

Hayek e il connessionismo: modelli con agenti che apprendono*

Pietro Terna

Dipartimento di Scienze economiche e finanziarie "G.Prato" dell'Università di Torino
corso Unione Sovietica 218bis, 10134 Torino
terna@econ.unito.it

Estratto: l'uso di modelli computazionali fondati su agenti rappresenta un valido strumento per la conduzione di esperimenti (artificiali) nelle scienze sociali. Lo scopo è definire in modo plausibile percorsi che vadano dalla interazione tra agenti semplici alla complessità del sistema economico e sociale.

Si presenta lo sviluppo di un semplice mercato fondato su agenti, costruito tramite un codice a oggetti fondato su Swarm (<http://www.santafe.edu/projects/swarm/>); gli agenti sono alternativamente strutturati con regole o con reti neurali. Nel primo caso si descrive il comportamento degli agenti, sulla base di regolarità (statistiche) note *ex ante*; nel secondo, le regole derivano dall'apprendimento degli agenti mentre agiscono all'interno del mercato simulato. Si sperimentano così i legami tra il "ragionamento" degli agenti, le azioni conseguenti e l'ordine che emerge nel mercato.

Abstract: Agent based computational models are basic tools useful to run (artificial) experiments in social sciences. Our goal is the definition of plausible paths from the interaction among simple agents to the emergence of complexity in social and economic systems.

We simulate a simple market, based on artificial agents; the model is built upon an object oriented code in Swarm (<http://www.santafe.edu/projects/swarm/>); agents are alternatively wired with rules or neural networks. In the first case agent's behavior follows (statistical) regularities known *ex ante*; in the second case, rules arise from the agents' learning processes, while interacting in the simulated market. So we can investigate the links connecting agents' "reasoning" and agents' behavior to the kind of order emerging in the market.

1. Premessa: un programma hayekiano di ricerca con modelli fondati su agenti

Epstein e Axtell (1996, Cap. 1, *Introduzione*) annotano:

Herbert Simon ama affermare che le scienze sociali sono, nei fatti, le scienze *hard*. In primo luogo, molti processi sociali crucialmente importanti sono complessi. Non sono scomponibili in modo ordinato in sub-processi separati - economico, demografico, culture, territoriale - le cui analisi distinte possano essere aggregate per fornire una analisi adeguata del processo sociale come un tutt'uno. Finora questo è esattamente il modo in cui il campo delle scienze sociali è organizzato, in comparti che sono più o meno delle isole e con riviste di economia, demografia, scienza politica e così via (...)

Le scienze sociali sono anche difficili perché certi tipi di sperimentazione controllata sono difficoltosi. In particolare è difficile sottoporre a test le ipotesi relative ai legami tra comportamenti individuali e regolarità a livello macro. Se gli individui agiscono in questo o quel modo - cioè, seguono specifiche regole - allora la società come un tutto avrà certe proprietà. Come opera l'eterogeneo livello micro dei comportamenti individuali nel generare le regolarità della società al livello macro?

Un'altra preoccupazione di fondo di molti scienziati sociali è che l'agente razionale - un individuo perfettamente informato con capacità di calcolo illimitata che massimizza una certa funzione esogena di utilità (non soggetta a evoluzione) - ha poche relazioni con un essere umano (...)

* Questa ricerca è stata finanziata da un contributo del Ministero dell'Università e della Ricerca scientifica e tecnologica, all'interno del progetto "Intermediazione finanziaria, funzionamento dei mercati ed economia reale".

In relazione a ciò, è pratica *standard* nelle scienze sociali dimenticare l'eterogeneità degli agenti del mondo reale nella costruzione dei modelli. Ciò può essere fatto sia esplicitamente, come nel caso dell'agente rappresentativo nei modelli macroeconomici (Kirman, 1992) o implicitamente, come accade quando modelli molto aggregati sono usati per rappresentare processi sociali. Mentre tali modelli possono offrire significativi spunti interpretativi, annullano ogni conseguenza della eterogeneità. Pochi scienziati negherebbero che tali conseguenze possano essere molto importanti, ma non esiste nessuna metodologia specifica per lo studio sistematico di popolazioni altamente eterogenee.

Infine è corretto affermare che le scienze sociali, e specialmente la teoria dei giochi e la teoria dell'equilibrio economico generale, si sono sempre occupate di trovare equilibri statitici, essenzialmente ignorando le dinamiche temporali. Di nuovo, pur d'accordo sul punto, molti scienziati sociali affermerebbero che non esiste una metodologia specifica per studiare le dinamiche proprie del disequilibrio nei sistemi sociali.

Le questioni centrali sono dunque quelle dell'equilibrio e della eterogeneità degli agenti, cui si cerca di dare risposta con la metodologia dei modelli fondati su agenti (per un'ampia bibliografia, soprattutto sull'internet, vedere Terna, 1998a).

Dalla sconfinata produzione hayekiana traiamo un passo (Hayek, 1937) particolarmente significativo in questa direzione; passo cui potremo fare agevolmente riferimento nel prosieguo di questo lavoro:

Non possiamo progredire molto in questa direzione, se non ci si interroga circa le ragioni a sostegno dell' attenzione da noi riposta in una configurazione così chiaramente fittizia qual è quella di equilibrio. Qualunque argomentazione possa essere stata occasionalmente avanzata da economisti ultra-puristi, sembra non esservi dubbio alcuno che la sola possibile giustificazione per questo interesse sia da ricercare nella supposta esistenza di una tendenza verso l' equilibrio. E' solamente grazie a questo asserto che l' economia cessa di essere un esercizio di pura logica e diventa una scienza empirica: ed è all'economia in quanto scienza empirica che dobbiamo ora volgere la nostra attenzione.

Alla luce della nostra analisi del significato da attribuire ad uno stato di equilibrio, dovrebbe risultare facile stabilire qual è il reale contenuto dell' asserto dell' esistenza di una tendenza verso l' equilibrio. A ben considerare non può trattarsi d' altro che di questo: che sotto determinate condizioni si ritiene che il grado di conoscenza e le intenzioni dei differenti soggetti convergano sempre di più ovvero - per esporre la stessa idea in termini meno precisi e generali, ma certamente più concreti - che le aspettative della gente e in particolar modo quelle degli imprenditori diventino via via più corrette. Posta in questi termini, l' affermazione dell' esistenza di una tendenza verso l' equilibrio diviene chiaramente una proposizione empirica, cioè un asserto su ciò che avviene nel mondo reale che, almeno in linea di principio, si dovrebbe riuscire a verificare. E questo ci consente di attribuire alla nostra piuttosto astratta formulazione del concetto di equilibrio un significato a livello di senso comune. Il solo guaio è che brancoliamo, ancora, nel buio per quanto concerne:

1. le *condizioni* in base alle quali si suppone esista questa tendenza verso l'equilibrio,
2. la natura del *processo*, mediante il quale cambia la conoscenza individuale.

Nella tradizionale presentazione dell' analisi di equilibrio, la questione del modo in cui si realizza la configurazione di equilibrio viene tacitamente indicata come risolta. Se però sottoponiamo ad un più attento esame la cosa, risulta allora subito evidente che queste apparenti dimostrazioni non fanno altro che dimostrare ciò che in effetti si era già ipotizzato. Lo stratagemma generalmente adottato a tal fine consiste nell' assumere un mercato perfetto, dove ogni evento è conosciuto istantaneamente da ciascun individuo. E' opportuno rammentare a tale riguardo che il mercato perfetto, la cui esistenza è richiesta per soddisfare le ipotesi dell' analisi di equilibrio, non deve essere limitato ai mercati di tutte le singole merci; è l' intero sistema economico che deve essere ipotizzato alla stregua di un unico mercato perfetto, nel

quale ciascuno è a conoscenza di tutto. L'ipotesi di un mercato perfetto, pertanto, significa semplicemente che tutti i membri della collettività, anche se non onniscienti in senso stretto, si ritiene conoscano perlomeno automaticamente tutto quanto è rilevante per le loro decisioni. E' come se l' "uomo economico", questa nostra vergogna di famiglia che abbiamo esorcizzato con la preghiera e il digiuno, fosse rientrato per la porta di servizio sotto la veste di un individuo quasi onnisciente. L'affermazione, secondo la quale, se i soggetti conoscono tutto essi si trovano in equilibrio, è certamente vera, ma solo perché ciò corrisponde al modo in cui noi definiamo il concetto di equilibrio. L'ipotesi di un mercato perfetto altro non è, da questo punto di vista, che un modo diverso di dire che l'equilibrio esiste, ma ciò non ci avvicina affatto alla spiegazione del come e quando tale configurazione di equilibrio si realizza. E' chiaro, infatti, che se vogliamo affermare che i soggetti conseguiranno, sotto certe condizioni, tale stato di equilibrio, dobbiamo altresì spiegare attraverso quale processo essi acquisiranno la necessaria conoscenza. Chiaramente, qualsiasi assunzione venga formulata circa l'effettiva acquisizione delle informazioni nel corso di tale processo, essa avrà pur sempre natura ipotetica. Ma ciò non vuol assolutamente dire che tutte queste assunzioni siano ugualmente giustificabili. Abbiamo infatti a che fare, in relazione a questo problema, con ipotesi che concernono processi causali, e pertanto ciò che ipotizziamo non solo deve potersi considerare possibile (il che non è certamente il caso, se pensiamo agli individui come esseri onniscienti), ma anche verosimile; inoltre, deve essere possibile, almeno in linea di principio, dimostrare che ciò che ipotizziamo è vero in relazione a casi determinati.

2. La costruzione di modelli di simulazione fondati su agenti

Una possibile risposta risiede nella costruzione di modelli di simulazione che, tramite il *computer*, determinino l'interazione di agenti artificiali, intesi come programmi in grado di eseguire compiti o assumere decisioni. Dalla interazione di quegli agenti, che si svolge nel tempo, emerge una economia che dipende dalle regole, dalla struttura degli agenti, dalla loro capacità di apprendere. Una economia di cui possiamo studiare equilibri, disequilibri, stati transitori, con agenti eterogenei o no, ricercando la plausibilità dei singoli agenti come dell'economia nel suo complesso.

Modelli di questo tipo sono tanto più utili quanto più è agevole replicare gli esperimenti che ne discendono, sia per verifica, sia per sondare prospettive diverse di analisi, anche variando i parametri del sistema (grandezze relative agli agenti, all'ambiente, alla numerosità degli agenti o di altre entità - ad esempio prive di regole di comportamento, quali oggetti inanimati - presenti nell'ambiente).

La replicabilità di questo tipo di esperimenti non è senza costi; ne ho esperienza diretta con l'uso di un mio *software* - denominato CT dal nome del metodo dei *Cross Target* (vedi oltre) - per la costruzione di modelli fondati su agenti costruiti con reti neurali, precedente a quello utilizzato per l'esperimento riportato nel paragrafo 4 di questo saggio: anche lavorando con molta accuratezza, non tanto la replica dell'esperimento, ma certo l'introduzione di modifiche è molto difficoltosa, soprattutto a distanza di tempo. Per questo l'originario CT è stato completamente riscritto, seguendo le regole di un ambiente che possa essere considerato uno *standard*; si tratta in questo caso della biblioteca di oggetti in *Objective C*, e dell'insieme dei protocolli da applicare per utilizzarli, denominati *Swarm* (<http://www.santafe.edu/projects/swarm>).

Anche un modello scritto utilizzando una piattaforma *standard* può essere però molto difficile da utilizzare per chi non ne sia l'autore. Inoltre può diventare - con il tempo - difficile da modificare per lo stesso autore. Per superare

queste difficoltà e per mettere a disposizione degli interessati uno schema molto generale, come guida di metodo per la costruzione di modelli di agenti, si introduce lo schema riportato nella Figura 1.

