

UNIVERSITÀ DEGLI STUDI DI TORINO

FACOLTÀ DI ECONOMIA
CORSO DI LAUREA IN ECONOMIA E COMMERCIO

TESI DI LAUREA:

**Strumenti per la didattica: la simulazione ad
agenti in JavaSwarm e software web-based.**

Relatore:
Prof. Pietro Terna

Correlatore:
Prof. Sergio Margarita

Candidato:
Saverio Iacomussi

ANNO ACCADEMICO 2001-2002

Torino, novembre 2002

Ringrazio i miei genitori Dante e Maria Augusta, il loro lavoro ed il loro continuo esempio hanno reso possibile raggiungere questo importante traguardo.

Desidero esprimere la mia riconoscenza a Mariella per la sua costante presenza.

Sono inoltre grato ai miei fratelli, Paola e Alberto per la loro pazienza.

In fine un grazie a Marco e Cinzia per i preziosi consigli.

Saverio Iacomussi

Indice

Introduzione	1
1 L'insegnamento dell'economia: problemi e prospettive	5
1.1 La riforma degli ordinamenti universitari	5
1.2 Il modello organizzativo 3+2+3	6
1.3 Le tipologie di attività formativa	7
1.4 Effetti specifici della riforma sull'insegnamento universitario nei corsi di primo livello	8
1.5 L'insegnamento dell'economia nel nuovo modello didattico	10
2 L'Unione Europea e i nuovi modelli di didattica	11
2.1 Lo spirito europeo	11
2.2 Piano di azione eLearning	15
2.2.1 Uno spazio europeo di ricerca sui nuovi ambienti di apprendimento	17
2.3 Strumenti per la realizzazione del piano di azione eLearning	19
2.3.1 Socrates	19
2.3.2 Leonardo	25
2.4 Fondi strutturali	27
2.4.1 FESR- Fondo Europeo di Sviluppo Regionale	27
2.5 Situazione nazionale	29
3 La teoria del consumatore	30
3.1 Introduzione	30
3.2 Il vincolo di bilancio	31
3.3 Variazioni della retta di bilancio	33
3.4 Preferenze	35
3.5 Assunzioni	36
3.6 Le curve di indifferenza	37
3.7 Preferenze well-Behaved	42
3.8 Saggio marginale di sostituzione MRS	44
3.9 La scelta del paniere ottimo	45
3.10 L'utilità	46
3.11 Utilità cardinale e ordinale	47
3.11.1 L'utilità ordinale	48
3.12 Utilità marginale	47
3.13 Scelta ottima	53
3.14 La funzione di utilità di Cobb-Douglas	56

3.15	Critiche al modello del consumatore	59
4	L'economia e la complessità	65
4.1	La complessità	65
4.2	Lo scenario di Santa Fe e le teorie economiche	70
4.3	Esempi di sistemi complessi	72
4.4	La sfida della complessità	74
5	La simulazione e l'apprendimento dell'economia	79
5.1	Modelli didattici per lo studio dell'economia	79
5.2	Il progetto Swarm	82
5.3	Dal progetto Swarm all'applicazione	83
5.4	Lo schema ERA	85
5.5	Java	87
5.6	Le librerie in Swarm	91
6	Didattica e simulazione: il modello Consumer	94
6.1	Premesse sul software	94
6.1.1	Free software e Open Source	94
6.2	Consumer	95
6.2.1	Avvio del programma	96
6.2.2	I menu di scelta	96
6.2.3	Avvio della simulazione	101
6.2.4	L'ambiente	104
6.2.5	Le imprese	105
6.2.6	I consumatori	106
6.3	Applicazioni	109
6.3.1	Nessuno acquista	109
6.3.2	Tutti acquistano	110
6.3.3	Prezzi oscillanti	111
6.4	Numerosità	112
6.5	Conclusioni	114
6.6	Evoluzioni del modello	114
7	Formazione a distanza	115
7.1	Cenni storici	115
7.2	Prospettive future della didattica a distanza	118
7.3	L'aula virtuale	120
7.4	Punti critici ed accorgimenti	123
7.5	Il ruolo dell'human factor	125
8	Esempi di software web-based	128
8.1	Introduzione	128
8.2	Future Learning Environment 3	129

