L'apertura del *gripper* è molto ampia e consente connessioni sicure in qualsiasi angolazione. Quando completamente chiuso tale meccanismo garantisce una presa ferma, che permette anche di sollevare un altro *s-bot* (non più di uno). Se la chiusura è solo parziale, lo *s-bot* è fornito di un meccanismo che permette dei movimenti di assestamento in diverse direzioni, così come osservato nel comportamento di alcuni insetti sociali;
- semi-flessibile : resa possibile tramite un altro *gripper* posizionato alla fine di un braccio meccanico mobile nelle tre direzioni dello spazio e situato sulla torretta. Questa modalità permette configurazioni in cui gli *s-bot*, collegati tra loro, mantengono un certo grado di libertà nei movimenti.



Come è stato osservato da esperimenti svolti [6], tali connessioni conferiscono agli *Swarm-bot*:

- prestazioni super lineari : in quanto i risultati raggiunti sono molto lontani da quelli ottenuti dalla semplice somma dei contributi dei singoli *s-bot*. Questa è una chiara indicazione che le connessioni fisiche giocano un ruolo costruttivo nella collaborazione tra gli *s-bot*;
- scalabilità limitata : la scalabilità dei risultati di alcune prove è limitata ad un massimo di 5 elementi per *Swarm-bot* assemblati. Questo limite è dovuto alle caratteristiche fisiche e meccaniche dei collegamenti usati dagli *s-bot*.

3 Comportamenti sociali di uno Swarm-bot

I meccanismi di maggiore interesse per tale progetto sono emersi dallo studio delle colonie di insetti sociali. L'auto-assemblamento (*self-assembling*), il movimento coordinato e la distribuzione dei compiti (*task allocation*) sono le tre abilità principali implementate nello *Swarm-bot*. Alla base di queste c'è l'auto-organizzazione (*self-organization*), aspetto fondamentale che permette al sistema di comportarsi in modi diversi in base alle situazioni che gli si presentano.

3.1 Self-organization

Per raggiungere un comportamento collettivo all'interno di una colonia di insetti si parte dalla nozione di auto-organizzazione (*self-organization*) [7]. L'auto-organizzazione spiega come un sistema può passare da uno stato disordinato ad uno ordinato solo tramite interazioni locali tra i suoi componenti, senza nessun riferimento al sistema in generale.

In uno stato disordinato il comportamento del sistema è influenzato dal risultato delle azioni casuali dei suoi elementi. Al cambiamento del valore di alcuni parametri il processo di auto-organizzazione può iniziare o meno, in base all'interazione tra due semplici meccanismi: feedback positivi e negativi. Un feedback positivo consiste nell'amplificazione di alcune proprietà del sistema che emergono dalle interazioni casuali tra gli elementi, questo può essere visto come un effetto “palla di neve” che fa crescere esponenzialmente nel tempo queste proprietà. Al contrario, il feedback negativo serve come meccanismo regolatore, e spesso è esso stesso il risultato di un'amplificazione. I feedback negativi e positivi cooperano per mantenere stabile lo stato del sistema, rendendolo robusto a dispetto delle influenze esterne.

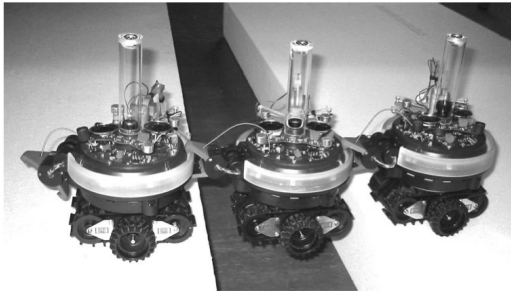
La capacità di riprodurre i meccanismi osservabili in un sistema auto-organizzato permette un efficiente controllo sugli *Swarm-bot*. Gli esperimenti dimostrano che la scelta di utilizzare sistemi neurali i cui parametri sono impostati tramite algoritmi evolutivi (*evolutionary algorithm*) porta all'ottenimento di ottimi risultati nell'auto-organizzazione in questo sistema robotico. L'evoluzione aggira il problema della decomposizione dello sviluppo dei *controller* su due livelli, cioè quello di trovare il meccanismo che porta all'emergere di un comportamento globale e quello di implementare tale meccanismo nel *controller* per gli *s-bot*. Infatti l'evoluzione artificiale fa affidamento sulla valutazione del sistema globale, cioè sull'emergere di un desiderato comportamento globale partendo dalla definizione dei comportamenti individuali. In più, l'evoluzione può utilizzare la ricchezza delle soluzioni possibili offerte dalle interazioni dinamiche agente-ambiente. Le soluzioni trovate tramite l'evoluzione sono semplici e in molti casi sono generali rispetto alle condizioni ambientali. I sistemi auto-organizzati così ottenuti sono robusti ai cambiamenti ambientali e sono anche scalabili.