La caratteristica principale dello schema *Environment Rules Agents* (ERA) è quella di separare l'ambiente, con le sue regole, dagli agenti, che non comunicano direttamente tra loro (come costruito informatico), ma tramite l'ambiente; ad esempio, l'ambiente fornisce ad ogni agente le informazioni necessarie per costruire una lista dei propri vicini. Per le proprie azioni gli agenti ricevono istruzioni da oggetti esterni, da intendersi come rappresentazione di enti astratti quali insiemi di regole o di meta-regole, queste ultime viste come strumenti da utilizzare per modificare le regole. Si tratta di gestori di regole che infatti a loro volta comunicano con oggetti produttori di regole che modificano l'azione degli agenti, per esempio simulando l'apprendimento. Tutte le informazioni necessarie ad un gestore di regole provengono dall'agente oggetto dell'azione o da agenti speciali che operano nell'ambiente con funzioni statistiche; allo stesso modo le informazioni necessarie per l'azione dei produttori di regole provengono dai gestori di regole, che le ricevono dagli agenti chiamati in causa dalle regole stesse. La rigidità della struttura, che rifiuta legami diretti tra livelli diversi (i quattro blocchi verticali della figura) e non consente collegamenti diretti all'interno dei livelli, aumenta il peso iniziale della struttura del codice informatico, ma si ripaga abbondantemente grazie alla maggior comprensibilità e agevolezza delle modificazioni.

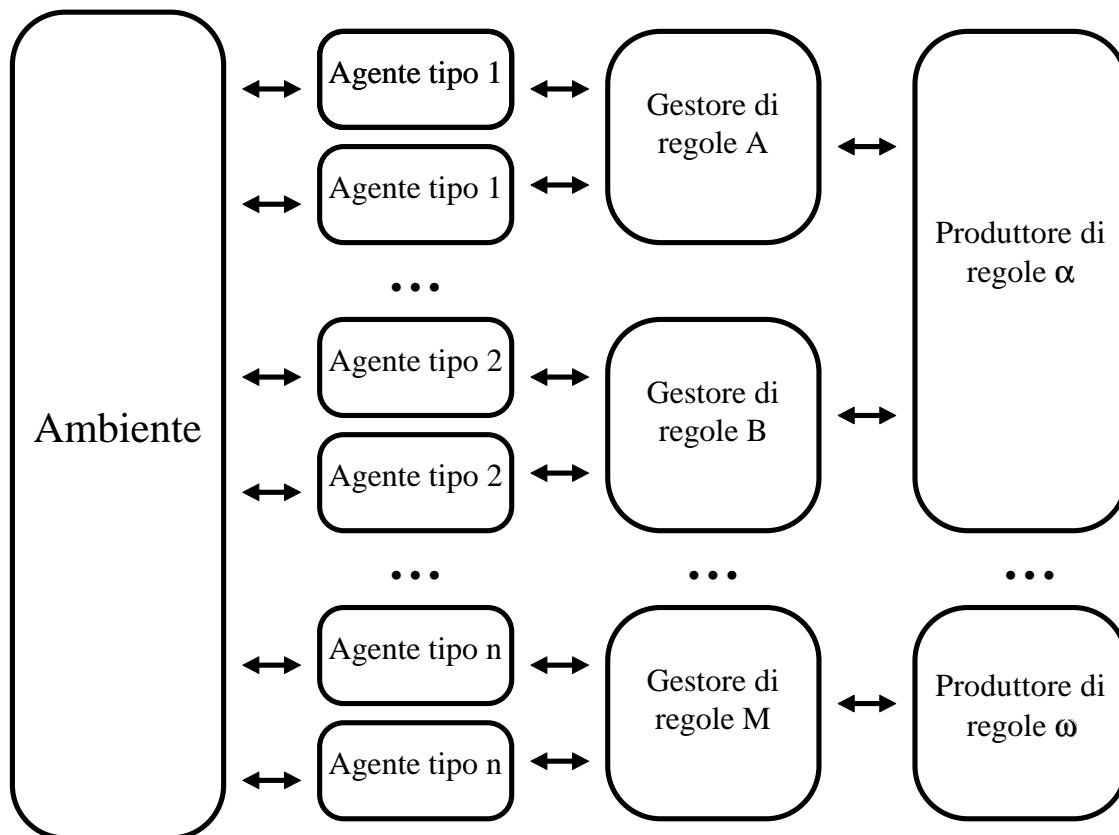


Fig. 1. Schema *Environment Rules Agents* (ERA) per la costruzione di modelli

L'architettura proposta deve ovviamente appoggiarsi su ambienti informatici ad oggetti, con particolare attenzione alla piattaforma *Swarm*. Una ulteriore motivazione per la sua adozione è la modularità, che consente

facilmente di utilizzare quasi in modo intercambiabile *classifier system* o reti neurali o insiemi di regole o passi evolutivi o altro strumento, nei livelli dei gestori e dei produttori di regole.

3. Un esperimento di mercato con regole note a priori e senza apprendimento

Questo esempio riguarda un esperimento fondato su agenti nel campo della simulazione degli scambi e della negoziazione. L'esperimento può essere modificato o replicato utilizzando Swarm (a partire dalla versione 1.0.5; la versione corrente è la 1.4.1) richiedendo i *file* sorgente all'autore.

L'esperimento, seguendo Terna (1998b) mostra l'emergere di sequenze caotiche di prezzi in un semplice modello interattivo di compratori e venditori, equipaggianti con regole minimali.

Gli studiosi del caos esaminano insiemi di equazioni che siano in grado di produrre serie di dati apparentemente casuali, ancorché deterministici nella formazione. Ma con quale plausibilità? In questo esperimento il caos emerge come effetto collaterale dell'azione degli agenti.

Operiamo con 10 consumatori e 10 venditori, costruiti come oggetti, cioè piccoli programmi in Objective C, capaci di reagire a messaggi, per esempio di decidere se comperare o vendere e a quale prezzo. Consumatori e venditori operano in ordine casuale ad ogni ciclo di tempo: ogni consumatore cerca un venditore e confronta il proprio prezzo soglia, al di sopra del quale non compera, con quello del venditore.

In dettaglio, la sequenza degli eventi, che comprende l'aggiornamento dei prezzi di ciascun compratore e di ciascun venditore, è la seguente:

- ad ogni intervallo di tempo t , ogni consumatore artificiale (agente) incontra un venditore artificiale (altro agente), scelto casualmente;
- il venditore stabilisce il proprio prezzo di vendita, scelto in un piccolo intervallo;
- il consumatore accetta l'offerta solo se il prezzo di vendita cade al di sotto del suo livello di prezzo di acquisto;
- ad ogni intervallo di tempo t , ogni agente (consumatore o venditore) incrementa il proprio contatore delle transazioni di 1 unità, se compie una transazione; lo diminuisce di 1 unità nel caso opposto;
- quando i loro contatori valgono meno di -1 (sensitività = 0) o meno uguale -1 (sensitività = 1), i consumatori aumentano il livello del proprio prezzo di acquisto;
- quando i loro contatori valgono più di 1 (sensitività = 0) o più uguale 1 (sensitività = 1), i consumatori diminuiscono il livello del proprio prezzo di acquisto;
- quando i loro contatori valgono meno di -1, i venditori riducono di un ammontare casuale scelto tra zero e un valore fisso, entrambi i limiti l'intervallo all'interno del quale scelgono, in modo casuale, il proprio prezzo di vendita;
- quando i loro contatori valgono più di 1, i venditori aumentano di un ammontare casuale scelto tra zero e un valore fisso, entrambi i limiti l'intervallo all'interno del quale scelgono, in modo casuale, il proprio prezzo di vendita.

In tutti gli esperimenti condotti all'interno di questo schema il risultato è quello di un prezzo ciclico, con transizioni caotiche da una fase all'altra. La prima è una emergenza imprevedibile, ma in qualche modo prevedibile, dato che il meccanismo sopra descritto introduce un ciclo delle scorte nel modello. La seconda è una emergenza imprevedibile: il caos è ovviamente osservabile nei fenomeni economici e sociali, ma non è agevole predisporre uno schema di *reverse engineering* in grado di produrlo come interazione tra agenti.

Definiamo i parametri delle simulazioni riportate nelle figure di questo paragrafo e nella Tabella 1. Parametri: “theLevel” è il prezzo iniziale al di sotto del quale il consumatore acquista (tale prezzo cambia indipendentemente per ciascun consumatore, con la simulazione); “agentNumber” è il numero di consumatori e di venditori; “minStartPrice” e “maxStartPrice” sono i valori iniziali all’interno dei quali i venditori scelgono il loro prezzo di vendita (tali valori cambiano indipendentemente per ogni venditore durante la simulazione); “use_printf” è un parametro tecnico per la stampa dei risultati; il “reactivityFactor” è un fattore moltiplicativo che rinforza i valori usati dai consumatori e dai venditori per modificare i propri livelli di prezzo; “sensitivity” è zero o uno, con il significato detto sopra.

Negli esperimenti riportati nelle figure da 2 a 5, il punto di partenza è un livello di prezzo di acquisto di 50 (su una scala da 0 a 100) e un prezzo di vendita tra 45 e 55 sulla stessa scala. Consumatori e venditori hanno gli stessi parametri iniziali, che si modificheranno con la simulazione. La memoria del sistema sta dunque nell’interazione tra consumatori e venditori.

Parametri e Esperimenti	Fig. 2	Fig. 3	Fig. 4	Fig. 5	Fig. 6	Fig. 7	Fig. 8	Fig. 9
theLevel	50	50	50	50	25	25	25	25
agentNumber	10	10	10	10	10	10	10	10
minStartPrice	45	45	45	45	70	70	70	70
maxStartPrice	55	55	55	55	90	90	90	90
use_printf	0	0	0	0	0	0	0	0
reactivityFactor	1	1	3	3	1	1	3	3
sensitivity	0	1	0	1	0	1	0	1
standard Swam random seed	yes	yes	yes	yes	yes	yes	yes	yes

Tabella 1. Parametri degli esperimenti delle Figure da 2 a 9.

I prezzi riportati sono medie, minimi e massimi di tutti i prezzi offerti in ogni ciclo. (La “alternative way” dei titoli è riferita ad un problema interno di calcolo).

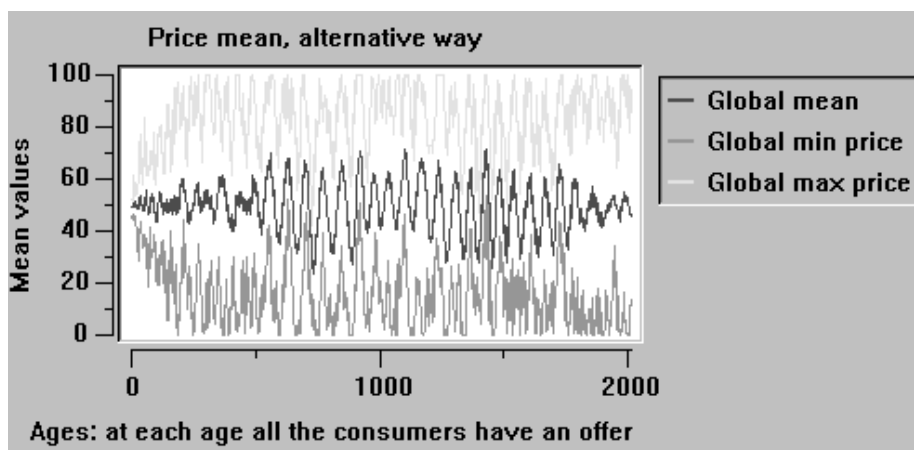


Fig. 2. L’esperimento inizia da una situazione equilibrata, con bassa reattività e bassa sensitività.

