

PIETRO TERNA*

SIMULAZIONE AD AGENTI IN CONTESTI DI IMPRESA

Nelle scienze sociali, la simulazione ad agenti, dal basso, come è ora largamente praticata, ha poco più di dieci anni di vita. Nel 1990 un articolo che gettava un ponte tra pensiero di Hayek ed elaborazione automatica decentrata (Lavoie et al., 1990) proponeva (p.135) come innovative - e lo erano - le considerazioni che seguono:

(. . .) lo scopo sarebbe quello di stabilire i vincoli, specificare l'ambiente istituzionale o le regole di decisione per gli agenti e poi eseguire la simulazione per vedere che cosa accade. L'idea non è quella di creare un modello matematico che ha le proprie conclusioni implicite nelle premesse. Invece, è quella di eseguire la simulazione come un esperimento mentale, dove ciò che è interessante non sono tanto i risultati finali, ma il modo in cui il processo funziona. E noi, i programmatori, non sapremo come il processo finirà per risultare fino a che non avremo eseguito l'esperimento mentale. L'ordine dovrebbe emergere non dal disegno del programmatore, ma dall'interazione spontanea delle parti componenti.

Del brano ci colpiscono il carattere anticipatore, ma anche la formulazione condizionale, come di chi ancora si chiede se ciò che propone è fattibile. Ora sappiamo che è fattibile e che rappresenta un cambiamento fondamentale nel metodo (Parisi, 2001); inoltre raccoglie una gran messe di lavori (Tsfatsion, 2001). Abbiamo anche due riviste elettroniche specializzate: il *Journal of Artificial Societies and Social Simulation*, JASSS (jasss.soc.surrey.ac.uk) e l'*Electronic Journal of Evolutionary Modeling and Economic Dynamics*, e-JEMED (www.e-jemed.org).

E' però anche indubitabile che i risultati sono al momento ancora insufficienti a costituire massa critica per il riconoscimento della validità del metodo al di fuori della schiera degli specialisti. Una forte critica interna è quella di Pryor (2000), in un lavoro auto-ironico: guardando indietro, uno sconosciuto autore del 2028 (quando un asteroide sarà caduto sulla Terra), osserva che "in un tipico libro sulla complessità della fine degli anni '90 (. . .) quasi tutti i lavori non contengono applicazioni empiriche reali, a parte qualche interessante aneddoto".

L'interdisciplinarietà metodologica, che va dalla simulazione alla scienza della complessità, rappresenta un altro problema rilevante: è ben evidente che i lavori di carattere interdisciplinare ed eterodossi hanno meno peso di quelli legati al nucleo delle varie discipline, nell'ortodossia.

* pietro.terna@unito.it

Soprattutto è urgente ottenere risultati significativi e utili per il mondo reale. Anche per questo è utile indirizzare una parte degli sforzi della ricerca verso i concreti problemi di gestione della conoscenza e di organizzazione nelle unità produttive (aziende).

L'applicazione della metodologia della simulazione ad agenti, ad esempio fondata sulla piattaforma di sviluppo *Swarm* <www.swarm.org>, si estende infatti anche alle applicazioni aziendali, in particolare per lo studio delle catene di fornitura (Lin e al., 2000; Schlueter-Langdon, 2000).

La costruzione di modelli di simulazione, fondati su agenti autonomi interattivi e applicati alla realtà aziendale, conduce alla predisposizione di strutture virtuali relative a parti di strutture produttive o al loro insieme. Strutture in cui siano riprodotte le regole di funzionamento e di interazione interne, con agenti che non solo operino, ma stabiliscano come essi stessi o altri agenti possano o debbano operare.

L'azienda virtuale così ottenuta può essere oggetto di indagine e di sperimentazione, esattamente come se si trattasse di una struttura riprodotta in laboratorio.

Al contempo, si rappresenta la conoscenza in azienda, utilizzando il formalismo del terzo tipo, all'interno della classificazione che comprende (i) i modelli letterario-descrittivi, (ii) quelli matematico-statistici e infine (iii) quelli realizzati (con la finalità della simulazione) con codice informatico (Gilbert e Terna, 2000).

La rappresentazione della conoscenza, spesso implicita o tacita, non solo ne consente la trasmissione, ma permette di introdurre livelli sofisticati di simulazione relativi al diverso contenuto o diverso uso della conoscenza disponibile o acquisibile dall'esterno.

Con queste premesse, introduciamo: nel Paragrafo 1, una presentazione del sistema di simulazione che utilizziamo per le applicazioni aziendali, ricordando che la valenza dell'oggetto *software* in questione è molto più ampia; nel Paragrafo 2, uno schema generale di impostazione dei modelli di agenti; nel Paragrafo 3, alcuni esempi di applicazione delle metodologie ad agenti in contesti aziendali; nel Paragrafo 4, lo strumento jVE (java Virtual Enterprise), vale a dire una metodologia di simulazione ad agenti per l'azienda; nel Paragrafo 5, la finalizzazione teorica del lavoro empirico proposto.

1. UNO STRUMENTO PER LA SIMULAZIONE

Per le applicazioni delineate in premessa, la base informatica utilizzata da chi scrive è, da alcuni anni, *Swarm*.

Il progetto *Swarm*, nome che sta per sciame (di consumatori, aziende, unità produttive, ma anche insetti, pesci, molecole ...) è nato, a metà anni '90, all'interno del *Sante Fe Institute* <www.santafe.edu> per realizzare una *lingua franca* della simulazione ad agenti. Si tratta non tanto di un programma, quanto di una biblioteca di funzioni, sviluppare tramite classi e metodi, secondo l'impostazione della programmazione a oggetti, che utilizza classi astratte per realizzare esemplari (*instance*) specifici (nella Figura 1: gli insetti, lo spazio del cibo, . . .); esemplari che reagiscono a ordini/messaggi (i metodi). Il tutto corredato da un protocollo utile per

utilizzare le classi in modo ben formato, scrivendo in Objective C, e più recentemente in Java, le proprie applicazioni.

