

IL FUORIGIOCO A QUATTRO IN UNA SQUADRA DI CALCIO



NOTES

Progetto di Sistemi Intelligenti

Distribuiti

Docente: Andrea Omicini

A CURA DI STEFANO SEVERI

Indice

Indice	3
1 IL GIOCO	5
1.1 MAIN GOAL	5
1.2 IL FUORIGIOCO	5
1.3 SCENARIO	6
1.4 DISPOSIZIONE DEI GIOCATORI	7
1.5 L'ARBITRO	7
1.6 RISOLVERE IL MAIN GOAL	8
1.7 SECONDARY GOAL	9
2 IL SOFTWARE	13
2.1 IL MAIN	13
2.2 GLI AGENTI	14
2.2.1 STRUTTURA INTERNA DELL'AGENTE	14
2.3 IL CONTESTO	15
2.4 L'INTELLIGENZA	16
2.5 SIMULAZIONE GRAFICA	17
2.6 SVILUPPI FUTURI	17

Capitolo 1

IL GIOCO

1.1 MAIN GOAL

Generare un software ad agenti in grado di simulare il comportamento ottimale di una difesa a quattro di una squadra di calcio in fase di applicazione del fuorigioco. Per comportamento ottimale si intende la messa in fuorigioco dell'avversario quando possibile e la non applicazione dello schema quando troppo rischioso o impossibile. In seguito verranno illustrate entrambe le situazioni.

1.2 IL FUORIGIOCO

REGOLA 11 - FUORI GIUOCO

Posizione di fuori giuoco

Essere in posizione di fuori giuoco non è di per sè un'infrazione. Un calciatore si trova in posizione di fuori giuoco quando:

- è più vicino alla linea di porta avversaria sia rispetto al pallone sia al penultimo avversario.

Un calciatore non si trova in posizione di fuori giuoco quando:

- si trova nella propria metà del terreno di giuoco *oppure*
-

- si trova in linea con il penultimo avversario *oppure*
- si trova in linea con i due ultimi avversari

Infrazione

La posizione di fuori giuoco di un calciatore deve essere punita solo se, nel momento in cui il pallone è toccato o giuocato da uno dei suoi compagni, il calciatore, a giudizio dell'arbitro, prende parte attiva al giuoco:

- intervenendo nel giuoco *oppure*
- influenzando un avversario *oppure*
- traendo vantaggio da tale posizione

Non infrazione

Non vi è infrazione di fuori giuoco quando un calciatore riceve direttamente il pallone:

- su calcio di rinvio *oppure*
- su rimessa dalla linea laterale *oppure*
- su calcio d'angolo

Infrazioni e sanzioni

Per tutte le infrazioni alla regola del fuori giuoco, l'arbitro accorda alla squadra avversaria un calcio di punizione indiretto, che deve essere eseguito nel punto in cui l'infrazione è stata commessa.

1.3 SCENARIO

Il software considera solo la metà campo difensiva della squadra che vuole applicare il fuorigioco, unica zona del campo in cui l'offside è possibile. Per default il portiere è sempre considerato in porta e non prende mai parte alla simulazione. E' simulato il comportamento di 4 difensori e di 2 attaccanti avversari. Uno dei due attaccanti al via della simulazione entra

in possesso di palla, per cui lo schema da applicare in difesa deve eventualmente prevedere la messa in fuorigioco del solo secondo attaccante, quello senza palla. Non è contemplata la presenza di altri giocatori nelle zone interessate dallo svolgimento dell'azione.

1.4 DISPOSIZIONE DEI GIOCATORI

I quattro difensori partono disposti secondo il classico schema della difesa a quattro, occupando cioè ognuno la propria zona del campo, con l'unica eccezione che non sono in linea. Sulla fascia destra parte il numero 2 Marco Ambrogioni; centrale di destra il numero 6 Daniele Ficagna e centrale sinistro il numero 5 Riccardo Bocchini; sulla sinistra il numero 13 Angelo Rea. Sono definiti quattro rettangoli virtuali, non sovrapposti, all'interno dei quali possono trovarsi i difensori al momento dell'inizio della simulazione. Ciò garantisce che ogni zona del campo sia presidiata, ma non che un difensore si trovi in linea con gli altri. Schema simile per i due attaccanti, con la differenza che il numero 10 Simone Cavalli parte sempre senza la palla ed il suo "rettangolo" di partenza è in parte più avanzato di quello del 9, Marco Bernacci, attaccante con la palla. Anche in questo caso si vuole garantire, per un simulazione più reale, che l'attaccante senza palla il più delle volte parta davanti a quello con la palla (che in teoria lo dovrebbe lanciare), senza però escludere l'ipotesi inversa (incursione dell'attaccante con la palla). La tecnica dei rettangoli di partenza è riassunta dalla figura 1.