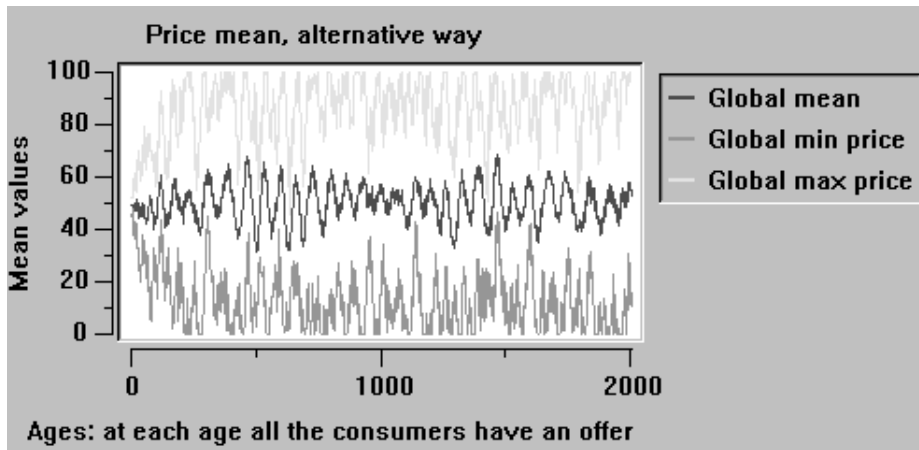


Fig. 3. L'esperimento inizia da una situazione equilibrata, con bassa reattività e alta sensibilità.

Negli esperimenti riportati nelle figure da 2 a 5, il punto di partenza è un livello di prezzo di acquisto di 50 (su una scala da 0 a 100) e un prezzo di vendita tra 45 e 55 sulla stessa scala. Consumatori e venditori hanno gli stessi parametri iniziali, che si modificheranno con la simulazione. La memoria del sistema sta dunque nell'interazione tra consumatori e venditori.

Anche gli esperimenti riportati nelle Fig. 3 e 4 iniziano da una situazione bilanciata, come descritto all'inizio del paragrafo, ma con un valore di reattività elevato che aumenta gli effetti ciclici; nel secondo caso la presenza del parametro di sensibilità riduce l'ampiezza delle fluttuazione, ma il caos emerge comunque (vedere sempre Terna 1998b).

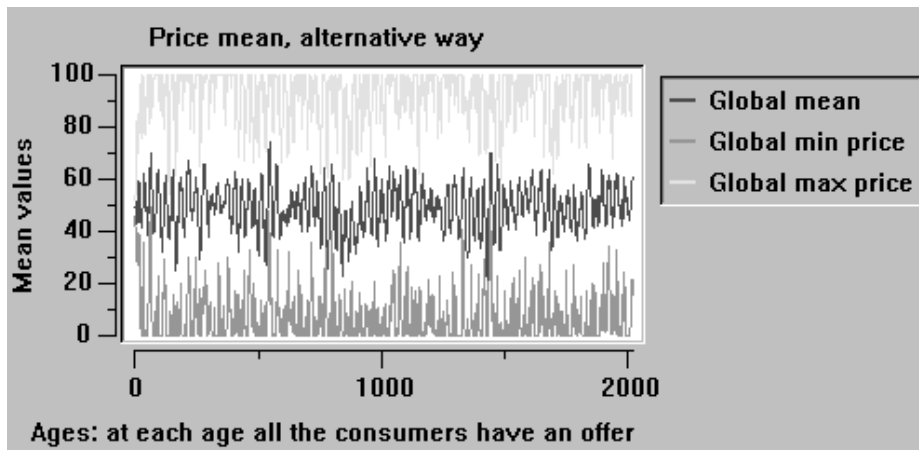


Fig. 4. L'esperimento inizia da una situazione equilibrata, con alta reattività e bassa sensibilità.

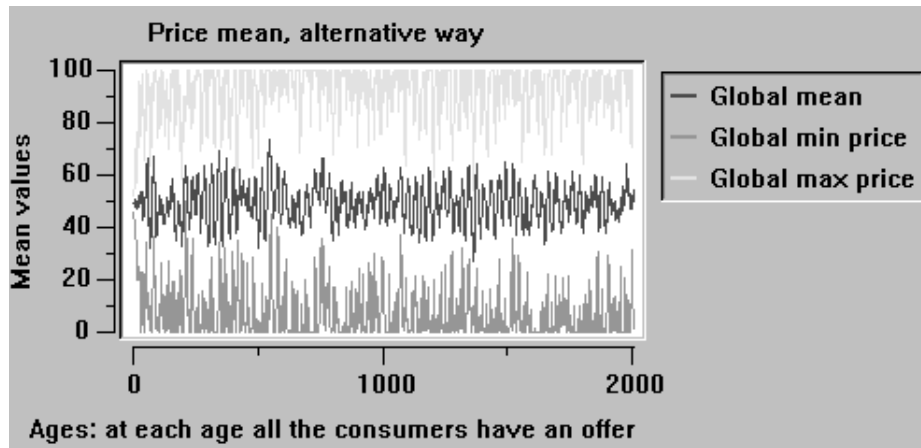


Fig. 5. L'esperimento inizia da una situazione equilibrata, con alta reattività e alta sensibilità.

Gli esperimenti delle Fig. da 6 a 9 utilizzano una situazione non equilibrata, con un livello del prezzo di acquisto pari a 25, su una scala da 0 a 100, e con il prezzo di vendita scelto tra 70 e 90 sulla medesima scala.

Con un fattore di reattività basso, nelle Fig. 6 e 7, l'emergenza del caos è evidente e la presenza del parametro di sensibilità rinforza la situazione.

Con un fattore di reattività alto, nelle Fig. 8 e 9, l'effetto ciclico sovrasta quello caotico; comunque, nella Fig. 9, introducendo il parametro di sensibilità, il caos riappare (vedere sempre per le relative misure Terna 1998b).

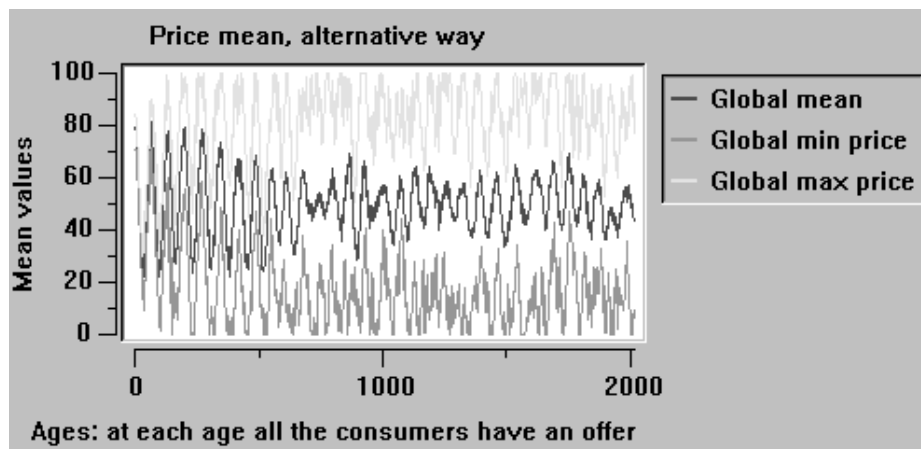


Fig. 6. L'esperimento inizia da una situazione non equilibrata, con bassa reattività e bassa sensibilità.

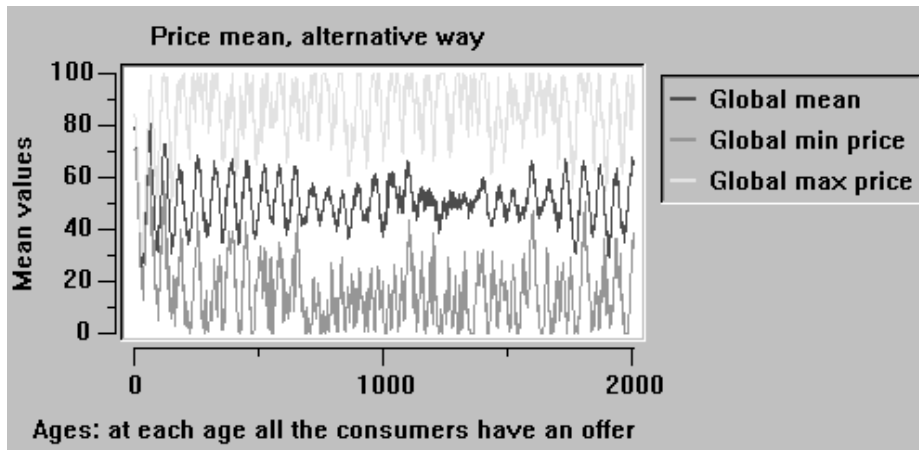


Fig. 7. L'esperimento inizia da una situazione non equilibrata, con bassa reattività e alta sensitività.

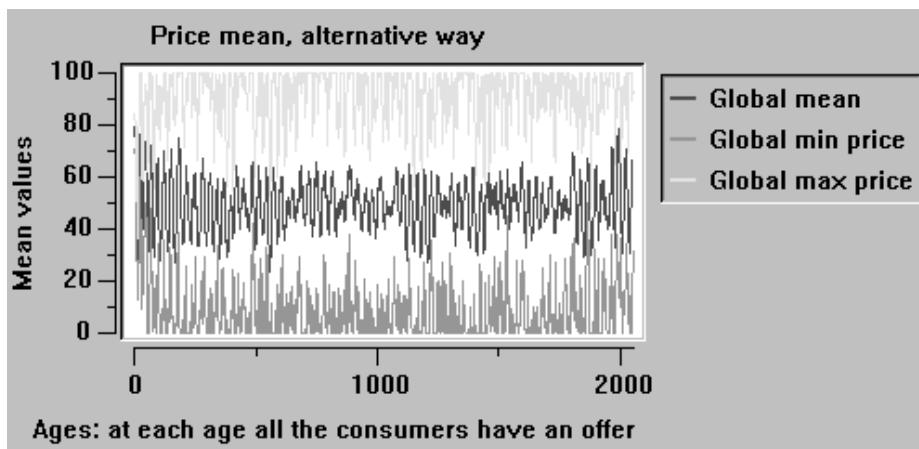


Fig. 8. L'esperimento inizia da una situazione non equilibrata, con alta reattività e bassa sensitività.

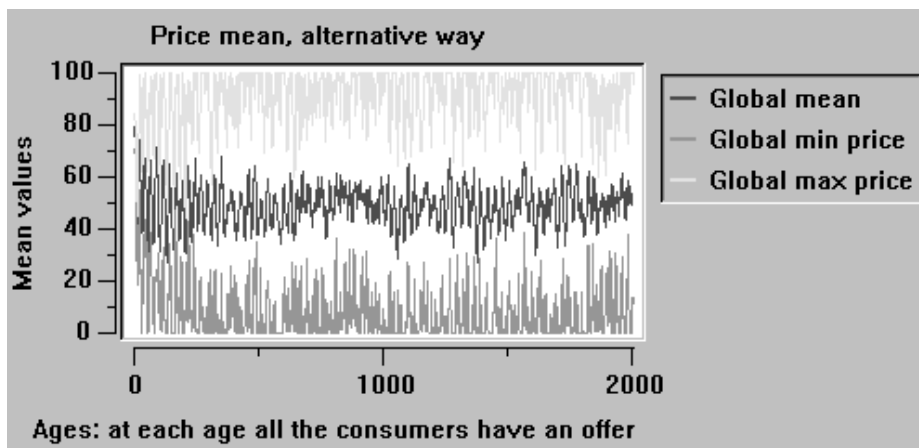


Fig. 9. L'esperimento inizia da una situazione non equilibrata, con alta reattività e alta sensitività.