Da <sourceforge.net/projects/swarm> leggiamo che “Swarm è un nucleo informatico e una biblioteca di codici informatici per la simulazione multi-agente di sistemi complessi. L’architettura di base di *Swarm* consiste in una collezione di agenti interattivi in modo parallelo: all’interno di tale architettura può quindi essere sviluppata una varietà molto ampia di modelli basati su agenti.”

1.1 Struttura di una applicazione Swarm

La tipica struttura di una applicazione scritta in *Swarm* prevede la presenza di due livelli. Si veda la Figura 1 in cui sono indicati (i) gli oggetti che devono essere costruiti all’interno del modello; nel nostro esempio: un reticolo su cui far muovere gli oggetti della simulazione; uno spazio in cui sarà codificata la presenza o assenza di qualche caratteristica (il cibo, nel nostro esempio); la classe degli agenti della simulazione, qui semplici insetti che si muovono a caso e utilizzano il cibo, da cui far discendere gli esemplari di insetti inclusi in un lista cui - grazie al protocollo *Swarm* che prevede che i messaggi possano essere indifferentemente inviati agli oggetti o alle liste che li contengono - invieremo il comando di compiere un passo.

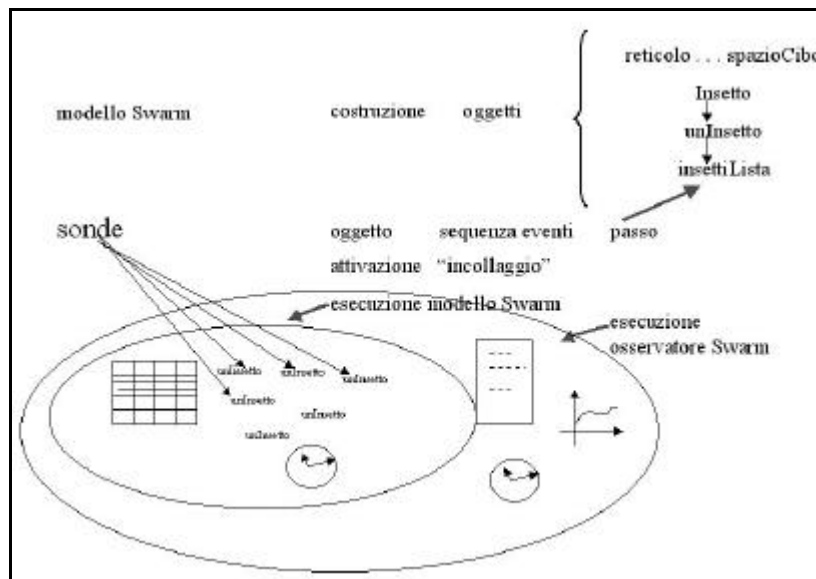


Fig. 1 – Una struttura esemplificativa di una simulazione con *Swarm*

Nella Figura 1 è anche indicata (ii) la costruzione di oggetti relativi alla gestione del tempo e cioè le sequenze degli eventi che, all’interno dell’esperimento, devono succedersi; infine (iii) un passaggio di caricamento congiunto degli oggetti che rappresentano gli agenti della simulazione e di quelli che gestiscono gli eventi del tempo (“incollati” in memoria). Infine, la definizione di due livelli di astrazione che consentono: (a) di gestire l’osservatore - “noi” che, con opportuni strumenti quali grafici,

tabelle, istogrammi, osserviamo l'effetto del scorrere del tempo all'interno del modello, con i relativi accadimenti - e (b), ad un livello subordinato nella scala dei tempi anche se autonomo dal punto di vista del contenuto, il funzionamento dell'esperimento vero e proprio.

Swarm consente sia di guardare al funzionamento del modello dal livello dell'osservatore, sia di entrare direttamente all'interno dei singoli oggetti che popolano la simulazione, grazie a sonde che, sviluppate automaticamente con la costruzione degli oggetti, permettono di tenerne sotto controllo variabili e funzioni via via durante il funzionamento.

1.2 La gestione del tempo

Nella Figura 1 sono anche rappresentati due orologi, uno relativo all'ambiente dell'osservatore, l'altro relativo all'ambiente dell'esperimento; gli orologi possono gestire il tempo con incrementi definiti secondo le esigenze della simulazione (all'interno dell'unità di tempo possono essere definiti da 1 a n passi, cui corrispondono o no eventi, quindi anche passi "vuoti") e sono automaticamente sincronizzati grazie alle proprietà che ereditano in quanto derivati dalla biblioteca di classi *Swarm*.

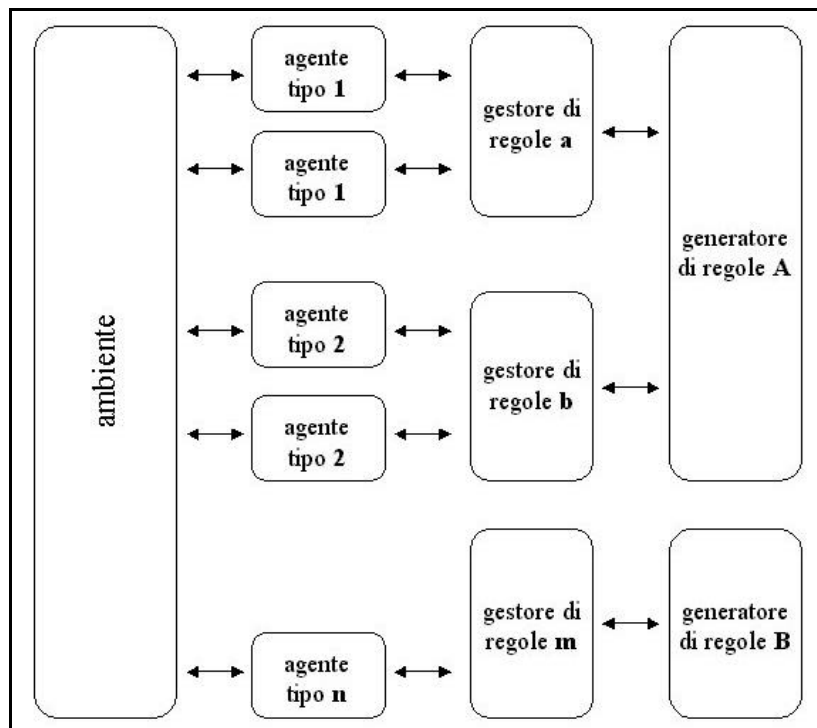


Fig. 2 - Schema ERA, Environment-Rules-Agents

Ciò semplifica moltissimo uno degli aspetti più complessi della scrittura dei programmi, per la conduzione di esperimenti di simulazione, vale a dire la gestione sincronizzata e ben coordinata degli eventi nel tempo.