3.2 Self-assembling

Per il progetto *Swarm-bot* l'aspetto particolarmente interessante derivante dall'auto-organizzazione è l'auto-assemblamento, cioè l'auto-organizzata creazione di strutture, definibile anche come “un processo reversibile nel quale preesistenti entità discrete di collegano una all'altra senza essere

dirette dall'esterno". Ciò permette agli *s-bot* di scegliere autonomamente quando e se assemblarsi in base all'ambiente che loro incontrano o in base al compito che devono svolgere.

Data l'ampiezza e la qualità dei risultati ottenuti da vari test [8] si può affermare che lo *Swarm-bot* si è qualificato come attuale stato dell'arte nell'auto-assemblamento.



3.2.1 Self-assembling : trasporto

Per trasportare un oggetto pesante (troppo pesante per un singolo *s-bot*) verso una zona-obiettivo uno *Swarm-bot* utilizza l'auto-assemblamento [9]. I *controller* degli *s-bot* sono stati implementati con due moduli:

- il modulo assemblamento (*assembly module*), che è incaricato di controllare lo *s-bot* finché non afferra l'oggetto (o un altro *s-bot* che è sua volta collegato all'oggetto);
- il modulo di trasporto (*transport module*), che è incaricato di controllare lo *s-bot* per il trasporto dell'oggetto verso la zona-obiettivo.

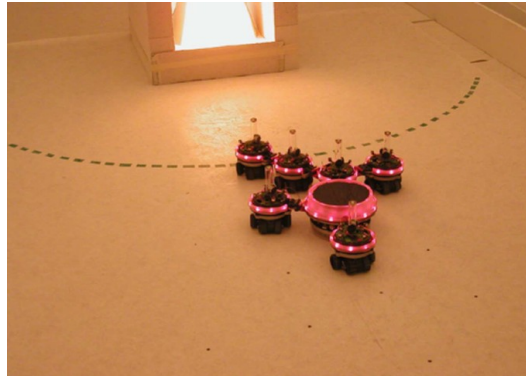
Il modulo assemblamento è scatenato dalla percezione (del colore dei LED) di un oggetto-preda o di uno *s-bot* che è collegato all'oggetto-preda. Questo porta lo *s-bot* a muoversi verso tale obiettivo, agganciarsi a questo e cambiare il colore dei propri LED in modo da renderlo uguale a quello dell'oggetto-preda (in tal modo esso stesso diventa parte dell'oggetto-preda). Una volta completato tale compito il controllo passa al modulo di trasporto che prima allinea lo *s-bot* in direzione della zona-obiettivo e poi spinge/tira l'oggetto per muoverlo verso la zona-obiettivo. Per evitare danni meccanici durante il trasporto se uno *s-bot* avvisa una torsione del *gripper* rigido che supera un dato limite allora si ferma. Tale *controller* è scalabile e si è dimostrato essere efficace [10].

Prendendo spunto dalle colonie di formiche, ad esempio nel caso dell'approvvigionamento, è stato notato che in generale una formica trova un oggetto-preda, prova a muoverlo e, dopo un certo numero di insuccessi, recluta altre formiche. A questo punto tutte quelle reclutate si raggruppano intorno all'oggetto, lo afferrano e spingono/tirano finché tale oggetto non si muove. Allo stesso modo gli *s-bot* devono localizzare, avvicinarsi, afferrare e portare un oggetto verso una zona-obiettivo, ma a differenza delle formiche agli *s-bot* è stata data la possibilità di afferrare non solo l'oggetto ma anche altri elementi del gruppo già attaccati all'oggetto. Il modo in cui gli questi si assemblano intorno a tale oggetto è determinato durante lo sviluppo del compito.