1.5 L'ARBITRO

E' quello a cui spetta sanzionare il fuorigioco. Ovviamente non può essere un agente giocatore a chiamare il fuorigioco, poiché sarebbe "di parte". Tale compito è quindi affidato al contesto, che si limiterà ad osservare le posizioni degli agenti giocatori in campo ed a fermare il gioco nei casi di offside. Il contesto interromperà la simulazione anche nei casi in cui la

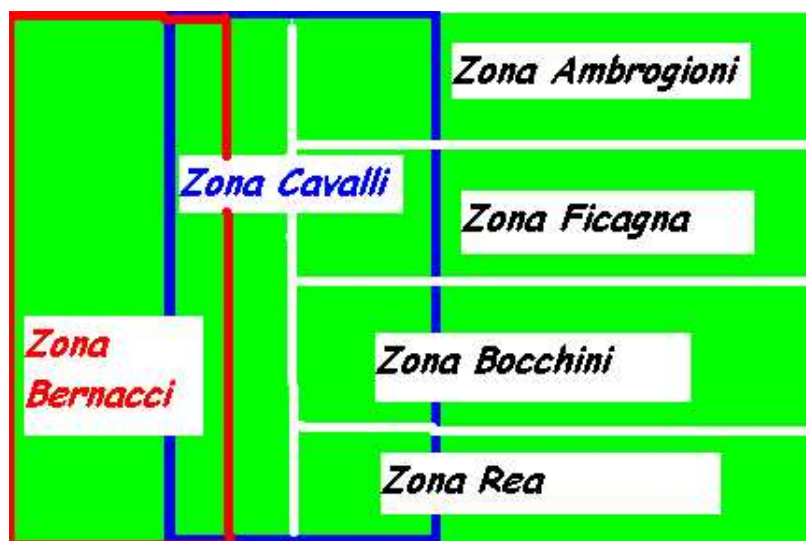


Figura 1.1: Schema dei rettangoli di gioco

difesa decida non sia opportuno applicare il fuorigioco. In questo caso il suo comito sarà quello di intercettare il messaggio della difesa, senza valutarne la fondatezza, ed interrompere la simulazione. Non è compito di questo progetto gestire anche il contrasto tra giocatori.

1.6 RISOLVERE IL MAIN GOAL

Vitale per una squadra di calcio a zona è non tanto l'applicare il più possibile la tattica del fuorigioco, quanto il capire se sia conveniente farlo o meno. Se non sussistono le condizioni, è più sicuro andare a contrasto per recuperare palla o permettere il rientro dei propri compagni. Per questo la difesa deve essere in grado di osservare e soprattutto capire ciò che sta succedendo in campo e distinguere tutti i possibili sottocasi, quindi risolvere i goal secondari ad essi connessi. Queste le situazioni possibili:

1. L'attaccante senza palla si trova in partenza già davanti a tutti i 4 difensori.
2. L'attaccante con la palla è davanti a quello senza palla.

3. L'attaccante con la palla e quello senza sono troppo vicini, anche se quello con la palla è più indietro. Il fuorigioco si potrebbe applicare ma sarebbe troppo rischioso: anche l'attaccante con la palla potrebbe passare la linea dell'offside, trovandosi di fatto solo davanti al portiere.
4. Attaccante senza palla dietro a tutti e 4 i difensori e sufficientemente davanti a quello senza palla.
5. Attaccante senza palla davanti a quello con la palla ma dietro a soli 3 difensori.
6. Attaccante senza palla davanti a quello con la palla ma dietro a soli 2 difensori.
7. Attaccante senza palla davanti a quello con la palla ma dietro ad un solo difensore

1.7 SECONDARY GOAL

In tutti e 7 i casi possibili gli agenti difensori hanno dei goal secondari che concorrono però sempre alla risoluzione del goal principale. Eccoli nel dettaglio, con riferimento alla numerazione dei casi precedenti:

