

Andrea Di Ferdinando, Simone Cutini, Domenico Parisi, Marco Zorzi

SYNAPTHINK: CREARE SCENARI ADATTIVI PER I VIDEOGIOCHI

1. L'INTELLIGENZA (ARTIFICIALE) DELLE MACCHINE

Le macchine fanno sempre più parte della nostra vita quotidiana. Oggi, sarebbe difficile per ciascuno di noi immaginare un mondo senza più televisioni, automobili, cellulari, computer. Per quasi tutte le nostre attività quotidiane, infatti, sia nel lavoro che nel tempo libero, facciamo uso di artefatti a cui poter delegare i compiti più faticosi, ripetitivi o impossibili da realizzare con le nostre forze. Tuttavia, nonostante le macchine siano diventate oramai indispensabili per la nostra vita, non è ben chiaro se esse possano essere considerate intelligenti. Si sente parlare di meccanismi intelligenti, tecnologie intelligenti, sistemi intelligenti, persino bombe intelligenti, ma si tratta davvero della stessa forma di intelligenza che attribuiamo ad una persona, ad un bambino, o ad un semplice pulcino? In altre parole: le macchine sono davvero intelligenti?

Evitando di addentrarci in questo spinoso argomento, che da Cartesio in avanti ha interessato e continua ad interessare studiosi delle più varie discipline, quella che vorremmo affrontare in questo capitolo non è la questione filosofica se le macchine pensino o possano pensare come gli esseri umani, se possiedano o meno una coscienza, ma la questione più concreta di cosa manchi alle macchine attuali per poter mostrare quei comportamenti che definiamo intelligenti. Probabilmente non si tratta di un'unica caratteristica, ma di un insieme di proprietà che attualmente le macchine non hanno, e che invece costituiscono una componente essenziale dell'intelligenza biologica. In particolare, nel presente lavoro metteremo in evidenza due di queste proprietà, che più di altre sembrano contraddistinguere l'intelligenza umana e più in generale quella animale dall'intelligenza di tipo artificiale, ossia quella delle macchine (attuali).

Innanzitutto, gli organismi biologici mostrano una notevole capacità di adattarsi, cosa che invece le macchine non sembrano in grado di fare adeguatamente. Solitamente, infatti, le macchine vengono progettate per determinati scopi ed ambienti, dimostrandosi insofferenti ad ogni cambiamento. Ad esempio, un programma progettato per Windows non sarà in grado di funzionare su Linux, o addirittura un programma che gira su Windows XP comporterà dei problemi quando utilizzato su Windows 2000, o ancora peggio un programma potrebbe funzionare su una specifica macchina ed invece avere dei problemi su un'altra a causa di qualche incompatibilità a livello hardware. Questi problemi non riguardano solamente i

programmi per computer, ma più in generale tutti gli artefatti con cui abbiamo quotidianamente a che fare, dal cellulare alla TV, dall'automobile al frigorifero. Gli organismi biologici sono invece più flessibili e capaci spesso di adattarsi, come dimostra ad esempio il fatto che ci siano esseri umani in pressoché tutti gli angoli del globo, dalle fredde regioni artiche a quelle calde dell'equatore. La differenza tra esseri umani e macchine appare subito evidente anche quando proviamo a farci una chiacchierata con uno dei numerosi chatterbot che popolano la rete (il capostipite di tali programmi è sicuramente Eliza; si veda Weizenbaum, 1965). Tali programmi si dicono in grado di dialogare con i visitatori del sito, ma tutto quello che riescono ad offrire è una conversazione al limite del delirio in cui ci si ritrova con una disarmante sensazione di impotenza. Questo perché anche i sistemi più sofisticati rivelano una sconcertante rigidità di fondo: essendo il programma incapace di adattarsi all'interlocutore e all'andamento della conversazione, ci si trova a doversi adattare noi agli schemi del programma.

In sintesi, la rigidità delle macchine attuali ci porta spesso a parlare di "artificial stupidity" piuttosto che di "artificial intelligence". Come ad esempio nella figura seguente, che mostra chiaramente l'incapacità delle macchine di prendere in considerazione il contesto in cui si trovano ad operare, seguendo esclusivamente le regole con le quali sono state costruite.