Un commento generale a proposito della presenza del caso in questi esperimenti, anche per ricollegarci al programma hayekiano di analisi cui questo lavoro è dedicato.

Le serie riportate sono complesse e addirittura caotiche non per costruzione, ma come conseguenza della Emergenza di Dinamiche Caotiche sulla Base di Agenti (da cui il nome ABCDE attribuito a questa serie di prove). Ciò conferma la lezione di Hayek (vedere la lunga citazione riportata all'inizio) sulla finzione degli stati di equilibrio,

in quanto da agenti interattivi che potenzialmente cercano l'equilibrio può emergere, come questo modello mostra in un possibile caso, una serie di stati transitori senza temine, con l'apparenza di un andamento casuale.

4. Un esperimento di mercato senza regole esplicite a priori, fondato su reti neurali che apprendono

Rispetto al programma hayekiano proposto, l'esperimento precedente presenta dunque risultati di interesse, ma anche la debolezza della rigidità degli agenti.

Si introduce pertanto una impostazione esplicitamente connessionista, fondata sulla metodologia delle reti neurali, anche in riferimento alle fondamentali intuizioni del progetto hayekiano in questa direzione (Hayek, 1952).

4.1 Il metodo dei *Cross Target* (CT)

La tecnica Cross Target (CT) è sviluppata per costruire Agenti Artificiali Adattivi (AAA) senza fare necessariamente ricorso a regole economiche a priori. Si consideri lo strato di *output* di una Rete Neurale Artificiale (RNA) che rappresenta un agente, suddiviso in due parti, caratterizzate come: 1) nodi di uscita relativi ad azioni da compiere (congetture in merito); 2) nodi di uscita relativi agli effetti di tali azioni (sempre: congetture in merito). Con i CT, sia i *target* necessari per l'apprendimento della rete dal lato delle azioni, sia quelli necessari dal lato degli effetti, sono determinati in modo incrociato. I primi sono costruiti in coerenza con gli *output* della rete concernenti gli effetti (lato 2), allo scopo di sviluppare la capacità di decidere azioni che producano i risultati attesi. I secondi sono costruiti in coerenza con gli *output* della rete relativi alle congetture di azione (lato 1), con lo scopo di sviluppare la capacità dell'agente di stimare gli effetti delle azioni che sta decidendo.

L'idea guida è che un agente economico sviluppi con l'apprendimento la capacità di valutare in modo coerente: quale azione deve compiere per ottenere un risultato specifico; quali conseguenze derivano da un'azione data. La tecnica dei CT pone quindi al centro dell'attenzione il meccanismo di apprendimento con lo scopo essenziale di condurre esperimenti senza il sovraccarico di troppe ipotesi economiche a priori. Si può verificare che i CT possono riprodurre il comportamento di soggetti economici sulla base di un apparato interno spesso elementare o "ingenuo", ma con risultati che dall'esterno appaiono complessi. Per un osservatore, questo tipo di AAA apparentemente opera con piani e obiettivi, mentre internamente non ha nulla che possa essere riferito a tali entità. Le capacità richieste all'agente sono in linea con le caratteristiche proprie dei modelli a razionalità limitata, in quanto *æet* informativo e capacità computazionali.

In una RNA a tre strati, definiamo dunque gli *output* suddividendoli: 1) da un lato, in azioni che il soggetto compie o meglio congettura di compiere; 2) dall'altro, in congetture in merito agli effetti di quelle azioni. La fig. 10 descrive un AAA che agisce e apprende all'interno di uno schema CT. L'elemento centrale dell'algoritmo, che può essere interpretato come strumento di costruzione di modelli di comportamento senza uso di ottimizzazione, è una RNA, ma la costruzione - forse più complessa e meno neutrale - potrebbe anche fondarsi su altre metodologie.

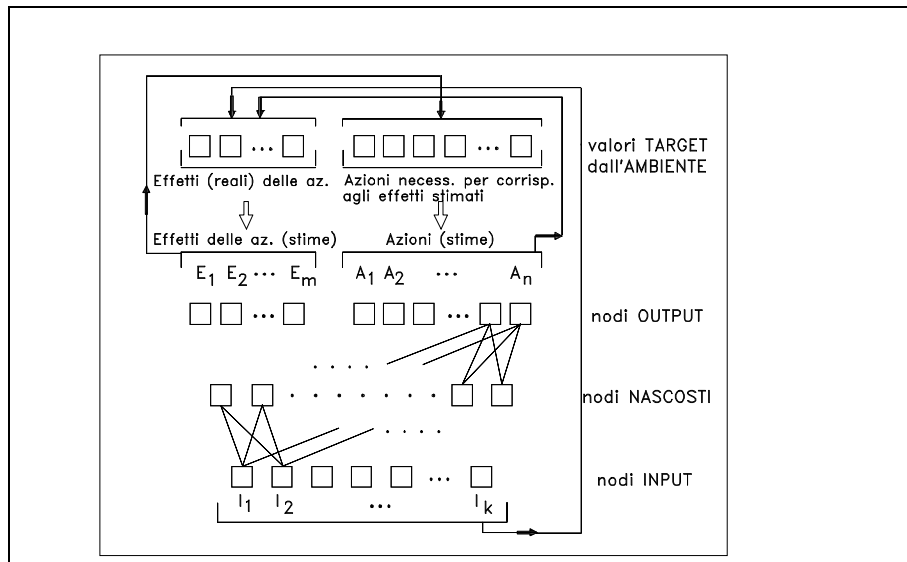


Fig. 10. Metodo dei *cross target*: interpretazione della RNA sottostante.

L'azione e l'apprendimento si svolgono in quattro fasi ogni "giorno"; un giorno è la somma degli eventi necessari per realizzare un ciclo completo di produzione degli *output*, valutazione degli errori, retropropagazione degli stessi, correzione dei pesi o parametri della RNA. I pesi sono all'inizio determinati a caso, equidistribuiti in un dato intervallo. Le quattro fasi sono le seguenti.

- 1) *Output* della RNA: sulla base dei valori di *input* e dei pesi (pesi con valori in quel particolare "giorno", essendo l'apprendimento continuamente ripetuto) la RNA valuta (fa congetture su) le azioni da compiere (lato destro della Fig. 10) e gli effetti di tali azioni (lato sinistro della Fig. 10).
- 2) *Target* per il lato sinistro della figura: i *target*, o valori che la rete deve imparare a riprodurre, per quel che riguarda le congetture sugli effetti sono costruiti sulla base delle azioni decise nell'altro lato della RNA. In questo modo le valutazioni sugli effetti diventano (tendono a diventare) coerenti con le valutazioni sulle azioni.
- 3) *Target* per il lato destro della figura: le differenze tra *target* e *output* della rete misurate nella fase 2) possono essere interpretate in modo inverso, come punto di partenza per la modificazione delle azioni al fine di avvicinarle alle congetture sugli effetti. Dunque il processo di avvicinamento è duplice: dagli effetti verso le azioni e viceversa, costruendo appunto i *target* in modo incrociato. Essendo spesso indeterminate le formule inverse necessarie per la costruzione dei *target* delle azioni, le correzioni sono distribuite in modo casuale tra tutti i *target* da costruire; inoltre, se più correzioni si riferiscono ad uno stesso *target*, si prende in considerazione soltanto la maggiore in modulo.
- 4) *Backpropagation*: si effettua infine l'apprendimento, correggendo i pesi della rete al fine di ottenere stime di effetti più prossime alle conseguenze delle azioni congettrate e congetture di azione più coerenti con le stime sugli effetti.

Con una opportuna variabilità in *input*, si ripetono le quattro fasi indicate per più cicli o "giorni", con un adattamento continuo della rete alle modificazioni ambientali. Analizzando dopo ogni "giorno" le matrici dei pesi, si rileva che la matrice che collega i nodi *input* a quelli *hidden* si modifica marginalmente, mentre il cambiamento più rilevante si concentra nella matrice che collega i nodi *hidden* a quelli *output*. La rete è quasi certamente sotto-adattata e di conseguenza l'agente simulato sviluppa una capacità solo locale di agire nell'ambiente.

Si può indicare l'apprendimento presentato sopra come apprendimento a breve termine, individuando per contro anche un apprendimento a lungo termine o apprendimento ripetuto, in analogia con la distinzione tra memoria a breve termine e memoria a lungo termine: 1) la fase di apprendimento e azione sin qui esaminata produce agenti che modificano continuamente i propri pesi, adattandosi in modo locale ai cambiamenti dell'ambiente; 2) *ex post* è possibile introdurre un nuovo apprendimento, sulla base dei dati storici ottenuti da quanto è via via accaduto durante la fase precedente; ne deriva una RNA in grado di reagire correttamente, senza più modificare i pesi, a cambiamenti di rilievo che intervengano nell'ambiente. Il secondo tipo di apprendimento può avvenire anche periodicamente, su un breve tratto di dati (ad esempio, ogni 50 giorni sui precedenti 100); oppure su un campione dei dati storici.

Con i CT si ottengono agenti che agiscono sulla base della coerenza interna prima delineata. Ciò è sufficiente a determinare lo sviluppo (autonomo) di micromeccanismi capaci, separatamente o congiuntamente, di determinare azioni complesse, in analogia a quanto illustrato nella letteratura sui *robot* connessionisti.

Per affinare l'utilità dei CT nella conduzione di esperimenti economici, offrendo al contempo agli sperimentatori strumenti per il controllo e la regolazione degli agenti, si introducono alcune sofisticazioni del metodo: 1) introduzione di obiettivi esterni (EO), per influenzare le stime degli effetti, come sopra definiti; 2) Introduzione di proposte esterne (EP), per influenzare le congetture relative alle azioni, sempre come sopra definite.

Gli EP e gli EO sono *target* esterni: gli EO sostituiscono quelli costruiti in modo incrociato, che sono però comunque calcolati al fine di disporre delle valutazioni necessarie per le correzioni dal lato delle azioni; invece gli EP rappresentano uno dei *target* a volta a volta disponibili per l'apprendimento dal lato delle azioni, tra cui è scelto il maggiore in modulo.

4.2 L'emergere di un mercato

Si introduce ora uno schema sperimentale che, fondato sui Cross Target, faccia esplicito uso di agenti che apprendono, per verificare le conseguenze della presenza, nei nostri modelli, della capacità di adattamento degli agenti. Gli agenti sono reti neurali; operiamo quindi in un ambito strettamente connessionistico.

Per realizzare questo tipo di esperimenti ho sviluppato - utilizzando come base la citata piattaforma *Swarm* - un complesso programma "contenitore", facilmente adattabile a contesti diversi; il programma generale, denominato *ct-bp* (da *backpropagation* e *cross target*) e l'applicazione specifica preparata per realizzare questa serie di esperimenti, denominata *ct-hayek*, utilizzano *Swarm* 1.3.1 o superiore e possono essere richiesti direttamente all'autore.

La configurazione dell'esperimento è la seguente.

1. La struttura di *input* della rete neurale che simula l'agente consumatore prevede le variabili *Expenditure0* (spesa del periodo precedente), *Requirement0* (fabbisogno del periodo precedente), *p0* (prezzo proposto dall'agente nel periodo precedente), *q0* (quantità proposta dall'agente nel periodo precedente).
2. La struttura di *input* della rete neurale che simula l'agente produttore prevede le variabili *Revenue0* (ricavo del periodo precedente), *ProductionStream0* (flusso produttivo del periodo precedente), *p0* (prezzo proposto dall'agente nel periodo precedente), *q0* (quantità proposta dall'agente nel periodo precedente).