2. UNO SCHEMA ASTRATTO PER L'IMPOSTAZIONE DEI MODELLI DI AGENTI

Come costruire gli oggetti che rappresentano gli agenti all'interno delle nostre simulazioni? La proposta dello schema ERA, *Environment-Rules-Agents* (Terna, 2000)¹, riportato nella Figura 2, è quella di gestire quattro diversi strati nella costruzione del modello e degli agenti.

(i) Un primo strato rappresenta l'ambiente in cui gli elementi sono chiamati ad interagire. All'interno del protocollo di *Swarm* corrisponde al Modello *Swarm* della Figura 1 (normalmente la classe è definita *ModelSwarm*), vale a dire il contesto all'interno del quale si definiscono gli agenti, se ne strutturano le liste, si individuano gli eventi nel tempo, si chiariscono le regole di interazione tra gli agenti grazie ai metodi (interpretabili come messaggi che gli agenti sono in grado di gestire, anche reagendo con azioni e informazioni) definiti all'interno degli oggetti creati dal Modello.

(ii) Un secondo strato è appunto quello degli agenti, che possono essere costruiti come esemplari di una o di più classi, a loro volta generate ereditando proprietà, caratteri, dati e metodi da classi più generali.

(iii) Il terzo strato gestisce le modalità attraverso cui gli agenti decidono il proprio comportamento. Ad ogni scelta, l'agente interroga un oggetto sovraordinato, definito gestore di regole (classi dette *RuleMaster*), comunicandogli i dati necessari ed ottenendo le indicazioni di azione.

(iv) Il quarto strato tratta la costruzione delle regole. Esattamente come gli agenti interrogano i gestori di regole, i gestori di regole interrogano i generatori di regole (classi dette *RuleMaker*) per modificare la propria linea di azione.

2.1 Modularità del codice informatico

Questo schema rappresenta sia un tentativo di rendere rigorosamente ordinato il codice informatico di una simulazione, sia una scelta nella direzione della modularità.

Se il codice è stato scritto secondo le modalità descritte, è infatti agevole sostituire a volta a volta gestori di regole con caratteristiche diverse, semplicemente sostituendo gli oggetti introdotti nel modello. Ad esempio: un sistema a regole fisse; una rete neurale; un algoritmo genetico, un sistema classificatore.

A loro volta reti neurali, algoritmi genetici, sistemi classificatori ecc. avranno la necessità di ricorrere a generatori di regole, differenziati a seconda della loro rispettiva tipologia: ciò sarà facilitato proprio dalla modularità del disegno adottato.

2.2 Metodologia aperta

Un'importante precisazione metodologica riguarda il quarto strato, indicato in (iv) sopra: i produttori di regole, e in parte i gestori di regole dello strato (iii), sono nient'altro che codice informatico.

Pertanto è possibile sviluppare ogni tipo di agente che possa essere scritto informaticamente; ciò vale in particolare per gli agenti cosiddetti

¹ Vedere anche <web.econ.unito.it/terna/ct-era/ct-era.html>.

cognitivi o BDI (*Beliefs, Desires, Intentions*), cioè agenti costruiti in modo che esprimano comportamenti correlati a un protocollo di comportamento, cui si attribuiscono significati.

Si può anche andare oltre: *Swarm*, integrato nella mia proposta con ERA, è il “guscio” che fa interagire gli agenti; la scrittura degli agenti può anche essere fondata su un motore inferenziale *à la* Prolog, che direttamente colleghi una eventuale impostazione dichiarativa della programmazione degli agenti al guscio che gestisce l’interazione e l’ambiente.

3. ESEMPI DI APPLICAZIONE DI MODELLI AD AGENTI IN CONTESTI AZIENDALI

L’applicazione a contesti aziendali delle tecniche di simulazione ad agenti - utilizzando la struttura *Swarm* descritta nel Paragrafo 1 e seguendo le indicazioni metodologiche del Paragrafo 2 - rappresenta un campo particolarmente promettente per i risultati applicativi e per le prospettive teoriche. Un interesse aggiuntivo sta nel fatto che simultaneamente si devono affrontare problemi di acquisizione della conoscenza (per fotografare come opera una determinata azienda) e di costruzione del modello dell’azienda, con gli effetti di quella conoscenza.

3.1 *I casi aziendali*

Le applicazioni di simulazione aziendale su cui si è fatta o si sta facendo esperienza sono al momento: due concluse; una in fase di notevole sviluppo; due in avvio. I diversi casi sono qui di seguito individuati con delle sigle, senza riferimento esplicito alle aziende: approfondimenti specifici per finalità di ricerca sono comunque possibili, con il consenso degli interessati.

3.1.1 Caso A, settore abbigliamento

Il caso A è quello di una azienda del settore dell’abbigliamento, fortemente legata al ciclo della moda, con notevoli problemi di previsione, in quanto la campagna di vendita si sovrappone alla emissione degli ordini dei semilavorati e alla produzione e i dati via via raccolti dalle diverse aree di mercato sono molto difforni.

La difficoltà principale della simulazione sta in questo caso nel riprodurre, e poi razionalizzazione, la sequenza che va dal lancio dei campionari, alla raccolta dei dati, alla previsione e alla loro utilizzazione. Si tratta quindi di una modellizzazione di una parte della conoscenza utilizzata nell’azienda.

Lo scopo della struttura ad agenti è stato quello di formalizzare un sistema ordinato di relazioni aziendali, sia interne, sia esterne con i rappresentanti operanti sul territorio; nonché quello di rendere sistematici i dati e le metodologie utilizzate per produrre le previsioni.