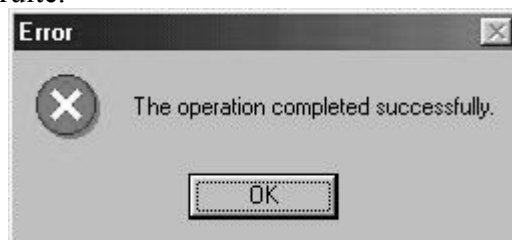


Figura 1. Un esempio di "artificial stupidity". I due messaggi sono chiaramente incompatibili, ma il computer non è in grado di capirlo.

Conseguenza di questa rigidità è una seconda caratteristica che contraddistingue le macchine attuali, ossia la loro incapacità di affrontare situazioni e compiti sconosciuti, cosa che invece gli organismi biologici riescono spesso a gestire con successo. Come non ricordare a questo riguardo quello che è successo con la fine del millennio e con il pericolo del Millenium Bug? Data l'incapacità dei computer e di altri dispositivi di gestire situazioni inaspettate, ci si è trovati a modificare una quantità enorme di programmi (tutti quelli che utilizzavano solo due cifre per indicare l'anno) con massicci investimenti di denaro, di risorse e di tempo, approntando veri e propri piani di emergenza.

Secondo la nostra analisi, dunque, quello che manca alle macchine attuali per mostrare comportamenti davvero intelligenti è soprattutto la

capacità di adattarsi continuamente al contesto ed all'ambiente circostante, e di saper generalizzare la conoscenza acquisita a nuove situazioni.

2. L'INTELLIGENZA ARTIFICIALE NEI VIDEOGIOCHI

Il mercato dei videogiochi è in continua espansione e il suo fatturato ha da diversi anni superato quello dell'industria cinematografica, apprestandosi a sopravanzare quello della musica e conquistare così definitivamente la leadership nel settore dell'entertainment. In parallelo, anche i videogiochi si sono evoluti, diventando sempre più realistici, complessi e sofisticati dal punto di vista grafico, sonoro e delle possibilità di interazione.

Anche per quanto riguarda l'intelligenza dei personaggi virtuali, ossia quelli non controllati dai giocatori umani, c'è stato soprattutto negli ultimi anni un sempre crescente interesse ed un sempre maggiore investimento verso sistemi di Intelligenza Artificiale (IA) che conferiscano loro una maggiore plausibilità. Basta dare un'occhiata al numero di libri e pubblicazioni usciti sull'argomento (si veda ad esempio Buckland, 2002; Deloura e Rabin, 2002; Cass, 2002; Johnson e Wiles, 2001) per rendersi conto di quanto importante sia al giorno d'oggi creare giochi che siano realistici non solo nelle ambientazioni che riescono a creare, ma anche nel tipo di interazione che rendono possibile con i personaggi virtuali. Nella maggior parte dei casi, ci si limita ad adattare ai videogiochi le più note tecniche di Intelligenza Artificiale classica, come ad esempio macchine a stati finiti, sistemi di regole, alberi decisionali, fuzzy logic, pathfinding e così via. Lo scopo è quello di creare comportamenti che il giocatore interpreti come prodotti da un'entità autonoma, che ha i suoi obiettivi, i suoi desideri, le sue conoscenze. Si tratta, in altre parole, di puntare con il comportamento dei personaggi virtuali ai risultati raggiunti nell'ambito delle dinamiche fisiche. Oggi, grazie a potenti motori fisici utilizzati dalla maggior parte dei giochi, è possibile simulare molte delle interazioni che accadono tra gli oggetti reali. Ad esempio, se controlliamo un personaggio che si tuffa in una piscina, il motore fisico è in grado di calcolare abbastanza accuratamente quale sarà lo spostamento d'acqua corrispondente, quale la velocità con la quale il personaggio raggiungerà il fondo della piscina, e così via. Il risultato finale verrà percepito dal giocatore come realistico, in quanto è quello che si aspetterebbe se davvero una persona si tuffasse in quella piscina. La stessa cosa dovrebbe accadere con il comportamento dei personaggi virtuali: le loro azioni dovrebbero venir percepite come realistiche in quanto sono quello che ci si potrebbe aspettare in situazioni reali. Così, in un ipotetico Test di Turing per i videogiochi, un'Intelligenza Artificiale soddisfacente sarebbe quella in cui, in un gioco multiplayer, non saremmo in grado di riconoscere se un

personaggio è controllato dal computer o da un altro giocatore umano. Purtroppo, nonostante le ottimistiche previsioni di qualche anno fa, ad oggi nessuno dei giochi realizzati supererebbe quasi certamente tale test.