3. Gli *output* della rete neurale che simula il consumatore e i relativi *target* sono: *actualP*, congettura sul prezzo di scambio (il *target* è dato semplicemente dal prezzo di scambio o da quello di scambio potenziale, vedi sotto); *actualQ*, congettura sulla quantità scambiata (il *target* è dato semplicemente dalla quantità scambiata, eventualmente 0); questi due primi valori sono effetti, nella terminologia dei CT, ma dipendenti anche dall'azione di un altro agente; non essendo in questo caso rilevante lo sviluppo di una capacità di previsione di tali valori, non partecipano ai *target* incrociati determinanti per l'apprendimento sulle azioni. *Expenditure*, spesa congetturata (il *target* è il prodotto di p e q); *Requirement*, fabbisogno di beni nell'intervallo di tempo (il *target* è *actualQ*); p e q , prezzo e quantità proposti per lo scambio, i cui *target* derivano da *Expenditure* e da *Requirement*, e quindi indirettamente da *actualQ*. Per i dettagli, vedere l'appendice.
4. Gli *output* della rete neurale che simula il produttore e i relativi *target* sono: *actualP* e *actualQ* (vedi sopra); *Revenue*, ricavo congetturato (il *target* è il prodotto di p e q); *ProductionStream*, flusso produttivo nell'intervallo di tempo (il *target* è *actualQ*); p e q , prezzo e quantità proposti per lo scambio, i cui *target* derivano da *Revenue* e da *ProductionStream*, e quindi indirettamente da *actualQ*. Per i dettagli, vedere l'appendice.
5. Operano 10 consumatori e 10 produttori (la produzione non è definita; semplicemente i produttori offrono la quantità prodotta), che si incontrano casualmente ogni giorno, come nel modello del paragrafo 3. Lo scambio è possibile se il prezzo proposto dal produttore è minore di quello proprio del consumatore; lo scambio in tal caso avviene al prezzo del produttore; la quantità è la minore tra quella del consumatore e quella del produttore; se lo scambio non avviene, il prezzo indicativo sarà quello del consumatore (serve a formare la media dei prezzi); la quantità sarà 0.
6. Sono anche introdotti degli obiettivi esterni (EO) con le misure e le modalità tecnicamente individuate nel codice riportato in appendice. In breve: ridurre *Expenditure* e mantenere costante *Requirement* (agendo se necessario sui prezzi, ricordando che: l'EO influenza l'*output*, l'azione inversa a CT deve avvicinare il *target* all'*output*, rendendo più facile o difficile l'acquisto effettivo *actualQ* via p . Simmetricamente, aumentare *Revenue* e mantenere costante *ProductionStream* (interpretabile come produzione compatibile con la struttura produttiva presente), agendo in tal caso in modo opposto al consumatore per quel che riguarda le correzioni di p .

Riprendendo la distinzione sulla durata e significato dell'apprendimento introdotta nel paragrafo 4.1, la simulazione è eseguita con modalità di apprendimento solo a breve termine (Fig. 11, 14, 17, 20, senza apprendimento ripetuto); o con apprendimento a lungo termine limitato (Fig. 12, 15, 18, 21) e di maggior peso (Fig. 13, 16, 19, 22).

Riprendendo la lunga citazione da Hayek riportata all'inizio, noi qui realizziamo agenti che tendenzialmente incorporano la proposizione empirica dell'esistenza di un equilibrio, che però non è né descritto, né misurato, né proceduralizzato in un "come si fa a raggiungerlo". Con la loro azione, ed eventualmente con l'apprendimento, che determina la capacità di valutare gli *output* sulla base degli *input*, sviluppiamo le situazioni seguenti.

I grafici si leggono interpretando le curve, dal basso verso l'alto, come: quantità media degli scambi (divisa per 10 per motivi di scala); prezzo minimo, medio e massimo degli scambi fatti o potenziali (prezzo del consumatore che non scambia, in tal caso).

Equilibrio spontaneo: compare nelle tre figure che seguono, in cui manca l'azione degli EO; in via interna, dati i valori congetturali medi che la rete neurale esprime all'inizio, la leggera tendenza all'aumento del prezzo può essere

spiegata sulla base di adeguamenti ai valori di *Requirement* e *ProductionStream* autogenerati. L'apprendimento provoca stabilizzazione e convergenza al mercato, ma se è limitato si possono generare situazioni di precario equilibrio, che generano scambi a valori molto lontani dalla media (es. Fig. 12, dopo l'ascissa 400); se è invece più approfondito (Fig. 13) può generare schemi molto complessi di comportamento, in particolare al di là dell'intervallo riportato nella figura.

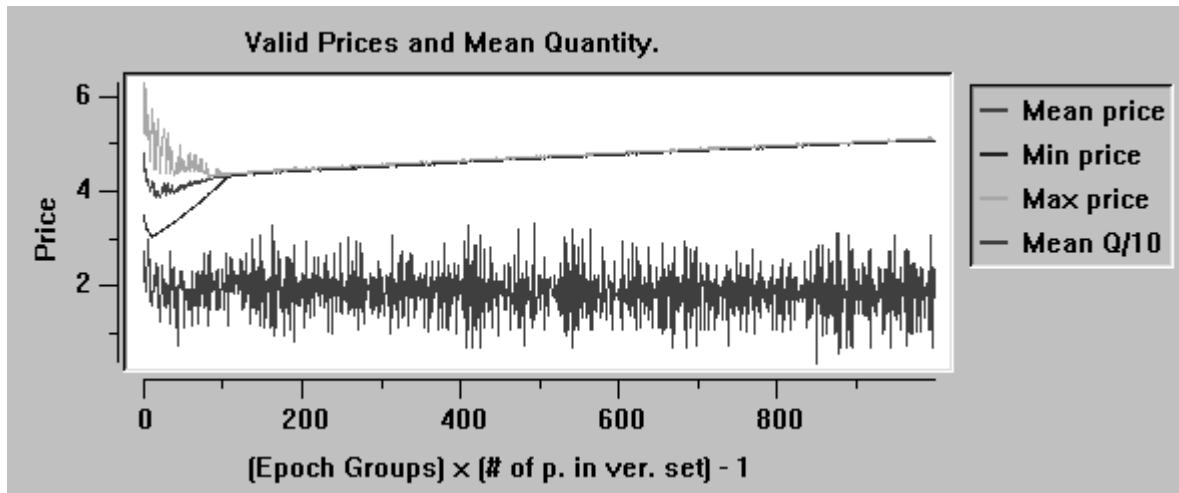


Fig. 11. Senza EO, senza apprendimento ripetuto.

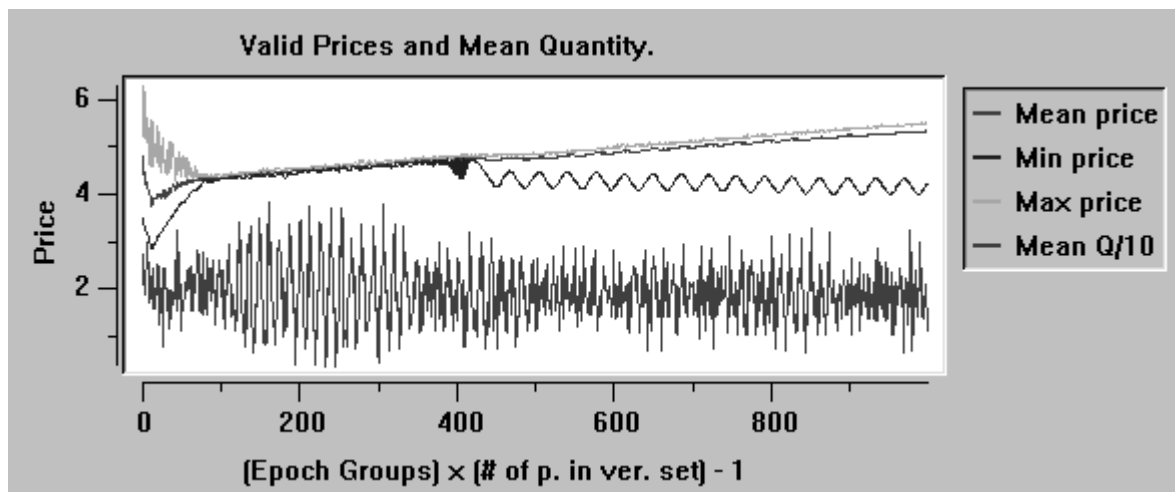


Fig. 12. Senza EO, con limitato apprendimento ripetuto.

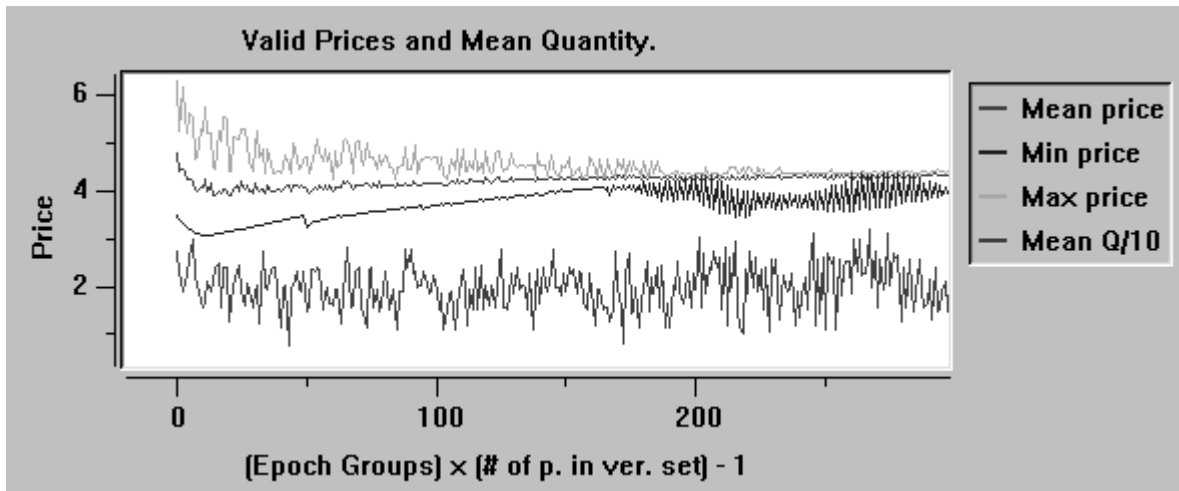


Fig. 13. Senza EO, con apprendimento ripetuto.

Equilibrio con azione esterna (EO) su spesa e ricavi: in questo caso (riportato nelle tre figure che seguono) il mercato si genera stabilmente, con prezzi costanti. Si rileva qualche segnale di diminuzione nei prezzi minimi (secondo caso) per la presenza di una asimmetria nel modello, stante il fatto che è il consumatore a decidere di non acquistare se il prezzo è alto, determinandone indirettamente la discesa (via *Requirement* e *ProductionStream*).

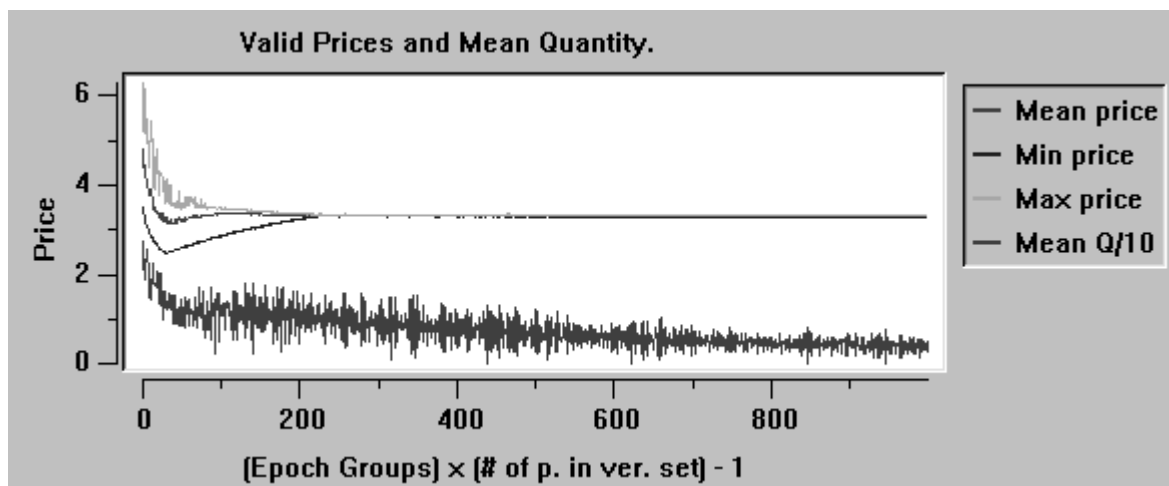


Fig. 14. Con EO su spesa e ricavi, senza apprendimento ripetuto.