È facile notare che il trasporto cooperativo non richiede necessariamente l'assemblaggio. Infatti due sono le possibilità:

- i) collegare gli *s-bot* solo con l'oggetto : questa strategia porta a risultati soddisfacenti se la superficie “di appiglio” all'oggetto è abbastanza ampia, altrimenti è risultata perdente; o
- ii) permettere la formazione di strutture per spingere l'oggetto : questa strategia permette agli *s-bot* di spostare l'oggetto indipendentemente dall'ampiezza della superficie “di appiglio”, ma le performance sono scarse.

Studi sono ancora in corso per migliorare le prestazioni in modo da poter garantire un comportamento efficiente in qualsiasi situazione [11].

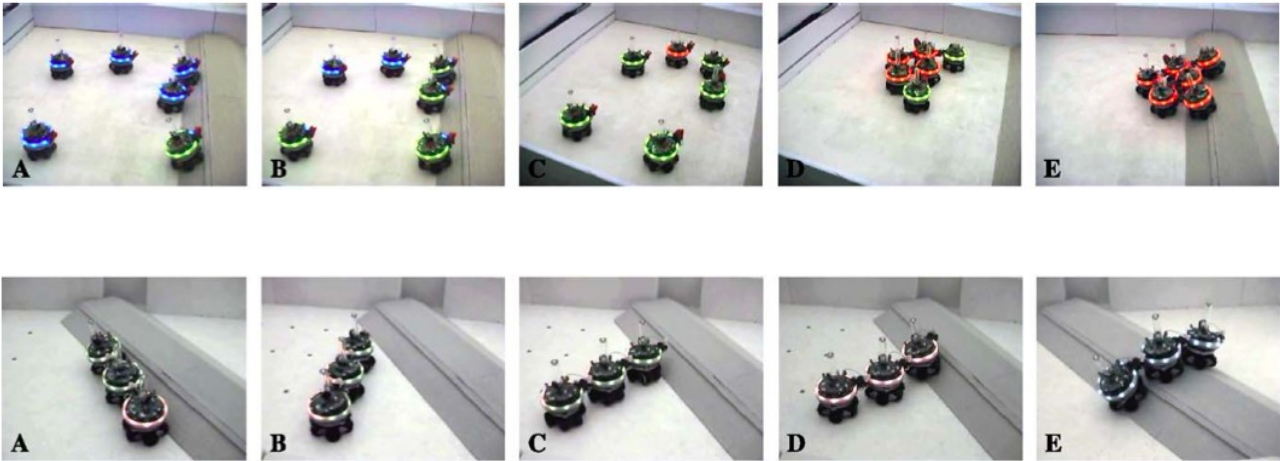


3.2.2 Self-assembling : avversità del terreno

Per superare delle avversità del terreno uno *Swarm-bot* può auto-assemblarsi seguendo diverse strategie. Studi di laboratorio [12] hanno messo a confronto le tre strategie principali nel caso attraversamento di una collina (*crossing hill task*):

1. agire sempre come una una grande entità auto-assemblata
2. lasciare che il sistema decida quando e se auto-assemblarsi
3. coordinare i sensori e gli attuatori degli *s-bot* connessi in modo che questi rispondano alle situazioni ambientali come una singola entita collettiva.

L'utilizzo della strategia 1 comporta sempre all'auto-assemblamento degli *s-bot* in un'unica entità indipendentemente dalla dimensione della collina, il che porta ad un decadimento delle performance nel caso in cui la collina possa essere superata anche dai singoli *s-bot* separatamente poichè il completamento del compito è più lento a causa dell'assemblamento. La strategia 2 permette delle prestazioni migliori rispetto alla 1 nel caso di situazioni che non richiedono l'assemblamento (collina poco ripida o collina assente), mentre nel caso in cui questo è necessario porta a prestazioni peggiori, poiché gli *s-bot* prima provano a svolgere il compito da soli e poi nel caso di insuccesso si assemblano e questo comporta il completamento più lento del compito a parità di risultato finale. La strategia 3 invece è migliore della strategia 2 poiché prevede che una volta scelto di auto-assemblare l'entità unica, questa continui a utilizzare i sensori e i motori di tutte le unità coinvolte. Quindi, prima di cominciare ad attraversare la collina, lo *Swarm-bot* assemblato controlla il suo posizionamento e se non è ritenuto abbastanza buono (perpendicolare alla collina) prima modifica la sua posizione e poi comincia l'attraversamento.