Sono stati fatti senza dubbio notevoli sforzi in questo campo e spesso con apprezzabili risultati. Si pensi, ad esempio, a giochi come Doom o Quake, in cui i nemici sono controllati da Macchine a Stati Finiti (Finite State Machines, FSM) con risultati abbastanza soddisfacenti. Le FSM sono state spesso usate per giochi in prima persona e sono molto diffuse in quanto facile da testare, modificare e personalizzare. Per implementare un nemico in Doom, ad esempio, basta programmarlo in modo che controlli periodicamente l'ambiente e quando avvista il giocatore decida se attaccarlo o no in base a semplici regole (il giocatore ha sparato? si sta avvicinando?). Per situazioni un po' più complesse, che richiedono di controllare anche condizioni non binarie, sono stati utilizzati sistemi che utilizzano ad esempio la logica Fuzzy, come nel caso di Unreal o Civilization per citare alcuni dei giochi più noti.

Se tuttavia sono stati trovati algoritmi sempre più potenti ed efficienti, l'impressione è che all'evoluzione in atto per quanto riguarda le capacità grafiche e sonore dei videogiochi, non sia seguita una corrispondente evoluzione per quanto riguarda il comportamento dei personaggi virtuali, pregiudicando spesso drasticamente il realismo dei giochi. Tipicamente, infatti, tali personaggi (soldati, animali, nemici, alieni, ecc.) possiedono un comportamento molto elementare e schematico, per cui solitamente il giocatore trova degli "stratagemmi" che, usati in maniera sistematica, permettono di avere la meglio nei loro confronti. A conferma di ciò, esistono oggi numerose riviste, siti web, gruppi di discussione dove è possibile scambiarsi informazioni su come superare un avversario o un livello, seguendo rigorosamente una lista di passi da compiere nell'ordine specificato. Tutto ciò è possibile perché il loro comportamento segue regole rigide, indipendenti dal contesto in cui si trovano ad operare. Da questo punto di vista, nonostante siano state proposte come detto varie tecniche di Intelligenza Artificiale, non sono stati fatti molti passi avanti dai tempi di Pacman e Space Invaders. Il comportamento è diventato sicuramente più complesso, le regole più sofisticate e numerose, sono state utilizzate euristiche realistiche, ma il problema di tali tecniche è che esse non sono adattive, ovvero non permettono ai personaggi di adattarsi al contesto e al comportamento del giocatore. Se un nemico è stato programmato a sparare quando il giocatore arriva ad una certa distanza, lo farà ogni volta che il giocatore supererà quella distanza. Il che può creare dei seri problemi: se in un gioco del calcio basta tirare in diagonale per spiazzare il portiere, non c'è modo per riparare a questa debolezza, ed il portiere continuerà a subire goal dalla diagonale anche dopo cento, mille partite.

Allo stesso modo, i personaggi virtuali non sono in grado di affrontare situazioni per cui non sono stati specificatamente programmati. In questi casi, così come i tradizionali programmi, essi rischiano di bloccarsi, o di agire in modo completamente inadeguato alla situazione. Cosa succede se il personaggio controllato dal giocatore si trova in una posizione che i programmatori non avevano preso in considerazione? Spesso sono proprio queste situazioni che rischiano di rovinare un gioco altrimenti ben fatto: gli utenti si rivelano infatti molto insofferenti verso situazioni di completa implausibilità, anche se si tratta di casi isolati.