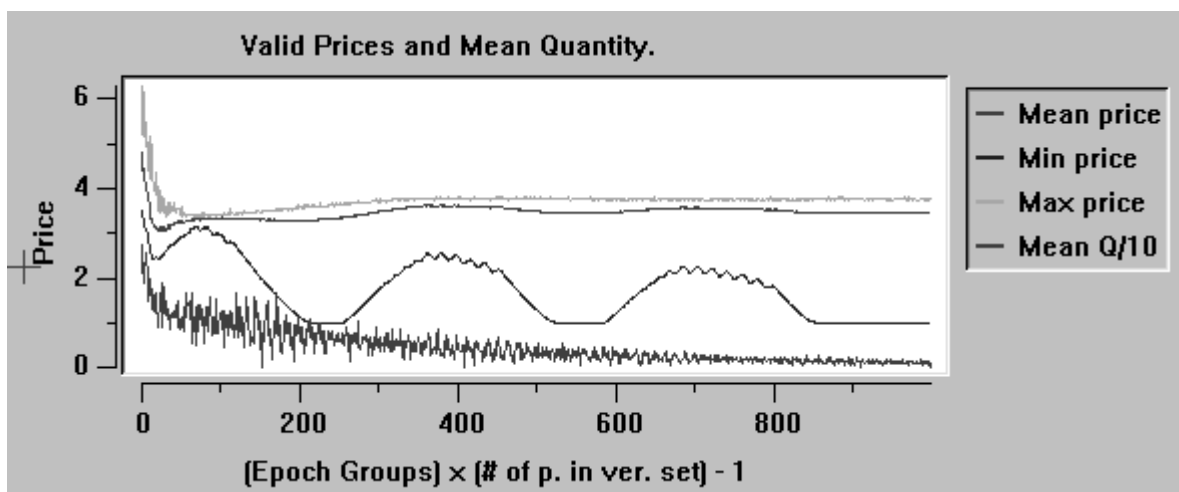


Fig. 15. Con EO su spesa e ricavi, con limitato apprendimento ripetuto.

La soluzione ad apprendimento più prolungata fa anch'essa presagire una diminuzione dei prezzi. Una nota tecnica: il modello è stato calibrato in modo che il primo di questi tre grafici riportasse un equilibrio a prezzo costante.

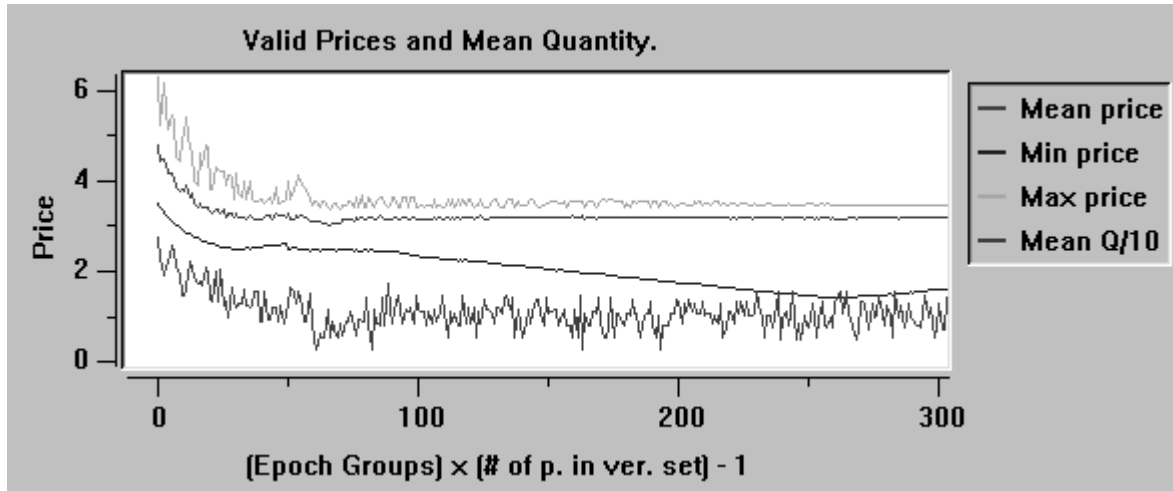


Fig. 16. Con EO su spesa e ricavi, con apprendimento ripetuto.

Equilibrio con azione esterna (EO) su fabbisogno e flusso produttivo.

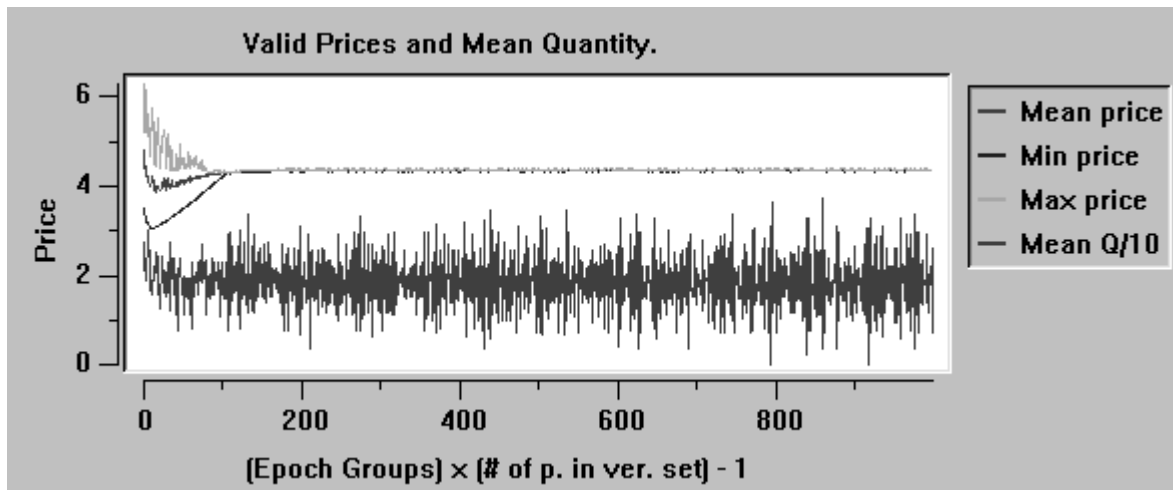


Fig. 17. Con EO su fabbisogno e flusso produttivo, senza apprendimento ripetuto.

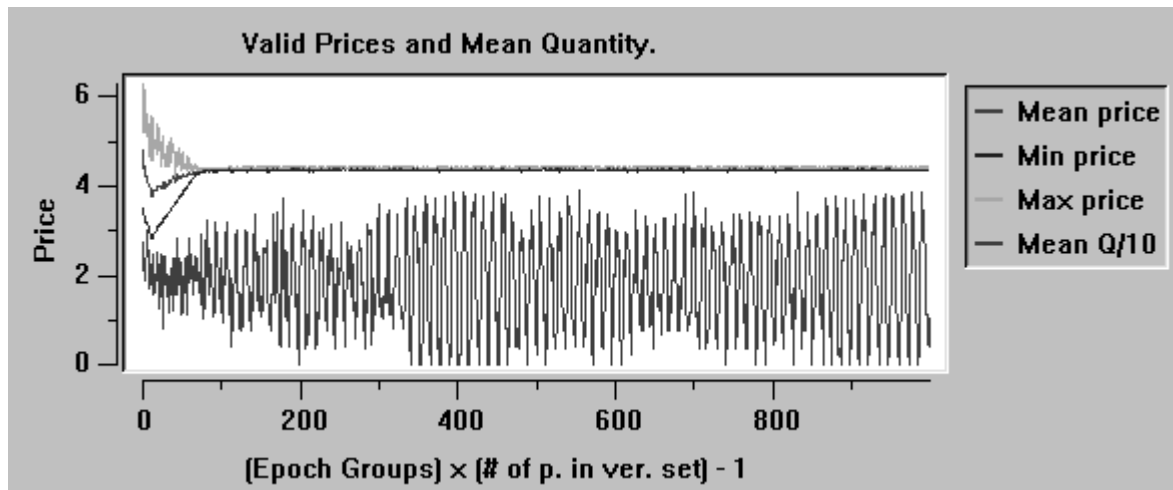


Fig. 18. Con EO su fabbisogno e flusso produttivo, con limitato apprendimento ripetuto.

Questo è il modello che più rapidamente converge all'equilibrio. Molto interessante il caso di apprendimento a lungo termine (Fig.19), dove pur con una più lenta convergenza, le oscillazioni delle quantità si ridimensionano.

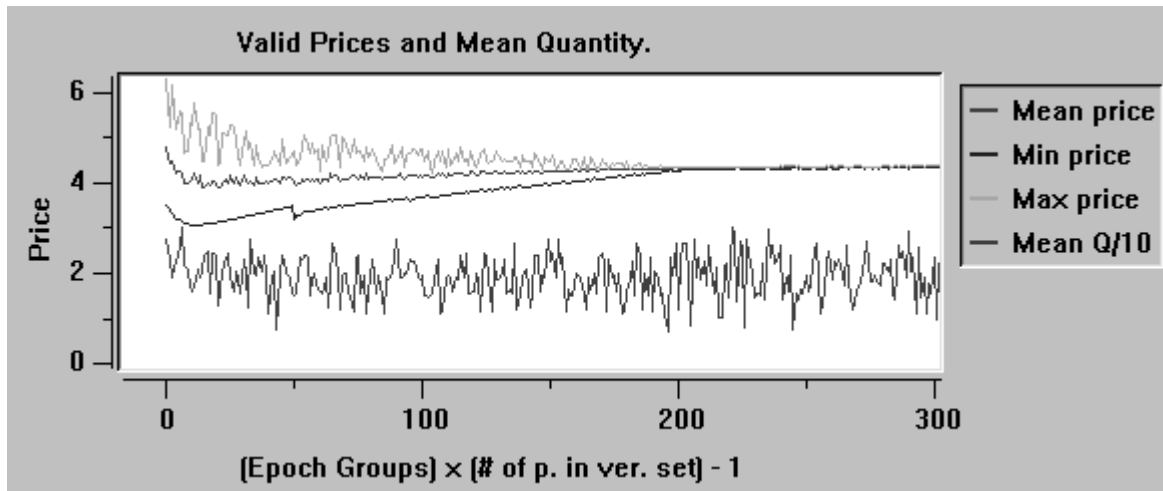


Fig. 19. Con EO su fabbisogno e flusso produttivo, con apprendimento ripetuto.

Equilibrio con azione esterna (EO) su spesa, ricavi, fabbisogno e flusso produttivo.