1. E' compito dei difensori comprendere la mancanza di necessità di alcuna manovra e fermarsi aspettando che l'arbitro (in questo caso il contesto) fischi il fuorigioco.
 2. Visto il primo comma dell'articolo 11 del regolamento del gioco del calcio, secondo il quale "Un calciatore si trova in posizione di fuori giuoco quando è più vicino alla linea di porta avversaria sia rispetto al pallone sia al penultimo avversario", è impossibile applicare il fuorigioco. I difensori devono quindi prepararsi ad andare a contrasto. Sarà cura del contesto a questo punto interrompere la simulazione al di là della correttezza della scelta effettuata dai difensori. Una
-

verifica finale potrà essere effettuata dalla persona che lancia la simulazione sulla base del grafico finale. Si ricorda che la simulazione si limita ad analizzare il solo aspetto del fuorigioco e non l'effettiva applicazione dello schema contrasto.

3. Gli agenti difensori devono comprendere la pericolosità della situazione e non applicare il fuorigioco ma andare a contrasto. Ancora una volta il contesto imporrà la fine della simulazione senza valutare la correttezza della scelta effettuata dai difensori, ed ogni altro giudizio è rimandato alla persona che lancia il programma sulla base del grafico finale.
 4. Gli agenti difensori devono prima mettersi in linea e poi salire contemporaneamente. E' importante la composizione della linea di fuorigioco. I difensori più avanzati prima di salire devono naturalmente attendere che tutti quelli dietro siano arrivati in linea: cioè rispecchia con buona approssimazione il caso reale, anche con gli svantaggi connessi. Se un agente difensore sfortunatamente si trova molto più indietro rispetto ai propri compagni, l'applicazione del fuorigioco risulterà meno efficace, poichè nel frattempo gli attaccanti avanzano e possono interporsi tra due difensori. E' compito dei difensori monitorare l'evoluzione della situazione ed eventualmente, se l'attaccante ha passato uno o più difensori prima della composizione della linea a quattro, applicare il fuorigioco con i difensori rimasti, in accordo ai casi 5), 6) e 7). Proprio come nella realtà la cattiva posizione di partenza di un difensore è elemento di svantaggio per la difesa. Si sottolinea nuovamente che scopo della simulazione non è tanto evitare che questo caso di verifici, proprio perchè resta possibile nella realtà: ci si limita a reagire nel migliore dei modi ad una situazione potenzialmente sfavorevole. Altrimenti una volta composta la linea a 4 quattro i difensori saliranno contemporaneamente fino alla messa in fuorigioco dell'attaccante.
 5. Il caso è identico al precedente, con l'eccezione che la linea deve essere composta dai 3 agenti difensori più arretrati. Ancora una vol-
-

ta durante il corso della simulazione, nelle situazioni più sfortunate di uno o più difensori eccessivamente arretrati, da questo caso si può passare ai casi 6) e 7). Compito degli agenti è comprendere la situazione e comportarsi di conseguenza.

6. Linea di fuorigioco composta da soli due agenti. Quello più avanzato deve attendere l'allineamento di quello più arretrato. Se nel frattempo tra i due si intromettesse l'attaccante senza palla, si passerebbe al caso 7).
 7. Situazione più rischiosa, perchè un solo difensore tiene in gioco l'attaccante senza palla avversario. Suo compito è quello di salire più velocemente possibile prima che l'avversario possa ricevere la palla e dar vita ad un "uno contro uno" da brividi.
-

Capitolo 2

IL SOFTWARE

Il software è composto essenzialmente da quattro parti. La **struttura portante** è in Java, linguaggio di programmazione scelto anche per la realizzazione degli agenti. Anche la **parte grafica**, che deve mostrare i risultati della simulazione, è completamente in Java: in entrambi i casi il linguaggio Java risultava sia sufficientemente user-friendly che adatto per una programmazione strutturata tra più parti. Anche per quel che riguarda la grafica Java è semplice ed efficace. Lo **spazio di comunicazione degli agenti** e di tutto il programma è un centro di tuple realizzato con tecnologia TuCSoN. L'**intelligenza degli agenti** è stata infine realizzata Prolog, linguaggio logico che meglio di Java si prestava a tal fine. In particolare si è scelto di usare tecnologia TuProlog.