3. SYNAPTHINK

SynapThink è il nome di una serie di metodologie che, ispirandosi alla plasticità propria del sistema nervoso (le sinapsi costituiscono il punto di contatto tra due o più neuroni, e l'apprendimento neurale sembra consistere proprio nella modifica di tali sinapsi), si propone di rendere adattivi e flessibili i sistemi intelligenti attuali. Nel caso dei videogiochi, questo significa permettere ai personaggi virtuali di modificarsi e di migliorare la propria performance nel corso del gioco stesso: il portiere imparerà come comportarsi con i tiri in diagonale, i nemici impareranno ad impedire i soliti “trucchetti” nei loro confronti, e così via. Non solo. Un personaggio adattivo non avrà solamente una serie di opzioni prestabilite tra cui scegliere quella più idonea alla situazione, ma potrà sperimentare varie possibilità per vedere quale è la migliore ed agire poi di conseguenza. Questo significa che esso potrà decidere di comportarsi in modo completamente inaspettato anche per il programmatore che lo ha ideato.

Perché ciò sia possibile, è necessario innanzi tutto utilizzare un sistema intelligente che mostri maggiore flessibilità rispetto a quelli classici. Una possibilità è quella di utilizzare un modello di quello che in natura è il sistema più flessibile ed adattivo, ossia il sistema nervoso. Il nostro cervello è infatti caratterizzato da un'architettura di tipo parallelo e da un elevato grado di plasticità, grazie alla possibilità di creare nuove sinapsi tra i neuroni, e di modificare o eliminare sinapsi già esistenti. Le Reti Neurali Artificiali rappresentano appunto una simulazione delle reti di neuroni del cervello e tentano di riprodurre i meccanismi principali alla base del loro funzionamento. Tra le caratteristiche principali di tali reti vi è la loro capacità di apprendere, di svilupparsi ed in generale di modificarsi in base al contesto in cui si trovano ad operare. A differenza dei sistemi di Intelligenza Artificiale classica, le reti neurali non vengono preprogrammate da chi le crea, ma piuttosto addestrate, solitamente mostrando degli esempi; inoltre, anche una volta addestrate continuano ad essere plastiche e capaci di imparare nuove cose; infine, le reti neurali sono in grado di generalizzare la loro conoscenza

rispondendo a nuovi stimoli, a nuove situazioni, senza la necessità di essere riprogrammate.

Ma il cervello umano non si modifica solo tramite l'apprendimento. Anche altri fattori, quali quelli genetici, culturali e sociali, devono essere presi in considerazione se si vogliono ricreare quelle proprietà che contraddistinguono l'intelligenza biologica. Per questo motivo, non basta utilizzare una rete neurale per ottenere un sistema adattivo, ma è necessario anche simulare tutti quegli aspetti che costituiscono l'ambiente in cui la rete neurale si trova ad operare. Da questo punto di vista SynapThink può essere visto come un'applicazione dell'approccio della Vita Artificiale (Langton, 1989; Parisi, 1999) alla creazione di videogiochi. In una simulazione di Vita Artificiale vengono create entità biologiche artificiali che evolvono e apprendono interagendo con l'ambiente e tra di loro. SynapThink applica questi stessi principi al mondo dei videogiochi, creando personaggi che evolvono e apprendono nel corso del videogioco o anche off-line, cioè quando il giocatore smette di giocare. Personaggi che possono interagire tra di loro o con il giocatore stesso (con un personaggio da lui controllato), che possono modificare l'ambiente o adattarsi ad esso quando è il giocatore a modificarlo (tramite il suo personaggio).

Con SynapThink si intende dunque un nuovo modo di guardare ai sistemi intelligenti, che può essere applicato a vari ambiti tra cui quello dei videogiochi. In quest'ultimo caso, SynapThink si propone di rendere più realistico il comportamento dei personaggi virtuali non solo utilizzando sistemi più flessibili (le reti neurali) rispetto a quelli classici, ma anche adottando una visione diversa dell'intero ambiente in cui essi si trovano ad operare.