Da diversi esperimenti si è dimostrato il notevole miglioramento apportato dalla strategia 3 che ha permesso il completamento del compito assegnato (*crossing hill task*) in tutte le prove e nel modo più efficiente. Quindi il coordinamento dei sensori e motori degli *s-bot* connessi si è dimostrato il migliore per affrontare le varie situazioni ambientali.

3.3 Movimento coordinato

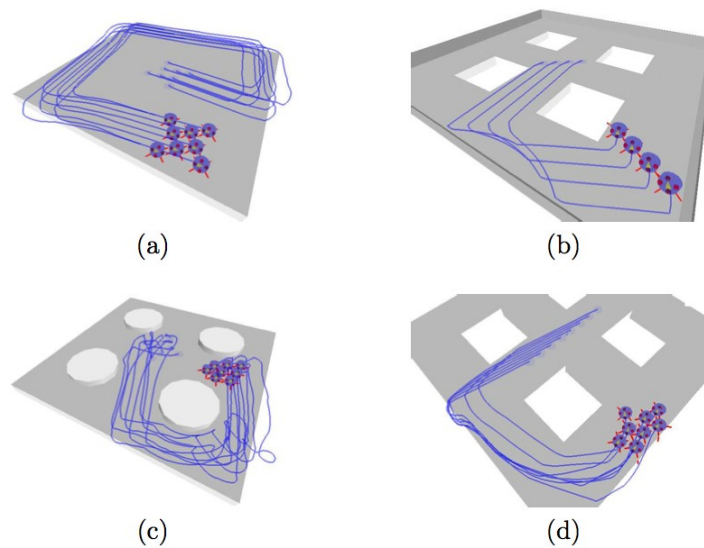
Il movimento coordinato rappresenta un'abilità basilare per uno *Swarm-bot* formato dall'unione di più *s-bot* ognuno dei quali indipendente nel controllo. Questi infatti devono scegliere la direzione unica per il loro movimento collettivo.

3.3.1 Movimento coordinato: trasporto

All'atto dell'auto-assemblamento in una unità unica può accadere che alcuni elementi di tale unità non abbiano visione della zona-obiettivo verso al quale trasportare l'oggetto-preda. Tale comportamento è stato oggetto di studi [13] poiché questa evenienza è possibile in diverse situazioni, ad esempio l'oggetto trasportato può coprire la visuale oppure le condizioni ambientali possono influire negativamente sui sensori (nebbia) oppure i sensori possono essere guasti. Quindi si è pensato a due possibili comportamenti di uno *s-bot* una volta che questi sia collegato all'oggetto-preda (o ad un altro *s-bot* a sua volta collegato all'oggetto-preda):

- se lo *s-bot* non è cieco, quindi ha visione della zona-obiettivo, allora allinea il suo corpo in direzione della zona-obiettivo e imposta la velocità massima del suo sistema di trazione in quel verso;
- se lo *s-bot* è cieco, quindi non ha visione della zona-obiettivo in cui trasportare l'oggetto-preda, allora regola la direzione e velocità del suo moto in base alla trazione percepita dai suoi sensori. Questo vuol dire che lo *s-bot* allinea la direzione del suo moto con la forza percepita e spinge o tira in base al verso di tale forza.

Il comportamento così sviluppato garantisce buoni risultati, infatti è stato studiato che se gli *s-bot* ciechi non sono più della metà di tutti i membri che trasportano un oggetto pesante allora il loro contributo è tra il 40% e il 60% del contributo fornito dagli *s-bot* non ciechi.



3.3.2 Movimento coordinato: avversità del terreno

Dallo studio delle colonie di insetti si è potuti arrivare alla distinzione di tre differenti forme di comunicazione tra questi:

- comunicazione indiretta (*stigmergic communication*) : una forma di comunicazione che avviene tramite l'ambiente, come risultato di azioni effettuate da alcuni individui, che indirettamente influenzano il comportamento di altri individui (ad es. i percorsi di fermoni);
- interazione diretta : una forma di comunicazione che implica la trasmissione (non mediata) di informazione, come risultato di azioni effettuate da alcuni individui, che direttamente influenzano il comportamento di altri individui (ad es. antennazione);
- comunicazione diretta : una forma di comunicazione che implica la trasmissione (non mediata) di informazione, senza necessità di alcuna interazione fisica (ad es. stridulazione).