Il riordino delle metodologie di previsione e di raccolta dei dati ha poi consentito di utilizzare, pure in presenza di serie storiche molto brevi, previsori sofisticati, quali funzioni a rete neurale applicati su serie di tempo. Si sono misurati miglioramenti delle capacità previsionali, anche se ancora insoddisfacenti, in quanto le previsioni stesse sono state realizzate su

famiglie aggregate di prodotti, non sufficientemente micro-individuati tanto da consentire l'emissione di ordini per specifiche sottocategorie di prodotto.

Si è trattato comunque di una esperienza molto interessante ed importante che si rifletterà su una delle applicazioni ora in avvio (caso E).

3.1.2 Caso B, settore filatura laniera

Il caso B ha invece riguardato l'ottimizzazione delle sequenze di assegnazione degli ordini produttivi in una filatura. L'elemento strategico della scelta, che rende molto problematica l'utilizzazione di tecniche standard di ricerca operativa, sta nell'assegnazione della produzione degli ordini ad una batteria di filatoi tenendo conto di un multiforme sistema di vincoli introdotti dalle esigenze dei clienti e dalla tecnologia, nonché da questioni logistiche interne all'azienda e da problemi di adattamento dei diversi impianti alle singole produzioni.

La gestione reale dell'azienda è operata da un "dispositore", il quale procede secondo regole di conoscenza tacita, difficile da formalizzare e anche da replicare in modo sempre coerente.

La simulazione con un modello ad agenti ha consentito di disporre di una rappresentazione uno ad uno della realtà aziendale e di simulare le conseguenze dell'assegnazione di una sequenza storica di ordini in modo differenziato rispetto a quello effettuato nella realtà. Le differenze introdotte sono state: (i) assegnazione degli ordini in modo casuale; (ii) ottimizzazione dell'assegnazione degli ordini impiegando un algoritmo genetico per definire sequenze di lavorazione continuamente ridefinite per tener conto dei nuovi ordini, in presenza di uno spazio delle soluzioni straordinariamente complesso.

Dal punto di vista dei costi aziendali, tenendo conto anche dei ritardi di consegna ai clienti - aspetto molto rilevanti in una filiera che sta introducendo la consegna "in tempo" - la soluzione casuale è risultata moderatamente peggiorativa rispetto all'operato dell'esperto umano. La soluzione fondata sull'algoritmo genetico è invece risultata sensibilmente migliorativa, anche se ha introdotto potenziali problemi di adattamento continuo dei diversi impianti, con possibili resistenze psicologiche da parte degli addetti a modificare ripetutamente lo stato dei macchinari.

In questa realizzazione la conoscenza dell'esperto può quindi integrarsi vantaggiosamente con i risultati derivanti dall'algoritmo genetico.

3.1.3 Caso C, settore meccanico

Il caso C riguarda una azienda meccanica produttrice su larga scala di prodotti relativamente ripetitivi, ma di alta qualità e segmentati in una molteplicità di dimensioni e sottocaratteristiche.

Il problema principale si è rivelato essere quello dell'assegnazione delle priorità di lavorazione per le sequenze produttive relative ai prodotti e alle loro sottoparti, tenendo conto della disponibilità di macchinari multifunzionali, anche se vincolati da risorse comuni (ad esempio il personale che può operare ad una stazione di lavoro o ad un'altra, in modo alternativo).

In questo terzo caso si è introdotta per la prima volta l'applicazione piena del modello jVE (Paragrafo 4), che non era ancora definito nella sua completezza quando sono stati affrontati i casi A e B.

Utilizzare il modello jVE descritto nel Paragrafo 4 significa mettere a fuoco separatamente due distinti elementi di conoscenza sulla realtà aziendale: (i) le sequenze di passi necessarie per realizzare i prodotti; (ii) le caratteristiche di ogni unità, esterna o interna, in grado di compiere una o l'altra o più delle fasi di lavorazione descritte nelle sequenze sopradette.

Codificare in modo totalmente disgiunto questi due aspetti determina la non ovvia conseguenza che il modello jVE fa effettivamente "funzionare" la realtà azienda simulata, non descrivendo una simulazione di processo in cui gli eventi sono minutamente descritti a priori e se ne segue lo svolgersi nel tempo, bensì realizzando una specie di *videogame* in cui gli eventi si svolgono in una sequenza di interazioni non controllata a priori dal programmatore che ha progettato il modello. Tale sequenza risulta dall'intrecciarsi della complessità dovuta al simultaneo operare delle sequenze produttive descritte ("ricette", vedi sotto 4.1) e delle capacità produttive individuate.

Anche nel caso C esiste un problema di scelte affidate ad un esperto umano, in particolare nel momento in cui vengono avviate le diverse produzioni; nel corso della simulazione ora in sviluppo l'esperto sarà a volta a volta sostituito da regole, assegnazioni casuali, strumenti di ottimizzazione, ...

Oltre ai tre casi già descritti, sono inoltre in avvio due altri casi particolarmente interessanti.

3.1.4 Caso D, logistica di un sistema di manutenzione

Il caso D riguarda la logistica di un sistema di manutenzione operante sul territorio. Gli interventi riguardano diverse tipologie di azione e sono al momento svolti da squadre monofunzionali; la simulazione deve verificare quali siano le conseguenze di una attività multifunzionale delle squadre, quali siano le esigenze formative necessarie, quale sia la diversa organizzazione da definirsi nei momenti decisionali, quali siano i vantaggi o gli svantaggi conseguenti alla nuova organizzazione proposta.

Le sequenze produttive si identificano nelle azioni che devono essere svolte sul territorio e le unità produttive nelle squadre mono o multifunzionali di intervento, con la condivisione di professionalità rare per determinate operazioni.

E' infine determinante l'interazione tra il modello di simulazione e un sistema di informazione geografico (GIS), per posizionare in modo realistico gli accadimenti nello spazio fisico.

La gestione della conoscenza riguarda le metodologie attuali di intervento e l'individuazione delle scelte relative alle priorità di azione; inoltre, la valutazione degli effetti di tali scelte.

La simulazione rappresenta qui il passaggio chiaro per mostrare la fattibilità e la convenienza qualitativa e economica di una razionalizzazione del sistema degli interventi, operando con unità multifunzionali, cosa che non è ovvia né, soprattutto, condivisa a priori.