4. NUOVE TIPOLOGIE DI GIOCHI

In virtù di questa nuova concezione dei personaggi di un videogioco e dell'ambiente in cui essi vivono, è possibile anche pensare a nuove tipologie di gioco, oltre che migliorare quelle esistenti. Di seguito sono elencate alcune delle più promettenti prospettive:

1- *L'adattamento come obiettivo*. Scopo del giocatore è quello di insegnare ad uno o più personaggi a svolgere un certo compito. In base alla sua bravura il giocatore può ottenere un punteggio, oppure può osservare il proprio personaggio competere con quelli addestrati da altri giocatori. La sfida può disputarsi nel computer di uno dei due giocatori, ma anche su Internet o comunque senza la presenza dei giocatori. Ciò permetterebbe la creazione di veri e propri tornei virtuali. Un'altra possibilità è che il giocatore collabori con i propri "allievi" nel raggiungimento di un certo risultato. Quest'ultima modalità potrebbe risultare estremamente utile per tutti quei giochi, come quelli di squadra, dove non è possibile ricoprire contemporaneamente tutti i ruoli. Ad esempio, nei tradizionali giochi

di calcio si controlla un solo calciatore per volta, e non si ha la possibilità di insegnare agli altri calciatori come comportarsi quando non hanno la palla (ossia nella maggior parte del tempo), se non tramite regole molto generali. Utilizzando metodologie adattive, invece, si avrebbe la possibilità di insegnare direttamente ai propri “compagni di squadra” come muoversi, semplicemente mostrandoglielo. Un’ulteriore possibilità è che il giocatore addestri un personaggio per poi vedere gli effetti dell’insegnamento su un dispositivo fisico (quale un robot), anziché sul monitor del computer. Ad esempio, il giocatore potrebbe addestrare sul suo computer un pilota virtuale a guidare una piccola automobile e poi “trasferire” le conoscenze del pilota su un’automobilina reale programmabile. Un’ulteriore possibilità, infine, è che il giocatore crei ed addestri il personaggio di un MMOG (Massive Multiplayer Online Game), una tipologia di giochi online in cui il giocatore attraverso un proprio personaggio vive in un mondo virtuale persistente (ossia che continua ad esistere anche quando gli utenti sono offline) e interagisce con i personaggi degli altri giocatori. SynapThink darebbe la possibilità al giocatore di influenzare lo sviluppo del proprio personaggio, contribuendo ad accrescere la sua conoscenza e le sue abilità fisiche in modo che anche quando egli non è connesso, il suo personaggio possa continuare a vivere nel mondo virtuale ed interagire autonomamente con gli altri personaggi secondo le “indicazioni” del suo creatore.

2- *L’adattamento come processo spontaneo.* Il giocatore non addestra esplicitamente i personaggi del videogioco, ma essi si adattano spontaneamente. Per esempio, i nemici possono imparare a combattere meglio, i calciatori avversari possono imparare gli schemi tattici del giocatore e comportarsi di conseguenza. Oppure, potremmo avere una storia con un “partner virtuale”, che impara a conoscere il nostro carattere proprio come un partner reale, e cambiare il proprio comportamento in base al nostro. In questo caso, il nostro scopo non sarebbe quello di addestrare il partner, ma per esempio quello di convincerlo a sposarci. Un’altra interessante possibilità è che alcuni personaggi imparino ad imitare il giocatore, per poi sostituirsi a lui in determinate occasioni. Questo permetterebbe ad esempio di sfidare se stessi in giochi come il tennis, la lotta, ecc. Oppure, avere una copia di noi stessi ci permetterebbe di sfidare più avversari (umani) contemporaneamente, anche quando siamo impossibilitati fisicamente a farlo. Un’ulteriore possibilità è che il giocatore crei solamente le condizioni perché i personaggi apprendano e si adattino, piuttosto che intervenire direttamente su di essi. Ad esempio, in un gioco che simula l’evoluzione di alcune forme di vita, potremmo dover decidere parametri importanti (ad esempio, riproduzione sessuata o asessuata), osservando poi l’effetto di tali modifiche sugli organismi. L’evoluzione potrebbe continuare ad operare anche in nostra assenza:

potremmo per esempio andare a dormire e svegliarci il giorno dopo per verificare cosa è emerso nel corso delle generazioni.