Generalmente per il coordinamento di uno *Swarm-bot* assemblato si utilizza solo la forma di comunicazione diretta rappresentata dal collegamento fisico tra gli *s-bot*, ma è stata studiata anche la possibilità di integrare tale comunicazione con quella sonora [14]. Nella situazione in cui bisogna evitare una buca (*hole avoidance*), l'uso della comunicazione diretta (segnale acustico) tra gli *s-bot*, come rinforzo della interazione diretta (collegamento fisico), si è dimostrato particolarmente benefico poichè gli *s-bot* che compongono lo *Swarm-bot* reagiscono più in fretta all'individuazione di una buca se ricevono un segnale acustico piuttosto che basandosi esclusivamente sulla percezione di una trazione forte (scatenante il comportamento reattivo).

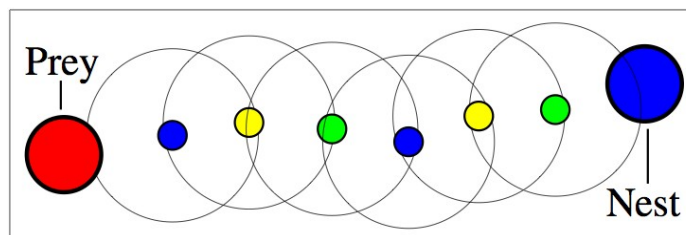
Naturalmente l'utilizzo del sistema sonoro comporta pro, ma anche contro. Infatti nello studio condotto su un sistema *Swarm-bot* composto da due *s-bot* [15] è stato osservato che se il segnale acustico viene inviato come prodotto di una decisione errata allora l'intero sistema *Swarm-bot* può collassare. L'importanza e l'effetto di tale evento ovviamente dipendono dalla dimensione del sistema *Swarm-bot*. L'inserimento di questa forma aggiuntiva di comunicazione richiede la creazione di *controller* più complessi, quindi bisogna bilanciare bene i costi e i benefici di tale comunicazione prima di considerarlo come un percorso che può portare alla soluzione di un dato compito.

3.4 Task allocation

Quando il sistema *Swarm-bot* diventa troppo complesso, cioè quando gli *s-bot* facenti parte del sistema diventano troppo numerosi, nasce la necessità di assegnare dei compiti agli elementi del sistema. Infatti se tutti gli *s-bot* svolgono lo stesso compito allora si assiste ad un degrado delle prestazioni dovuto principalmente all'intralcio che diventano gli uni per gli altri. Dato l'esempio di molti *s-bot* che portano ognuno un oggetto in una zona-obiettivo, è facile capire che se il loro numero aumenta allora aumenta anche il tempo che gli *s-bot* impiegano ad evitarsi (uno *s-bot* evita sempre il contatto con gli altri *s-bot* a meno che non voglia agganciarsi con loro), e questo porta ad un degrado delle prestazioni direttamente proporzionale all'aumentare degli elementi del sistema. L'assegnamento dei compiti (*task allocation*) è un meccanismo che permette di determinare automaticamente il quantitativo ottimo di *s-bot* per una determinato compito. Il metodo utilizzato non usa alcuna forma di comunicazione diretta o simbolica e non richiede alcun intervento umano. Si è fatto riferimento al modello di Deneubourg (con qualche modifica) secondo il quale ogni *s-bot*, per compiere un compito, lascia il "nido" con una certa probabilità P . Se lo *s-bot* ha successo con tale attività allora la probabilità P si incrementa di un certo valore x , altrimenti si decrementa di x . Quindi l'assegnamento avviene durante lo svolgimento delle "vita" dello *s-bot* in modo del tutto auto-organizzato [16].

4 Foraging scenario

Un esempio di comportamento complesso di uno *Swarm-bot* può essere trovato nel *foraging scenario* (approvvigionamento). Questa è la situazione in cui bisogna trovare un oggetto-preda e un oggetto-nido, stabilire un cammino tra i due e trasportare la preda al nido. Una volta stabilito il percorso, vengono reclutati gli *s-bot* per il trasporto del pesante oggetto-preda, questi si auto-assemblano in una struttura e insieme effettuano il trasporto seguendo percorso verso il "nido". Gli *s-bot* si auto-organizzano in gruppi per svolgere parallelamente diversi compiti (*task allocation*). Per la formazione del percorso nido-preda si utilizza il *Cyclic Directional Patterns*.