3.1.5 Caso E, gestione *web* di una catena produttiva nell'abbigliamento

Il caso E è invece riferito ad una azienda produttrice di abbigliamento, che ha scelto di operare interagendo con le diverse unità produttive quasi esclusivamente tramite il *web*, anche a motivo della localizzazione mondiale delle stesse. In questo caso esistono produzioni tradizionali, con ricette produttive e unità produttive come descritte nel caso C; ma esiste a monte un momento organizzativo di straordinaria importanza, che contiene sequenze produttive a sé e unità produttive (di servizio) attive in momenti temporalmente predeterminati rispetto a quelli della produzione tradizionale.

L'azienda è relativamente giovane e la gestione della conoscenza è ampiamente consapevole; c'è inoltre un interesse interno alla modellizzazione della realtà aziendale via simulazione, anche per formalizzare lo stato organizzativo e consentirne il raffronto con strutture più tradizionali.

La sfida è quella di adattare il processo jVE del Paragrafo 4 a casi molto differenziati quali sono quelli qui descritti. In quest'ultimo caso con l'obiettivo non agevole di consentire un paragone virtuale tra il funzionamento della azienda così come è organizzata via *web* e una analoga azienda di carattere più tradizionale; paragone necessario per le decisioni strategiche a venire e non condotto attraverso un processo di simulazione.

3.2 Generalizzazione della casistica

In tutti i casi sopradetti il problema della gestione della conoscenza e della relativa trasmissione e accessibilità della stessa è ben evidente. Per una trattazione formalizzata di questo tipo di problemi, si veda Garicano (2000). Per una trattazione della stessa tematica in un ambito cognitivo, si veda Pozzali e Viale (2002).

Infine una esperienza internazionale, che può servire da punto di riferimento per il lavoro di simulazione di contesti aziendali, è quella del progetto NIIP (*National Industrial Information Infrastructure Protocols*), <www.niip.org>, il cui scopo è "rendere le aziende industriali degli Stati Uniti globalmente più competitive ed efficienti tramite nuove forme di elaborazione collaborativa dei dati per consentire la formazione di Imprese Virtuali. Il progetto NIIP aiuterà i fornitori innovativi della base industriale americana a trarre vantaggio dai più recenti progressi nella tecnologia dell'informazione ad oggetti, nella definizione dei dati e delle reti di intercomunicazione, per costruire le Imprese Virtuali. Infine, il progetto NIIP consentirà agli individui, alle imprese, alle organizzazioni o a loro divisioni, di riunirsi in imprese virtuali al fine di realizzare prodotti, servizi o soluzioni senza essere vincolate dall'uso di dati diversi, di processi differenti, di ambienti di calcolo o di tecnologie dell'informazione differenziate".

Si tratta di un progetto che, al di là delle ambizioni da valutare sulla base delle realizzazioni concrete, sta definendo protocolli di interoperabilità aziendale di grande interesse.

3.3 *Dai casi al metodo: simulazione e gestione della conoscenza*

Come rappresentare sotto forma di modello la realtà aziendale in casi come quelli descritti o in altri, con differenze e sfumature diversissime? Le tradizionali forme dei modelli descrittivi realizzati in modo verbale o dei modelli matematico-statistici (flessibili, ma non computabili, i primi; computabili i secondi, ma spesso inadatti a cogliere le complesse articolazioni del fenomeno da riprodurre) devono cedere il passo ai modelli di simulazione, ovvero al “formalismo del terzo tipo” presentato nell’introduzione, con riferimento a Gilbert e Terna (2000).

La scelta costruttiva del modello di simulazione deve a sua volta essere ben specificata e fondarsi su una metodologia sufficientemente standard (*Swarm*, nel nostro caso, vedere Paragrafi 1 e 1.1) e su scelte di struttura (*ERA*, per noi; vedere Paragrafo 1.2) modulari ed esplicite. Infine, su una architettura del modello di simulazione (jVE in questo lavoro, vedere Paragrafi 4 sino a 4.4) tanto robusta da poter comprendere al suo interno casi molto diversi.

Una architettura che consenta di maneggiare adeguatamente, all’interno del modello, gli aspetti taciti e espressi della conoscenza aziendale, codificando le sequenze di azioni e le scelte individuate nei comportamenti. Nella metodologia proposta ciò avviene: dal lato informatico, con l’incontro tra le sequenze produttive (le “ricette”, vedere Paragrafo 4.1) e la descrizione molto dettagliata della struttura fisica o immateriale (ad esempio in una azienda di servizi) che consente la produzione; dal punto di vista dei comportamenti, tramite una interazione ripetuta tra analisi del problema, stesura del modello e interazione, a modello funzionante, con i personaggi chiave che popolano il sistema produttivo simulato.

L’obiettivo è disporre di un sistema computabile su cui provare i cambiamenti e le scelte necessarie per migliorare o semplicemente far funzionare processi complessi, come se si lavorasse sulla realtà, tenendo conto in primo luogo della conoscenza effettivamente disponibile nel contesto studiato.

4. JVE - UNA METODOLOGIA DI SIMULAZIONE AD AGENTIPER L’AZIENDA²

Con il modello introdotto qui di seguito si intende “far funzionare” l’azienda simulata, non rappresentarla in modo animato sulla base di sequenze predeterminate di eventi; nel nostro modello gli eventi accadono in modo indipendente, generando interazione anche imprevedibili tra atti produttivi e unità produttive, proprie della complessità.

Certo su un ideale asse³ che vada dall’astrazione dei cosiddetti vetri di *spin* come strumento per studiare la complessità allo sviluppo di un videogioco sofisticato con accadimenti non definiti a priori, soprattutto nelle

² Sono debitore dell’idea iniziale per la realizzazione di questa metodologia ad un imprenditore torinese, il dr. Andrea Giacardi, cui va il mio ringraziamento.

³ Sono debitore della contrapposizione “vetri di *spin* / videogiochi” allo studioso dell’intelligenza artificiale e delle relative applicazioni al mondo aziendale, Roberto Serra.

loro reciproche relazioni, ci collochiamo in prossimità della seconda prospettiva.