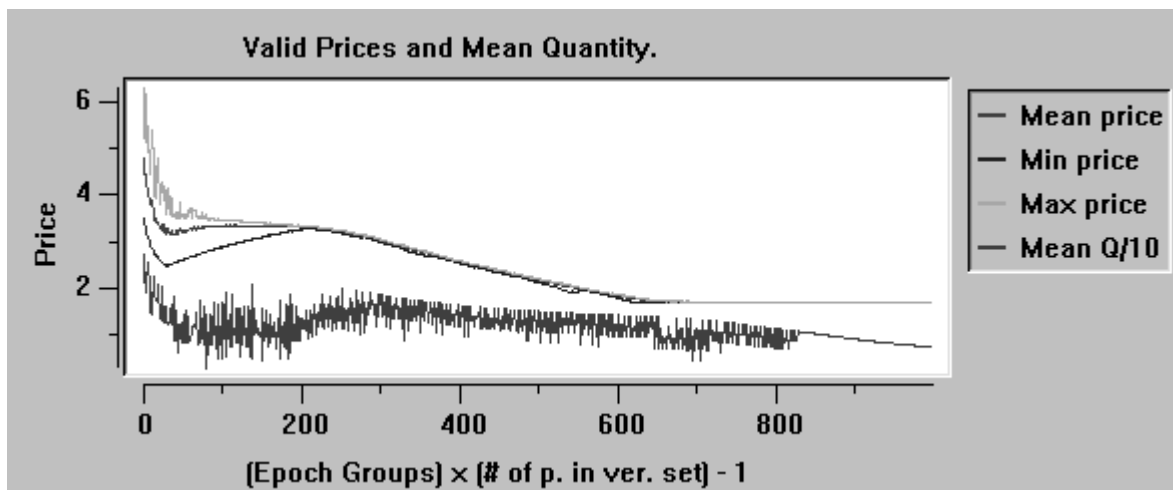


Fig. 20. Con tutti gli EO, senza apprendimento ripetuto.

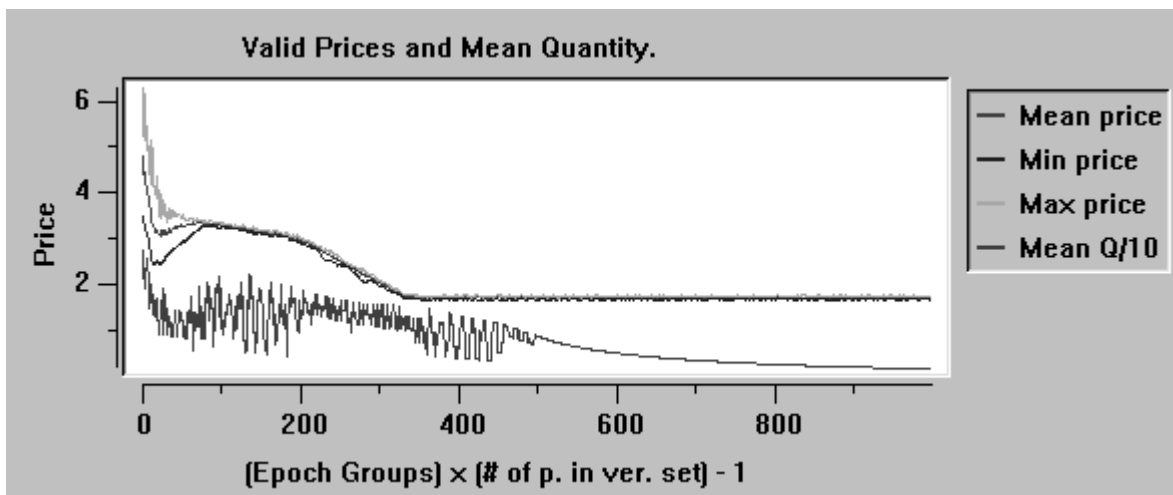


Fig. 21. Con tutti gli EO, con limitato apprendimento ripetuto.

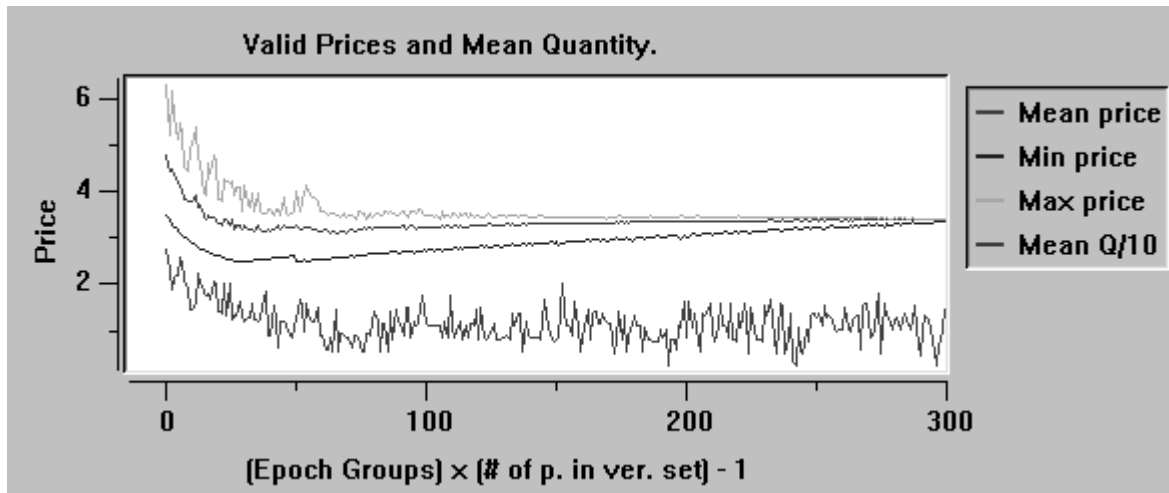


Fig. 22. Con tutti gli EO, con apprendimento ripetuto.

Nelle ultime tre figure, il mercato asimmetrico - per la presenza del consumatore che può non acquistare in presenza di un EO che gli impone di spendere di meno - influenza la riduzione del prezzo, ma ne consente la successiva stabilizzazione; evidentissimo l'effetto di stabilizzazione dato dall'apprendimento, in particolare nell'ultima figura.

5. Un commento a questo tipo di esperimenti

Il confronto tra i risultati del paragrafo 3 e del 4.2 non è ovvio né immediato: una costruzione deterministica genera un mercato complesso, con contenuti caotici e senza equilibrio; una costruzione intrinsecamente connessionista, via CT, genera situazioni di equilibrio, con un rilevante ruolo per l'apprendimento, che è comunque la ragione dell'adattamento dei nostri agenti artificiali anche nelle Fig. 11, 14, 17, 20. Ciò è dunque coerente con un programma di studi hayekiano, ma offre il fianco alla critica sul piano dell'adozione implicita di una qualche sorta di cedimento all'individualismo metodologico, là dove invece l'economia è sistema complesso e non solo somma di individui.

Per approfondire questo aspetto, possiamo introdurre alcune considerazioni problematiche conclusive, riprendendo Terna (1998c) e citando Epstein e Axtell (1996, Cap. 1, par. "Oltre l'individualismo metodologico"):

Il nostro punto di partenza nella costruzione di modelli fondati su agenti è l'individuo: attribuiamo delle regole di comportamento agli agenti e quindi facciamo procedere il sistema nel tempo, per scoprire quale struttura emerga a livello macroscopico. Questa scelta contrasta notevolmente con la rappresentazione fortemente aggregata della macroeconomia, della sociologia, di certi campi della scienza politica, in cui aggregati sociali, come classi e stati, sono definiti *ab initio*. In questo senso il nostro lavoro può essere a ragione etichettato come "individualismo metodologico". Comunque, noi condividiamo le posizioni di alcuni dei componenti del filone individualista se ed in quanto si ritenga che le strutture collettive o "istituzioni" che emergono possano avere effetti di retroazione nella popolazione di agenti, alterando il comportamento degli individui. I modelli fondati su agenti ci consentono di studiare le interazioni tra individui e istituzioni.

Come circoscriviamo il campo dell'individualismo metodologico e con quali contrapposizioni tra gli studiosi? Scegliamo una definizione neutrale, da Zamagni (1987), Cap. 1, par. 1.9, p.65:

La posizione secondo cui tutte le proposizioni circa i gruppi sono riconducibili a proposizioni circa il comportamento dei singoli facenti parte di quei gruppi e le loro interazioni, è oggi nota come *individualismo metodologico*, un'espressione coniata da Schumpeter (...).

Su questa posizione si concentrano critiche sia degli organicisti metodologici, che negano che sia possibile comprendere un sistema sociale o economico studiando i singoli individui che lo compongono; sia, all'opposto, di coloro che rifuggono dalle metodologie che comportano la scelta riduzionista della costruzione del così detto agente rappresentativo. Inoltre esiste la disputa relativa all'uso di agenti semplici come struttura e regole oppure, in alternativa, molto strutturati in termini di regole, capacità di ottimizzazione, conoscenza dei modelli dell'economia - paradossalmente, come nota Sargent (1993), non noti all'econometrico che li studia - e dei dati necessari per applicarli.

Le incomprensioni si manifestano quando chi nega interesse allo studio dell'agente rappresentativo sulla base dell'ipotesi che non si possa prescindere dallo studio della strutture (delle classi, delle istituzioni, ...) si contrappone anche a chi - costruendo modelli di agenti semplici - a sua volta rifiuta quella scelta riduzionista e cerca una alternativa alla definizione degli agenti così detti neo-classici, tanto complicati quanto necessario per spiegare la realtà economica dall'interno degli agenti stessi.

Si può invece lavorare nella direzione di ammettere esplicitamente il ruolo *ex ante* di istituzioni e strutture preesistenti e di considerare essenziale l'eterogeneità degli agenti (in linea di principio, semplici), per sperimentare la non linearità degli effetti aggregati dei loro comportamenti (cioè: il *tutto* che non necessariamente corrisponde alla somma delle *parti*) in presenza di interazione, con l'emergenza della complessità: è in ciò che consiste l'originalità della sperimentazione artificiale con modelli di simulazione fondati su agenti.

Individualismo metodologico e metodologie fondate su modelli di agenti non sono dunque coincidenti. Con i modelli fondati su agenti si studia l'emergenza di strutture, istituzioni e comportamenti a priori imprevisi o imprevedibili, sulla base di diversi livelli costruttivi e interpretativi appartenenti a schemi a più livelli, con interazione all'interno di ogni livello e tra livelli diversi: nulla vieta che un agente sia a sua volta costituito di parti; che il modello preveda l'interazione tra entità a loro volta costituite da agenti e così via. Quest'ultima è anzi la strada che rende fruttuoso l'impiego di queste metodologie in campo organizzativo.

Riassumendo con Kirman (1992):

La somma dei comportamenti di semplici individui economicamente plausibili può generare complesse dinamiche, mentre per costruire un singolo individuo il cui comportamento mostri la stessa dinamica può essere necessario fondarlo su caratteristiche ben poco naturali. Inoltre se si rifiuta (n.d.r. sulla base del modello) una particolare ipotesi di comportamento, non è chiaro se si sta realmente rifiutando l'ipotesi in questione o l'ipotesi addizionale che esista un solo individuo (n.d.r. rappresentativo).

Studiando l'emergenza di fenomeni si considera quindi la complessità come un processo a due vie, in cui l'interazione non lineare degli agenti produce effetti sociali e strutture sociali già emerse in precedenza o emergenti (via evoluzione, selezione genetica, co-determinazione) influenzano struttura e comportamento degli agenti.

Una definizione operativa di complessità, molto utile in questo contesto, è riportata in Kaneko (1998):

(...) occorre distinguere con cura tra “complesso” e “complicato”. Quest’ultimo è un sistema composto da una varietà di elementi, la cui suddivisione in parti sia difficile, ma in linea di principio possibile. Dall’altro lato, nei sistemi complessi, ci si deve confrontare con una situazione circolare in cui le parti possono essere capite solo tramite il tutto, benché il tutto, naturalmente, consista di parti.