Le catene CDP sono un nuovo metodo nell'esplorazione robotica in ambienti sconosciuti. Queste catene servono per

- i) esplorare l'ambiente,
- ii) stabilire un percorso tra il nido e la preda,
- iii) reclutare lavoratori per trasportare la preda lungo tale percorso
- iv) guidare il gruppo di trasporto verso il nido.

Il *controller* degli *s-bot* in questo caso è formato da due moduli:

- modulo esplorazione (*exploration module*)
- modulo trasporto (*transport module*).

Nel modulo esplorazione gli *s-bot*, inizialmente collocati in posizioni casuali, devono cercare il nido e una volta trovato cominciano ad auto-organizzarsi in una catena collegata visualmente. Ogni *s-bot* facente parte della catena emette un luce (tramite l'anello di LED) il cui colore (blu, verde o giallo) dipende dalla posizione nella catena. I membri di una catena in alcune condizioni possono lasciare la loro posizione. Questo comportamento è fondamentale per l'esplorazione dell'ambiente poiché permette la formazione di catene in area non ancora esplorate. Il processo di formazione e decomposizione continua finché non si incontra la preda. In questo caso la catena non si decompone più e automaticamente questo porta altri *s-bot* alla preda. I comportamenti previsti da questo modulo sono tre:

- *search* : gli *s-bot* effettuano dei cammini casuali nell'ambiente evitando gli ostacoli (e gli *s-bot*). Nessun LED è attivato;
- *explore* : un esploratore si muove lungo la catena verso la sua coda. Nel caso in cui uno *s-bot* diventa un esploratore mentre lascia una catena, allora questo si muove verso il nido dove può decidere di seguire un'altra catena. Nessun LED è attivato;
- *chain* : un membro della catena attiva il proprio LED nel colore appropriato. Per migliorare la catena è stato implementato un comportamento tale per cui se, presi tre elementi quello

centrale ha un angolo inferiore ai 120° rispetto agli altri due allora questo si sposta allinea a questi.

Il passaggio da un comportamento ad un altro può avvenire se:

- *search* -> *explore* : viene percepito un membro di una catena
- *explore* -> *search* : non si percepisce nessun membro di una catena
- *explore* -> *chain* : uno *s-bot* raggiunge la coda della catena allora lo *s-bot* si unisce alla catena (con probabilità P_{ec}) oppure la preda è individuata ad una distanza superiore a 30cm
- *explore* -> *assemble* : si individua una preda ad una distanza inferiore ai 30 cm
- *chain* -> *search* : uno *s-bot* non percepisce più la presenza dell'elemento precedente della catena
- *chain* -> *explore* : un membro è situato in coda alla catena, questo lascia la catena con probabilità P_{ce} .

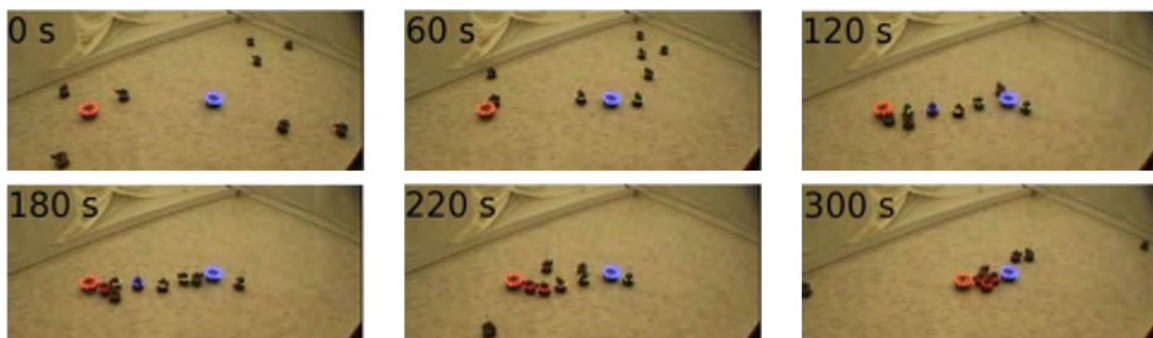
In base ai valori di P_{ec} e P_{ce} si influenza il numero e la lunghezza delle catene che si formano e anche la velocità del processo che porta alla loro formazione e disgregazione. I valori migliori trovati sono $P_{ec} = 0,14$ e $P_{ce} = 0,007$.