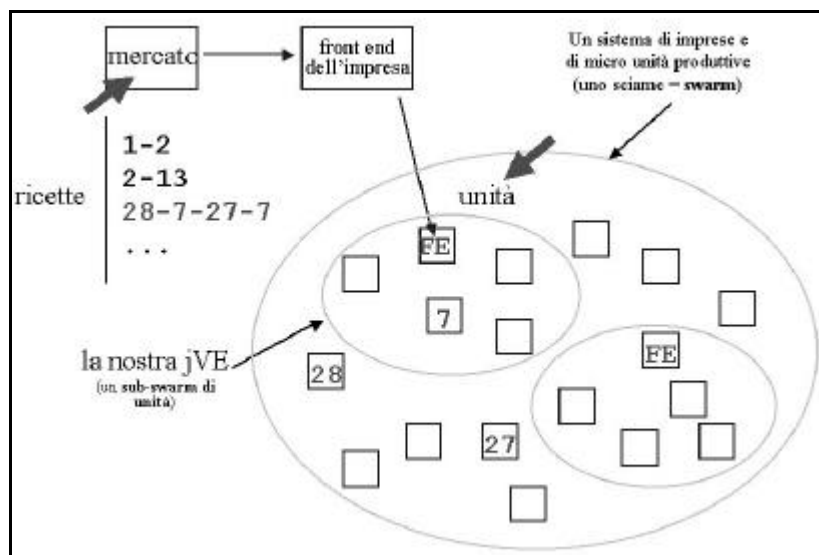


Fig. 3 – jVE: ricette dei prodotti e unità produttive

4.1 La struttura jVE

Da *Swarm* si passa quindi a jVE (Java Virtual Enterprise)⁴. Una descrizione del modello jVE non può che iniziare dal formalismo con cui sono indicati i prodotti.

Si tratta di una semplicissima rappresentazione tramite sequenze di codici numerici (le sequenze complete formano le “ricette”), ciascuno dei quali rappresenta un passo nella realizzazione di un bene, merce o servizio che sia. Nella Figura 3 sono proposti alcuni esempi, da semplici (1-2; 2-13) a meno semplici 28-7-27-7. Sempre nella Figura 3 sono rappresentate alcune unità produttive capaci di svolgere ciascuna un passo (o passi diversi, se l’unità è multifunzionale) della produzione all’interno delle varie ricette, oppure di operare come interlocutori del mercato, i *front end*. Le singole unità produttive possono essere autonomi micro-mechanismi operanti nel sistema economico, oppure unità integrate all’interno di imprese.

4.2 La sequenza di funzionamento di jVE

Nella Figura 4 si indica come i passi della produzione, muovendo dall’evento *a*, che rappresenta un ordine, ad esempio coincidente con la sequenza indicata in 28-7-27-7, siano via via gestiti da unità produttive diverse.

⁴ Il codice informatico del modello jVE, denominato *jveframe-x.tar.gz*, in cui *x* identifica la versione, può essere scaricato da <web.econ.unito.it/terna/jve>.

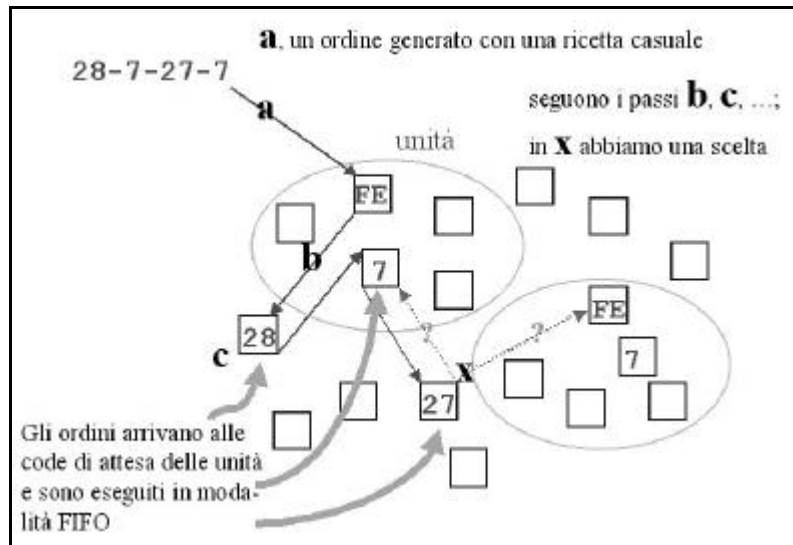


Fig. 4 – jVE: sequenza di funzionamento

L'informazione relativa ai passi da compiere è gestita in modo totalmente decentrato, considerando ogni ordine come un modulo che riporta la storia dei passi già espletati e la sequenza di tutti gli accadimenti anche futuri necessari per la produzione.

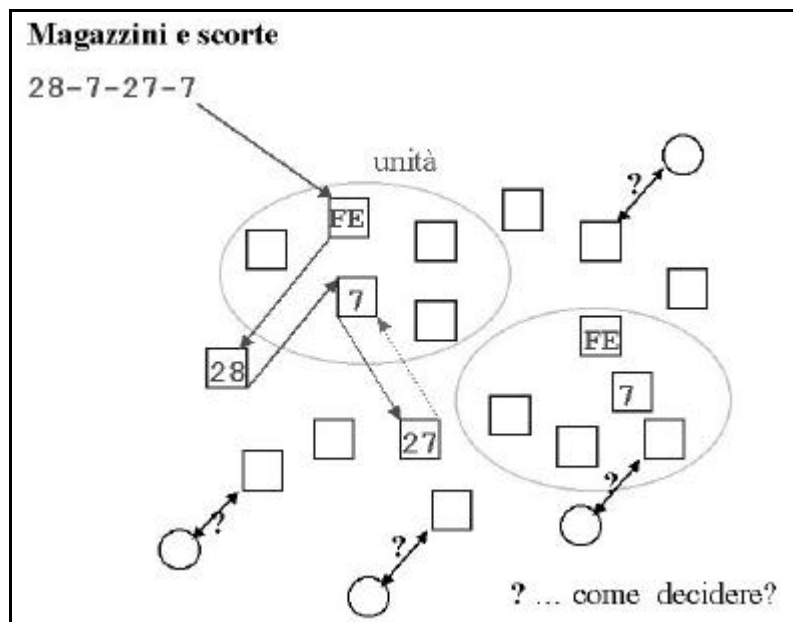


Fig. 5 – jVE: scorte e magazzini

Ogni unità produttiva, quando prende in carico un ordine, lo accoda nella propria lista di ordini da eseguire secondo la modalità cosiddetta FIFO

(*First In First Out*); dopo l'esecuzione richiede all'ordine (coincidente con il modulo che contiene la ricetta produttiva), l'informazione necessaria per trasmetterlo ad una successiva unità.