2.1 IL MAIN

Il file

```
FabrizioCastori.class
```

contiene il main del programma. Suoi compiti sono quelli di settare le specifiche del contesto descritte in un apposito file "offside2.rsp" scritto con `Respect`; inizializzare un nuovo Tuple Centre chiamato "maracana" (si legge *Maracanà, nda*) utilizzando il contesto di default nel quale immettere le tuple di inizializzazione quindi lanciare gli agenti. Infine scriverà a video sia posizione di inizio che di arrivo dei vari agenti gio-

catori e chiamerà un metodo del grafico al quale passare questi parametri. Tuple di inizializzazione sono `fine(0)`, vale a dire la bandiera del guardialinee che all'inizio della simulazione deve essere per forza abbassata, più altre tre tuple che servono per le barriere di sincronizzazione. `barrier_size(6)` specifica il numero di agenti da attendere prima di sorpassare la barriera, `ready_agents(0)`, e `ready_agents2(0)` settano a zero il numero di agenti inizialmente pronto a passare le 2 barriere di sincronizzazione. I quattro agenti difensori lanciati sono, da destra a sinistra, Ambrogioni, Ficagna, Bocchini e Rea: i due attaccanti Bernacci (quello con la palla) e Cavalli.

2.2 GLI AGENTI

Tutti e 6 gli agenti giocatori ereditano la classe

```
alice.tucson.api.Agent.
```

Ogni agente giocatore ha un proprio codice, anche se sostanzialmente in questa simulazione si è deciso di usare giocatori con le stesse caratteristiche, differenziati al loro interno solo per i nomi. E' comunque bene pensare che in realtà non esistono giocatori tra loro "cloni", per cui differenti programmatori in un contesto di simulazione più ampio potrebbero realizzare anche giocatori della stessa squadra in maniera differente, rispettando semplicemente i protocolli di comunicazione. Nel caso in esame la differenza sostanziale tra agenti attaccanti e difensori sta nell'intelligenza, codificata in due file TuProlog di Prolog: "attaccanti.pl" per gli attaccanti e "Finale.pl" per i difensori. Ancora una volta si ricorda che non è necessario che i giocatori abbiano lo stesso modo di ragionare, perchè così non avviene nella realtà. L'importante è che il risultato del loro ragionamento sia conforme a quello atteso.

2.2.1 STRUTTURA INTERNA DELL'AGENTE

L'agente per prima cosa si collega al Tuple Centre "maracana" e decide la propria posizione in maniera pseudo-random, entro i limiti posti dal

sistema dei rettangoli precedentemente illustrato. Decisa la propria posizione la si comunica al centro di tuple e in sequenza si rileva lo stato della tupla `fine(X)` per vedere quanto vale: se $X=0$, cioè la bandierina del fuorigioco è bassa, si entra in un ciclo `for` che sarà iterato finché X vale 0: in caso contrario si passano le proprie coordinate al main.

Una volta dentro il ciclo `for` l'agente passa a leggere la propria posizione e quella di tutti gli altri giocatori, quindi incontra la prima barriera di sincronizzazione. Quando ha letto tutte le posizioni lancia una tupla `ready()` e si mette in attesa di leggere `ready_all()`, il messaggio che tutti sono pronti. `ready_all()` è generata dal contesto quando tutti gli agenti hanno completato la lettura: sempre il contesto provvede a togliere dal centro di tuple le posizioni vecchie degli agenti.

A questo punto si passano tutte le posizioni al motore Prolog sotto forma di stringa di testo e si recupera, in base alle decisioni del motore Prolog, la propria nuova posizione. Segue una ulteriore `out` della propria nuova posizione prima di arrivare alla seconda barriera di sincronizzazione. Il contesto impone che tutti gli agenti comunichino la loro posizione aggiornata quindi valuta se ci sono gli estremi per il fuorigioco o per sospendere la simulazione, oppure naturalmente per continuarla. La decisione dell'arbitro è comunicata con la tupla `fine(X)` che gli agenti potranno leggere solo dopo il via libera dato dal contesto con una tupla `ready_all2()`.

Se il flag $X=0$ si ripete il ciclo `for`, in caso contrario si esce e si comunica la propria posizione al main, quindi l'agente si disconnette.

Il flag X può valere 1 se il contesto rileva il fuorigioco, mentre assume valore 2 se i quattro giocatori hanno deciso di andare a contrasto.