Il modulo trasporto controlla gli *s-bot* in modo da formare una struttura per trascinare la preda fino al nido lungo la catena formata da altri *s-bot*. I comportamenti previsti sono:

- *assemble* : lo *s-bot* si avvicina e si connette con l'oggetto-preda (illuminato con LED rossi). Una volta connesso anche lui attiva l'anello di LED col colore rosso e in questo modo viene visto anche lui come parte della preda da parte di altri *s-bot*;
- *transport-target* : lo *s-bot* allinea il proprio *chassis* verso il membro della catena più vicina, cioè si rivolge verso il nido e comincia a tirare/spingere
- *transport-blind* : non viene percepito nessun membro di alcuna catena, lo *s-bot* controlla le forze tra il sistema di trazione e la torretta. In base a queste informazioni capisce la direzione e la velocità desiderata.

Le transizioni tra i vari stati sono:

- *assemble* -> *transport-target* : se uno *s-bot* riesce con successo a collegarsi alla preda o ad un altro *s-bot* già collegato alla preda
- *transport-target* -> *transport-blind* : se lo *s-bot* non percepisce nessun elemento di una catena
- *transport-blind* -> *transport-target* : se lo *s-bot* percepisce un elemento di una catena.



Questo sistema è stato sviluppato per il controllo degli *Swarm-bot* tramite un algoritmo totalmente decentralizzato e omogeneo nel controllo. Esperimenti hanno dimostrato l'alta affidabilità e robustezza di tale algoritmo [17].

5 Swarmanoid

Il progetto Swarmanoid (<http://www.swarmanoid.org/>) è il successore del progetto Swarm-bot ed è stato costruito sui risultati ottenuti da tale progetto. Il principale obiettivo scientifico di ricerca del progetto Swarmanoid è quello di progettare, implementare e controllare un nuovo sistema robotico distribuito. Tale sistema sarà costituito da robot eterogenei, dinamicamente connessi, piccoli e autonomi. Collettivamente questi robot formeranno quello che viene chiamato *swarmanoid*. Lo *swarmanoid* che desiderano costruire sarà composto da circa 60 robot autonomi di tre tipi:

- *eye-bot* : specializzato nel sentire e analizzare l'ambiente da una posizione elevata che consente una visione che gli *hand-bot* e i *foot-bot* non possono avere;
- *hand-bot* : specializzato nel muoversi e agire nello spazio compreso tra quello coperto dai *foot-bot* (terra) e quello degli *eye-bot* (massima altezza). Può scalare superfici verticali di muri o di oggetti presenti nell'ambiente;
- *foot-bot* : specializzati nel muoversi sui terreni insidiosi e trasportare oggetti o altri robot.

Sono basati sulla piattaforma implementata nel progetto *Swarm-bot*.

La combinazione di questi tre tipi di agenti autonomi forma un sistema robotico eterogeneo che è capace di muoversi in uno spazio tridimensionale.