Un esempio pratico di applicazioni di tali idee, utilizzando la metodologia SynapThink, è costituito da un gioco per telefoni cellulari dal nome provvisorio di SynapCar (Fig. 2). Si tratta di un gioco di tipo automobilistico, ma a differenza dei giochi tradizionali il giocatore non riveste il ruolo del pilota, ma quello dell'insegnante. Infatti, il suo compito non è di guidare un'automobile lungo un percorso, ma di addestrare un pilota a farlo. In questo modo, cambia completamente la tipologia del gioco: non si tratta più di far valere la propria prontezza di riflessi ed abilità manuale, ma di elaborare una propria strategia di addestramento, il che lo rende più vicino ad un gioco di intelligenza.

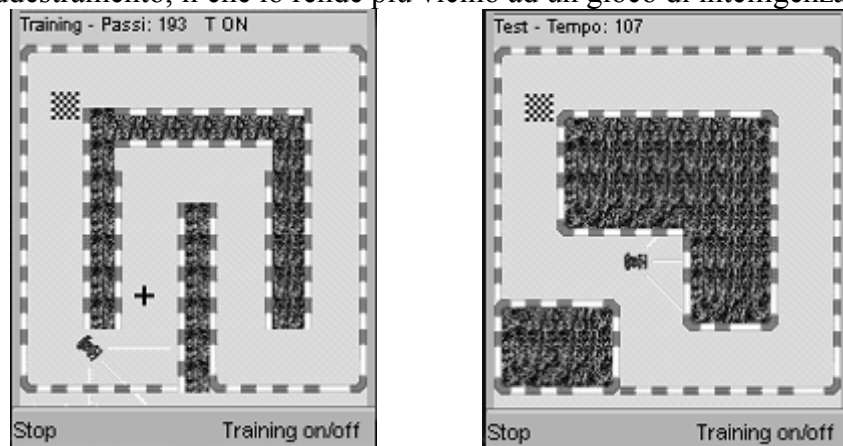


Figura 2. Due fasi di SynapCar: il training (sinistra) e il test (destra).

Una volta terminata la fase di addestramento, il giocatore può osservare i progressi fatti dall'“allievo”, mettendolo alla prova su vari percorsi. In questa fase, il pilota è completamente autonomo, in quanto possiede un sistema di controllo che si è formato secondo quanto precedentemente appreso. SynapCar, inoltre, non si esaurisce esclusivamente nella modalità “single player”, che consiste appunto nel cercare di addestrare un pilota a compiere un percorso o un insieme di percorsi nel minor tempo possibile, ma trova la sua naturale evoluzione nella modalità “multi player”, in cui i piloti addestrati da differenti giocatori competono tra di loro in una gara automobilistica. La gara si può svolgere nel telefonino di uno dei giocatori, oppure su Internet, dove è possibile inviare i dati relativi al proprio pilota e farlo partecipare ad emozionanti gare online.

5. UNA RIVOLUZIONE IN ATTO

Le idee alla base di SynapThink rappresentano oggi il punto di vista emergente nel settore dell'Intelligenza Artificiale e ricevono sempre maggiori consensi nella comunità del Game Programming. In particolare, l'utilizzo delle tecniche e dei metodi propri delle

Neuroscienze Computazionali e della Vita Artificiale è supportata da recenti autorevoli studi che imputano il relativo fallimento delle tecniche di IA, rispetto alle grandi aspettative iniziali, a due fattori principali:

- la loro scarsa aderenza ai principi della computazione corticale, ossia al modo di funzionare del cervello, che rappresenta ancora la più elevata forma di intelligenza (a questo riguardo, si veda ad esempio il best-seller *On Intelligence* (2004) del “guru” della tecnologia Jeff Hawkins, fondatore di Palm Computing e di Handspring).

- la loro tendenza a studiare i sistemi intelligenti artificiali (tra cui le reti neurali) isolatamente, ossia senza la presenza di un ambiente fisico con cui poter interagire, ed indipendentemente da altri importanti fattori come quelli genetici, culturali e sociali. Si è visto invece che SynapThink abbraccia una visione più ampia, denominata Vita Artificiale, che sta emergendo nell’ambiente scientifico come nuova disciplina e può rappresentare un punto di forza importante quando si vogliono utilizzare sistemi intelligenti per applicazioni che prevedono già l’esistenza di ambienti interattivi e complessi, come appunto quelli dei videogiochi.