2.3 IL CONTESTO

Per rilevare il fuorigioco il contesto deve rilevare che il giocatore senza palla sia dietro a tutti e 4 i difensori ma soprattutto che sia più avanzato del compagno di squadra con la palla. Lo rileva ad ogni fine della seconda barriera di sincronizzazione, ovvero ogni qualvolta gli agenti giocatori hanno aggiornato la loro posizione.

Nella realtà, come spiega il regolamento, "Essere in posizione di fuori giuoco non è di per sé un'infrazione", quindi la bandiera del fuorigioco non scatterebbe automaticamente. Ancora una volta scopo della simulazione non è creare un agente arbitro intelligente, ma assicurarsi che i difensori si comportino in maniera ottimale.

Se gli agenti difensori capiscono che devono andare a contrasto, lo devono comunicare al contesto perchè sospenda la simulazione. Come mezzo per simulare questa comunicazione, si è scelto di portare tutti e 4 i difensori, nel caso in cui decidano che si debba andare a contrasto, in una posizione speciale, più arretrata dei rettangoli di partenza, non raggiungibile altrimenti. Quando il contesto nota la posizione dei 4 giocatori "sulla linea di fondo", la interpreta come segnale per interrompere la simulazione.

Questa posizione speciale non deve essere considerata come reale, ma come semplice artificio tecnico del programmatore. L'obiettivo in caso di contrasto non è quello di muoversi secondo una determinata logica, ma semplicemente di rilevare l'impossibilità di applicare il fuorigioco.

2.4 L'INTELLIGENZA

L'intelligenza degli attaccanti è fittizia, nel senso che qualunque cosa succeda, loro avanzano. I difensori, agenti sui quali è riposta l'attenzione, devono poter discernere i 7 casi l'uno dall'altro e afferrare eventuali variazioni della situazione, cioè passare da un caso all'altro. Capacità richieste all'intelligenza dei difensori sono soprattutto quella di saper ordinare le posizioni di tutti e sei i giocatori in campo e di decidere eventualmente di quanto salire.

Il "passo" massimo possibile di tutti i giocatori, cioè l'avanzamento lungo l'asse longitudinale, è stato fissato per comodità a 30 pixel sia per gli attaccanti che per i difensori. Naturalmente introducendo il concetto di giocatore più forte, questo parametro può essere variato.

La soglia di pericolosità, cioè la distanza minima tra attaccante con la palla e attaccante senza palla oltre la quale è troppo rischioso applicare il fuorigioco, è stata fissata a 60 pixel, vale a dire 2 passi degli attaccanti. Si

tratta però di un parametro di giudizio che l'allenatore dovrebbe trasmettere ai suoi difensori in maniera omogenea, a seconda della pericolosità degli attaccanti avversari.

Nel corso di questa simulazione sono stati considerati solo gli spostamenti sull'asse longitudinale. Naturalmente agendo sull'intelligenza, si potrebbe facilmente fare in modo che i giocatori si spostino anche sull'altro asse. In particolare la somma dei moduli quadrati degli spostamenti lungo i due assi non dovrebbe mai essere maggiore di una costante che rappresenta il quadrato della velocità massima del giocatore stesso.

2.5 SIMULAZIONE GRAFICA

Su di un pannello `CesenaPanel` che estende il `JPanel` e che raffigura la metà di un campo di calcio, sono mostrate le posizioni di partenza e di arrivo, con relativi spostamenti, dei 6 agenti giocatori.

2.6 SVILUPPI FUTURI

Primo sviluppo futuro potrebbe essere implementare il passaggio di palla tra i due attaccanti. Quindi la simulazione potrebbe costituire la base per una sfida virtuale tra due squadre di calcio. Nel contesto andrebbero inserite le regole del gioco del calcio proprio come in questo esempio è stata inserita quella del fuorigioco. Una volta decise le regole di comunicazione tra i vari giocatori ed un tuple centre pubblico in cui inserire ed aggiornare la propria posizione, il resto sarebbe a carico dei programmatori delle varie squadre o addirittura dei vari giocatori. Proprio come per l'allenatore nella realtà, all'interno dello spazio virtuale definito dalle regole del gioco, il programmatore potrebbe valutare e decidere le migliori tattiche di gioco della propria formazione.