In McIntyre (1998) si affronta il tema dell’emergenza della complessità, tenendo conto della prospettiva ontologica, in relazione a quel che la realtà è, rispetto alla prospettiva epistemologica, legata a ciò che della realtà conosciamo e sappiamo rappresentare. L’autore privilegia la spiegazione epistemologica, il che è corretto nella maggior parte dei casi, ma lascia aperto il problema del *reverse engineering* qui introdotto del terzo paragrafo. Ogniquale sia impossibile partire dal risultato desiderato per costruire gli agenti che lo riproducono, e quindi si sia effettivamente in presenza di funzioni agente-sistema non invertibili, ci si trova probabilmente di fronte a problemi complessi in sé e non in quanto rappresentati con tecniche non appropriate.

Schematizzando, anche provocatoriamente: no al riduzionismo dell’individualismo metodologico, che tenta di riprodurre la complessità del sistema direttamente negli agenti, complicando gli agenti all’estremo, sovraccaricandoli di ipotesi e capacità; no all’organicismo, che considera non praticabile la strada degli agenti perché non ritiene che esistano a quella scala strutture adatte a riprodurre la complessità del reale; sì ad agenti costruiti con parsimonia di mezzi, cercando nell’interazione e in strutture *ex ante* o emergenti la strada per produrre situazioni complesse paragonabili a quelle reali, da leggersi alla scala più adatta, che non necessariamente è quella del singolo agente.

Sempre schematizzando, proviamo anche a discriminare in modo operativo tra modelli complicati (o artificiosamente complessi) e complessi (ricordando la distinzione tra rappresentazione e ontologia come cause possibili della nostra percezione della complessità), sulla base della possibilità di affrontare o no il problema oggetto di studio invertendo le funzioni implicite o esplicite che determinano il funzionamento dell’ambiente simulato, partendo dal sistema per arrivare agli agenti. Qui la reversibilità, o la possibilità di *reverse engineering* nella costruzione degli agenti, diventa indice di non complessità del sistema.

Appendice

Codice che descrive la formazione dei valori CT del consumatore del modello riportato nel paragrafo 4.2.

```
// CT values

// ActualP
myP          = [self getP_out];
aProducerP   = [aProducerInterface getP_out];
//printf("\ncons %f %f\n\n",myP,aProducerP);
if(myP >= aProducerP)
{
actualP_target = aProducerP;
[aProducerInterface setActualP_target_via_CT: actualP_target];
}
else
{
actualP_target=myP;
[aProducerInterface setActualP_target_via_CT: myP];
}
if(actualP_target < [minmax R: 4 C: 0])actualP_target=[minmax R:4 C: 0];
if(actualP_target > [minmax R: 4 C: 1])actualP_target=[minmax R:4 C: 1];

// ActualQ
myQ          = [self getQ_out];
```

```

aProducerQ      = [aProducerInterface getQ_out];

if(myP >= aProducerP)
{
actualQ_target = myQ;
if (aProducerQ < myQ) actualQ_target = aProducerQ;
[aProducerInterface setActualQ_target_via_CT: actualQ_target];
}
else
{
actualQ_target = 0;
[aProducerInterface setActualQ_target_via_CT: actualQ_target];
}
if(actualQ_target < [minmax R: 5 C: 0])actualQ_target=[minmax R:5 C: 0];
if(actualQ_target > [minmax R: 5 C: 1])actualQ_target=[minmax R:5 C: 1];

// Expenditure
expenditure_target = myP * myQ;
if(expenditure_target < [minmax R: 6 C: 0])
    expenditure_target=[minmax R:6 C: 0];
if(expenditure_target > [minmax R: 6 C: 1])
    expenditure_target=[minmax R:6 C: 1];

expenditure0 = [self getExpenditure0];
//printf("\n\n%f\n\n",expenditure0);
expenditureEO_target = expenditure0 * 0.98;
if(expenditureEO_target < [minmax R: 6 C: 0])
    expenditureEO_target=[minmax R:6 C: 0];
if(expenditureEO_target > [minmax R: 6 C: 1])
    expenditureEO_target=[minmax R:6 C: 1];

// Requirement (daily r.), set equal to the bought quantity; the goal
// of this effect is to use an EO
requirement_target = actualQ_target;
if(requirement_target < [minmax R: 7 C: 0])
    requirement_target=[minmax R:7 C: 0];
if(requirement_target > [minmax R: 7 C: 1])
    requirement_target=[minmax R:7 C: 1];

actualRequirement=20;
requirementEO_target = actualRequirement;
if(requirementEO_target < [minmax R: 7 C: 0])
    requirementEO_target=[minmax R:7 C: 0];
if(requirementEO_target > [minmax R: 7 C: 1])
    requirementEO_target=[minmax R:7 C: 1];

// P
expenditure_out=[self getExpenditure_out];
correction1 = ((expenditure_out-expenditure_target)/2.)/myQ;

requirement_out=[self getRequirement_out];
correction2=0;
if(requirement_out < requirement_target) correction2= -0.15;
if(requirement_out > requirement_target) correction2= 0.15;

correction = correction1;
if(fabs(correction2) > fabs(correction))correction=correction2;

p_target = myP + correction;
if(p_target < [minmax R: 8 C: 0])p_target=[minmax R:8 C: 0];
if(p_target > [minmax R: 8 C: 1])p_target=[minmax R:8 C: 1];
//printf("\nC p_target %f %f %f",p_target,correction1,correction2);

// Q
correction1 = ((expenditure_out-expenditure_target)/2.)/myP;
correction2=0;
// if(requirement_out < requirement_target) correction2= -2;
// if(requirement_out > requirement_target) correction2= 2;

correction = correction1;
if(fabs(correction2) > fabs(correction))correction=correction2;

q_target = myQ + correction;

```

```

if(q_target < [minmax R: 9 C: 0])q_target=[minmax R:9 C: 0];
if(q_target > [minmax R: 9 C: 1])q_target=[minmax R:9 C: 1];
//printf("\nC q_target %f %f %f",q_target,correction1,correction2);

```

Codice che descrive la formazione dei valori CT del produttore del modello riportato nel paragrafo 4.2.

```

// CT values

// ActualP
// coming from consumer, as actualP_target
//printf("prod %f\n\n",actualP_target);
if(actualP_target < [minmax R: 4 C: 0])actualP_target=[minmax R:4 C: 0];
if(actualP_target > [minmax R: 4 C: 1])actualP_target=[minmax R:4 C: 1];

// actualQ
// coming from consumer, as actualQ_target
if(actualQ_target < [minmax R: 5 C: 0])actualQ_target=[minmax R:5 C: 0];
if(actualQ_target > [minmax R: 5 C: 1])actualQ_target=[minmax R:5 C: 1];

// Revenue
myP=[self getP_out];
myQ=[self getQ_out];
revenue_target = myP * myQ;
if(revenue_target < [minmax R: 6 C: 0])revenue_target=[minmax R:6 C: 0];
if(revenue_target > [minmax R: 6 C: 1])revenue_target=[minmax R:6 C: 1];

revenue0=[self getRevenue0];
//printf("\n\n%f\n\n",revenue0);
revenueEO_target = revenue0*1.02;
if(revenueEO_target < [minmax R: 6 C: 0])revenueEO_target=[minmax R:6 C: 0];
if(revenueEO_target > [minmax R: 6 C: 1])revenueEO_target=[minmax R:6 C: 1];

// ProductionStream (daily p.), set equal to the sold quantity; the goal
// of this effect is to use an EO
productionStream_target = actualQ_target;
if(productionStream_target < [minmax R: 7 C: 0])
    productionStream_target=[minmax R:7 C: 0];
if(productionStream_target > [minmax R: 7 C: 1])
    productionStream_target=[minmax R:7 C: 1];

actualProductionStream=20;
productionStreamEO_target = actualProductionStream;
if(productionStreamEO_target < [minmax R: 7 C: 0])
    productionStreamEO_target=[minmax R:7 C: 0];
if(productionStreamEO_target > [minmax R: 7 C: 1])
    productionStreamEO_target=[minmax R:7 C: 1];

// P

revenue_out=[self getRevenue_out];
correction1 = ((revenue_out-revenue_target)/2.)/myQ;

productionStream_out=[self getProductionStream_out];
correction2=0;
if(productionStream_out < productionStream_target) correction2= 0.15;
if(productionStream_out > productionStream_target) correction2= -0.15;

correction = correction1;
if(fabs(correction2) > fabs(correction))correction=correction2;

p_target = myP + correction;
if(p_target < [minmax R: 8 C: 0])p_target=[minmax R:8 C: 0];
if(p_target > [minmax R: 8 C: 1])p_target=[minmax R:8 C: 1];
//printf("\nP p_target %f %f %f",p_target,correction1,correction2);

// Q

correction1 = ((revenue_out-revenue_target)/2.)/myP;

correction2=0;
// if(productionStream_out < productionStream_target) correction2= 2;///? <0
// if(productionStream_out > productionStream_target) correction2= -2;///? >0

```

```

correction = correction1;
if(fabs(correction2) > fabs(correction))correction=correction2;

q_target = myQ + correction;
if(q_target < [minmax R: 9 C: 0])q_target=[minmax R:9 C: 0];
if(q_target > [minmax R: 9 C: 1])q_target=[minmax R:9 C: 1];
//printf("\nP q_target %f %f %f",q_target,correction1,correction2);

```

Bibliografia

- EPSTEIN J.M. e AXTELL R. (1996), *Growing Artificial Societies - Social Science from the Bottom Up*, Cambridge MA, MIT Press.
- HAYEK F.A. (1937), Economics and Knowledge, in *Individualism and Economic Order*, London e Henley, Routledge e Kegan Paul, 1949; trad. it. Conoscenza e economia, in *Problemi di filosofia della scienza economica*, Roma, La Nuova Italia Scientifica, 1983; ripreso in F.A. von Hayek, *Conoscenza, Mercato e Pianificazione*, Bologna, Il Mulino, 1988, pp.227-252.
- HAYEK F.A. (1952), *The Sensory Order*, Chicago, The University of Chicago Press.
- KANEKO K. (1998), Life as Complex System: Viewpoint from Intra-Inter Dynamics. *Complexity*, 6, pp.53-63.
- KIRMAN A. (1992), Whom or What Does the Representative Agent Represent?. *Journal of Economic Perspectives*, 6, pp.126-139.
- MCINTYRE L. (1998), Complexity: A Philosopher's Reflection. *Complexity*, 6, pp.26-32.
- RASMUSSEN S. e BARRET C. L. (1995), Elements of a Theory of Simulation, in F. Moran e al. (a cura di), *Advances in artificial life: Third European Conference on Artificial Life*, Lecture notes in computer science 929, Berlino, Springer, (reperibile anche a <http://www.santafe.edu/sfi/publications/95wplist.html>).
- SARGENT T. J. (1993), *Bounded Rationality in Macroeconomics*, Oxford, Clarendon Press.
- TERNA P. (1998a), Simulation Tools for Social Scientists: Building Agent Based Models with SWARM, *Journal of Artificial Societies and Social Simulation*, vol. 1, no. 2, <<http://www.soc.surrey.ac.uk/JASSS/1/2/4.html>>
- TERNA P. (1998b), ABCDE: Agent Based Chaotic Dynamic Emergence, in J.Sichman, R.Conte e N.Gilbert (a cura di), *Multi-Agent Systems and Agent-Based Simulation*, LNAI series, vol. 1534, Berlin, Springer-Verlag.
- TERNA P. (1998c), Creare mondi artificiali: una nota su Sugarscape e due commenti, *Sistemi Intelligenti*, no. 3.
- ZAMAGNI S. (1987), *Economia politica - Teoria dei prezzi, dei mercati e della distribuzione*, Roma, NIS.