4.3 Magazzini e scorte; gestione dell'informazione

Esattamente come nella realtà aziendale, rileviamo problemi di assegnazione quando più unità produttive possono svolgere la stessa lavorazione, all'interno o all'esterno della realtà dell'impresa che stiamo simulando.

Osservando la Figura 5 si rileva che anche altre decisioni devono essere assunte all'interno del modello, come riproduzione di ciò che accade nella realtà: nel nostro caso, se produrre o no scorte da detenere nei magazzini.

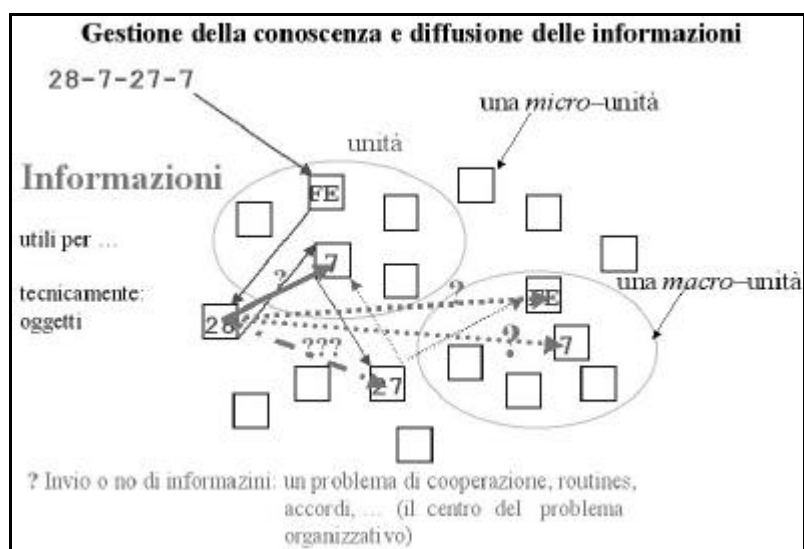


Fig. 6 – jVE: circolazione delle informazioni

Inoltre, nella Figura 6 emerge il problema della circolazione delle informazioni che ogni unità produttiva dovrebbe trasmettere alle altre unità con cui è logicamente connessa, per facilitarne le decisioni, ad esempio proprio nel campo della formazione delle scorte o nella gestione delle sequenze di attività, oppure ancora nella scelta della destinazione delle risorse scarse condivise tra più unità produttive.

Quest'ultimo è un aspetto molto interessante da simulare, in quanto è molto probabile che, in presenza di risorse condivise e di vincoli di impiego tra più unità produttive, emergano interazioni di rilevante complessità.

Infine, la circolazione o no delle informazioni (problema molto realistico nella realtà aziendale, che interseca anche aspetti di gestione della conoscenza) determina ovviamente il maggior o il minor successo dei risultati economici simulati dal funzionamento del modello, in diretta corrispondenza con la realtà riprodotta.

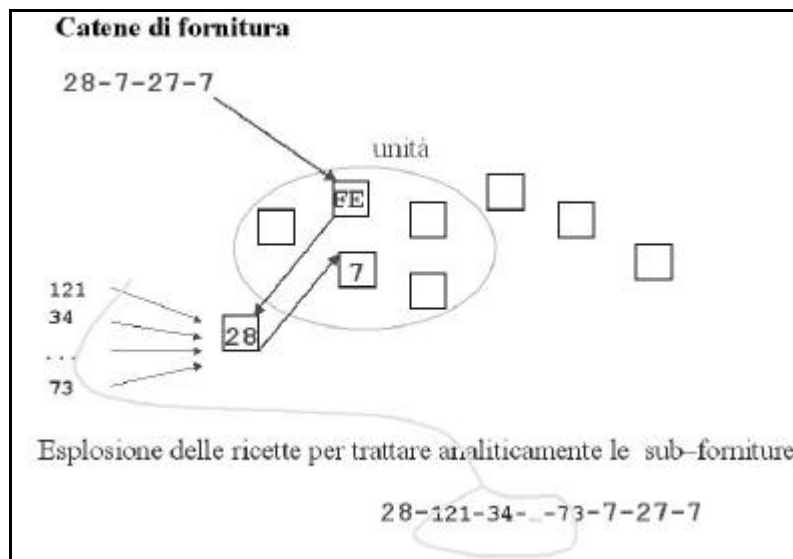


Fig. 7 – jVE: la catena di fornitura e i suoi segmenti

4.4 Contabilità; sub-forniture e catene produttive

Si tenga anche conto del fatto che il modello aggiorna continuamente la contabilità dei processi che sta simulando. Ciò sia secondo una prospettiva tradizionale di contabilità per unità produttive, con costi fissi e costi variabili, sia secondo la prospettiva analitica della contabilità secondo i prodotti, con risultati che ovviamente non sempre coincidono, non fosse che per l'attribuzione dei costi fissi.

Un'ultima considerazione in merito alla descrizione del funzionamento del modello jVE: la Figura 7 introduce la presenza di sub-processi, cioè di vere e proprie forniture che devono intervenire all'interno della catena di produzione, con problemi di sincronizzazione dell'inizio delle diverse produzioni e delle relative componenti, sia che si tratti di gestione di reti all'interno di una stessa azienda, sia di un sistema integrato di aziende.

Il formalismo delle "ricette" produttive consente di esplodere a piacere, in modo ricorsivo, questo tipo di informazione più dettagliata, sino al livello di analisi desiderato.