Due motivi principali hanno fino ad oggi ostacolato il diffondersi di queste idee al mondo dei videogiochi. Innanzitutto, la diffidenza nella comunità degli sviluppatori di videogiochi nei confronti di tutti quei metodi non strettamente deterministici, quali le reti neurali o gli algoritmi genetici. Anche perché spesso non c’è una sufficiente comprensione da parte di tale comunità dei meccanismi che stanno alla base di queste nuove tecniche. In secondo luogo, la mancanza di risorse e di tempo per poter sviluppare complessi sistemi di IA. Fino ad oggi, infatti, i maggiori investimenti sia di risorse che di tempo si sono concentrati nello sviluppo della parte grafica, in quanto in grado di attrarre molti più giocatori, mentre l’IA è stata spesso considerata una componente aggiuntiva, inserita solo in un secondo tempo, a gioco quasi completato. Se tuttavia fino ad oggi si è potuto relativamente trascurare l’IA di un gioco e concentrarsi sulla sua veste grafica, negli ultimi anni in cui accade sempre più comunemente di trovare giochi dalla grafica spettacolare, si comincia ad avvertire l’esigenza di trovare altri aspetti in grado di fare la differenza. Ecco quindi che sono sempre più numerosi gli esperti di Game Programming che vedono proprio nelle tecniche adattive una soluzione ai problemi dell’IA classica. Autorevoli voci, come quella di Matt Buckland, autore di “*AI Techniques for Game Programming*” (2002), stanno cominciando ad interessarsi alle reti neurali, agli algoritmi genetici ed in breve a quel nuovo approccio allo studio dell’intelligenza che prende il nome di Vita Artificiale. Non a caso, Steven Woodcock, curatore dell’autorevole sito GameAI.com, può affermare che “This kind of adaptation, or learning, is something of a

Holy Grail for developers and players alike [...]. Gamers want to be challenged by an AI that actually adapts to their style of play”.

Ad oggi un sempre crescente numero di giochi utilizza le tecniche che sono alla base di SynapThink: reti neurali (es: Battlecruiser 3000AD, Heavy Gear), algoritmi evolutivi (Creatures), algoritmi di apprendimento (Galapagos). D'altra parte, il successo di giochi come “Black & White” e “The Sims” dimostra come anche i videogiocatori siano particolarmente sensibili a queste tematiche. Nel primo gioco, ad esempio, è presente una Creatura, un personaggio virtuale che agisce autonomamente e che può essere addestrato tramite ricompense e punizioni. In The Sims, invece, il giocatore è il supervisore di un certo numero di persone simulate, i Sims appunto, che agiscono autonomamente e che mostrano un certo grado di adattività (sanno reagire ad esempio a nuovi oggetti creati dai giocatori).

Se dunque le idee alla base di SynapThink stanno oggi cominciando a ricevere i primi riconoscimenti, manca ancora una sistematizzazione di questo complesso di idee, di tecniche e di metodologie, ed una sua più efficiente predisposizione per l'applicazione ad ambienti diversi da quelli accademici. SynapThink si propone proprio di trasferire le conoscenze acquisite nel campo delle Neuroscienze Computazionali e della Vita Artificiale a nuovi settori di applicazione. E quello del Game Programming appare un terreno particolarmente fertile dove queste idee possano svilupparsi.

RIFERIMENTI BIBLIOGRAFICI

Buckland M. (2002), *AI Techniques for Game Programming*, Cincinnati, Premier Press.

Cass S. (2002), *Mind Games*, in <<IEEE Spectrum>>, 7, pp. 40–44.

Deloura M. e Rabin S. (a cura di) (2002), *AI Game Programming Wisdom*, Boston, Charles River Media.

Hawkins J. (2004), *On Intelligence*, New York, Henry Holt.

Johnson D. & Wiles J. (2001), *Computer games with intelligence*, in <<Australian Journal of Intelligent Information Processing Systems>>, 7, pp. 61-68.

Langton C.G. (1989), *Artificial Life*, Reading (MA), Addison Wesley.

Parisi D. (1999), *Mente. I nuovi modelli della Vita Artificiale*, Bologna, Il Mulino.

Weizenbaum J. (1965), *Eliza, a computer program for the study of natural language communication between man and machine*, in <<Communication of the ACM>>, 9, pp.36-45.