Bibliografia

1. G. Beni and J. Wang. '*Swarm intelligence*'. In *Proceedings of the Seventh Annual Meeting of the Robotics Society of Japan*, pages 425–428, Tokio, Japan, 1989. RSJ press.
2. E. Bonabeau, M. Dorigo, and G. Theraulaz. '*Swarm Intelligence: From Natural to Artificial Systems*'. Oxford University Press, New York, NY, 1999.
3. M. Dorigo , V. Trianni, E. Sahin, R. Groß, T. Labella, G. Baldassarre, S. Nolfi, J.-L. Deneubourg, F. Mondada, D. Floreano, and L. M. Gambardella. '*Evolving Self-Organizing Behaviors for a Swarm-bot*'. In *Autonomous Robots*, 17(2-3):223-245, 2004
4. F. Mondada , M. Bonani , S. Magnenat , A. Guignard, and D. Floreano. '*Physical connections and cooperation in swarm robotics*'. In Frans Groen, Nancy Amato, Andrea Bonarini, Eiichi Yoshida and Ben Kröse editors, *Proceedings of the 8th Conference on Intelligent Autonomous Systems (IAS8)*, Amsterdam, NL, March 10-14, 2004, pages 53-60. IOS Press.
5. F. Mondada , G. C. Pettinaro, A. Guignard, I. W. Kwee, J. -L. Deneubourg, S. Nolfi, L. M. Gambardella, D. Floreano, and M. Dorigo. '*Swarm-Bot: a New Distributed Robotic Concept*'. In *Autonomous Robots*, 17(2-3):193-221, 2004.
6. F. Mondada, M. Bonani, A. Guignard, S. Magnenat, C. Studer, and D. Floreano. '*Superlinear Physical Performances in a SWARM-BOT*'. Proceedings of the VIIIth European Conference on Artificial Life, Lecture Notes in *Artificial Intelligence* LNAI3630, pp 282-291, Springer Verlag.
7. M. Dorigo, V. Trianni, E. Sahin, R. Groß, T. Labella, G. Baldassarre, S. Nolfi, J.-L. Deneubourg, F. Mondada, D. Floreano, and L. M. Gambardella. '*Evolving Self-Organizing Behaviors for a Swarm-bot*'. In *Autonomous Robots*, 17(2-3):223-245, 2004
8. R. Groß, M. Bonani, F. Mondada, and M. Dorigo. '*Autonomous Self-Assembly in Swarm-Bots*'. In *IEEE Transactions on Robotics*, volume 22(6), 1115-1130, 2006
9. E. Tuci, R. Groß, V. Trianni, F. Mondada, M. Bonani, and M. Dorigo. '*Cooperation Through Self-Assembly in Multi-Robot Systems*'. In *ACM TAAS*, Volume 1(2), 115-150, December 2006
10. R. O'Grady, R. Groß, A. L. Christensen, F. Mondada, M. Bonani, M. Dorigo, and J.-L. Deneubourg. '*Performance Benefits of Self-Assembly in a Swarm-Bot*'.
11. R. Groß and M. Dorigo. '*Towards group transport by swarms of robots*'.
12. R. O'Grady, R. Groß, A. L. Christensen, and M. Dorigo. '*Self-assembly strategies in a group of autonomous mobile robots*'.
13. R. Groß and M. Dorigo. '*Group Transport of an Object to a Target That Only Some Group Members May Sense*'. In Yao X., Burke E., Lozano J. A., Smith J., Merelo-Guervós J. J., Bullinaria J. A., Rowe J., Tino P., Kabán A., and Schwefel H.-P., editors, *Parallel Problem Solving from Nature - 8th International Conference, PPSN VIII*, volume 3242 of Lecture Notes in Computer Science, pages 852-861. Springer Verlag, Berlin, Germany, 2004.
14. V. Trianni, T. H. Labella, and M. Dorigo. '*Evolution of Direct Communication for a Swarm-bot Performing Hole Avoidance*'. In M. Dorigo, M. Birattari, C. Blum, L. M. Gambardella, F. Mondada, and T. Stutzle, editors, *Ant Colony Optimization and Swarm Intelligence - Proceedings of ANTS2004 (Fourth International Workshop)*, volume 3172 of Lecture Notes

- in Computer Science, pages 131-142. [Springer Verlag](#), Berlin, Germany, 2004.
15. C. Ampatzis, E. Tuci, V. Trianni, M. Dorigo. '*Evolution of Signalling in a Group of Robots Controlled by Dynamic Neural Networks*'. In E. Sahin et al., Proceedings of the Second Workshop on Swarm Robotics, In press.
 16. T. H. Labella, M. Dorigo, J.-L. Deneubourg. '*Self-Organised Task Allocation in a Group of Robots*'. In R. Alami, editor, *Proceedings of the 7th International Symposium on Distributed Autonomous Robotic Systems (DARS04)*. Toulouse, France, June 23-25, 2004.
 17. S. Nouyan, R. Groß, M. Dorigo, M. Bonani, and F. Mondada. '*Group Transport Along a Robot Chain in a Self-Organised Robot Colony*'. In *Proc. of the 9th Int. Conf. on Intelligent Autonomous Systems*, pp. 433--442, IOS Press, Amsterdam, The Netherlands.