5. LA FINALIZZAZIONE TEORICA

Secondo Kirzner (1997, p.81):

Quando la scuola Austriaca sostiene che il processo di scoperta imprenditoriale si fonda su una tendenza verso l'equilibrio, l'espressione apparentemente vaga "tendenza verso" è usata deliberatamente, avvertitamente e in modo appropriato.

Tale tendenza esiste in ogni momento, nel senso che precedenti errori imprenditoriali generano opportunità di profitto che costituiscono incentivi per decisioni imprenditoriali correttive da assumere. Questi incentivi rappresentano ricompense per coloro che meglio sanno anticipare i

cambiamenti nelle condizioni dell'offerta e della domanda, cambiamenti che possono manifestarsi in modo anche sconcertante.

Dalla individuazione del processo di scoperta imprenditoriale non facciamo discendere l'esistenza di un processo continuo e a prova d'errore verso l'equilibrio, bensì la convinzione che esistano forze economiche che continuamente favoriscono tale tendenza all'equilibrio.

Di conseguenza, è del massimo interesse costruire modelli di simulazione che siano fondati su una formulazione astratta e generale di processo di produzione, ma che incorporino anche una realistica visione della realtà, come quella prima prospettata nello sviluppo dei casi aziendali, proprio per simulare processi continui di adattamento e innovazione "a prova ed errore".

In questo seguiamo direttamente il percorso tracciato dalla Scuola Austriaca, cui va il grande merito (Colombatto, 2001) di avere abbandonato la concezione perfettista e meccanicistica dell'economia neoclassica; nonché di avere proposto una visione dell'economia come l'analisi delle condizioni che consentono alla produzione e al benessere degli individui di crescere. Ciò proprio ponendo l'azione degli individui-imprenditori, con la capacità di trarre vantaggio dagli errori precedenti per sé e per gli altri, al centro della spiegazione dei fenomeni economici.

5.1 *Sperimentazione in contesti plausibili*

Mostrare, via simulazione, la plausibilità del funzionamento di un mercato hayekiano privo di coordinamento centrale e sorretto dal comportamento indipendente degli agenti, è operazione relativamente semplice (Terna, 2000).

Un passo decisamente più complesso è quello che si intraprende a valle della costruzione del simulatore di impresa sin qui descritto, per simulare la formazione dell'impresa e soprattutto l'innovazione o l'invenzione imprenditoriale, in contesti plausibili.

Per questo è molto importante trarre da casi di impresa realistici una varietà di strutture riproducibili tramite la modellistica standard qui presentata, per arrivare alla interazione tra aziende e alla modificazione interna alle aziende, in termini di processi produttivi, prodotti, strutture e aggregazioni. Il tutto per valutare, grazie alla simulazione, quale delle strutture che si formano sia più robusta ai cambiamenti, più rapida nel cogliere le innovazioni, più adatta allo sviluppo.

CONCLUSIONI

I "lavori in corso" sono quindi nella direzione di una unificazione (Kilpatrick, 2001) tra il paradigma della complessità e la visione hayekiana dell'ordine spontaneo e della filosofia del *laissez-faire* economico, con al centro un sistema di scelte micro, in continuo rinnovamento, da cui far derivare le conseguenze macro: un bell'obiettivo per chi costruisce modelli fondati sulla simulazione ad agenti nelle scienze sociali.

BIBLIOGRAFIA

- COLOMBATTO E. (2001), Dall'impresa dei neo-classici all'impresa di Kirzner. *Economia Politica*, n. 2, pp.157-179.
- GARICANO L. (2000), Hierarchies and the Organization of Knowledge in Production. *Journal of Political Economy*, vol. 108, n. 5, pp. 874-904.
- GILBERT N., TERNA P. (2000), How to build and use agent-based models in social science. *Mind & Society*, n. 1, pp.57-72.
- KILPATRICK H.E. jr. (2001), Complexity, Spontaneous Order and Friedrich Hayek: Are Spontaneous Order and Complexity Essentially the Same Thing? *Complexity*, vol. 6, n. 4, pp. 16-20.
- KIRZNER I. (1997), Entrepreneurial discovery and the competitive market process: an Austrian approach. *Journal of Economic Literature*, vol.XXXV, n.1, pp. 60-85.
- LAVOIE D., BAETJER H., TULLOH, W. (1990), High-Tech Hayekians: Some Possible Research Topics in the Economics of Computation. *Market Process*, vol.8, Spring, pp.120-147
- LIN F., STRADER T.J., SHAW M.J. (2000), Using Swarm for Simulating the Order Fulfillment Process in Divergent Assembly Supply Chains, in F. Luna and B. Stefansson (eds.), *Economic Simulations in Swarm: Agent-Based Modelling and Object Oriented Programming*. Dordrecht and London, Kluwer Academic, pp. 225-250.
- PARISI D. (2001), *Simulazioni - La realtà rifatta nel computer*. Mulino, Bologna.
- POZZALI A., VIALE R. (2002), Cognizione e Conoscenza Tacita nei Processi Innovativi. *Sistemi intelligenti*, n. 1, pp. ??.??.
- PRYOR F. L. (2000), Looking Backwards: Complexity Theory in 2028, in D. Colander (ed.) *The complexity vision and the teaching of economics*. Cheltenham, Edward Elgar, pp. 63-69.
- SCHLUETER-LANGDON C, BRUHN P., SHAW M.J. (2000), Online Supply Chain Modeling and Simulation, in F. Luna and B. Stefansson (eds.), *Economic Simulations in Swarm: Agent-Based Modelling and Object Oriented Programming*. Dordrecht and London, Kluwer Academic, pp. 251-272.
- TERNA P. (2000), Economic Experiments with Swarm: a Neural Network Approach to the Self-Development of Consistency in Agents' Behavior, in F. Luna and B. Stefansson (eds.), *Economic Simulations in Swarm: Agent-Based Modelling and Object Oriented Programming*. Dordrecht and London, Kluwer Academic, pp.73-104.
- TESFATSION L. (2001), Agent-Based Computational Economics: Growing Economies from the Bottom Up. *ISU Economics Working Paper*, n. 1, <<http://www.econ.iastate.edu/tesfatsi/acealife.pdf>>.