8.3	FLE3 Tool	133
8.4	Real Presenter basic	136
8.4.1	Modifiche alle presentazioni	139
8.5	Microsoft Producer	140
9	Conclusioni	142
	Riferimenti bibliografici	145
	Appendici:	
I	Il codice Java - Swarm del modello Consumer	151
II	L'output del modello Consumer	191

Introduzione

L'apprendimento è un fenomeno di massa, che investe l'intera società, e che nel corso della storia ha attraversato un cammino tutt'altro che lineare. Ogni volta che una nuova tecnologia è penetrata in profondità nei costumi e nei modi di vivere degli uomini, si è prodotto uno scarto, una trasformazione che ha modificato e modellato il modo di essere e di apprendere di ognuno. Uno di essi è, ad esempio, il passaggio dalla cultura orale a quella scritta.

L'estendersi della comunicazione globale porta oggi alla vigilia di una nuova trasformazione radicale. Al libro si affianca il computer. Nelle teorie economiche classiche vi è un marcato distacco dalla realtà, ciò poiché si tenta con il rigore della matematica e della statistica di spiegare fenomeni sociali e comportamentali. Negli ultimi anni per sopperire a queste mancanze molti ricercatori hanno accostato al modello matematico-statistico, quello sociologico-cognitivo con crescenti riconoscimenti a livello internazionale, tra cui Daniel Kahneman, Nobel per l'economia nell'anno 2002, premiato:

“[...] per aver integrato argomenti della ricerca psicologica con le scienze dell'economia, con particolare riguardo ai processi decisionali e di giudizio nelle incertezze.”

All'approccio sociologico-cognitivo si è affiancato l'ausilio della

simulazione, con altrettanto importanti riconoscimenti dalla comunità scientifica, ultimi in ordine di tempo a Vernon L. Smith vincitore a pari merito con Kahneman del premio Nobel per l'economia:

“[...] per aver individuato delle prove di laboratorio come strumento per le analisi economiche empiriche, specialmente nello studio del funzionamento dei mercati alternativi”.

Smith fin dal 1970 utilizza simulazioni economiche per sperimentare quanto si realizza nei mercati.

Nel processo di apprendimento è opportuno approfondire sia la componente teorica che quella pratica. Nella pratica si verifica e si applica quanto in precedenza appreso dalla teoria. La simulazione contribuisce alla semplificazione e all'avvicinamento delle due componenti, permettendo di colmare la distanza che intercorre tra:

- pratica economica,
- mondo reale,
- teoria economica.

L'economia così impostata diventa materia non solo teorica ma anche tecnica. Tale visione è indispensabile per la formazione universitaria che il legislatore ha disciplinato nella riforma dell'ordinamento universitario.

Il presente lavoro nasce dalla riflessione sulle potenzialità che la tecnologia offre al mondo della scuola e all'università. In particolare si vuole dimostrare come attraverso la simulazione lo studente

può ripercorrere e concretizzare ciò che ha appreso nella teoria.

Il capitolo primo analizza il “Regolamento recante norme concernenti l’autonomia didattica degli atenei” (in breve, RAU), pubblicato sulla Gazzetta Ufficiale del 4 gennaio 2000, evidenziando le caratteristiche innovative della riforma degli ordinamenti didattici universitari.

Il capitolo secondo esamina i progetti dell’Unione Europea, che incentivano la nascita di un nuovo modello di insegnamento. Inoltre ripercorre gli obiettivi fissati dal Consiglio Europeo di Lisbona del marzo 2000.

Nel capitolo terzo si verifica la teoria del consumatore l’obiettivo è quello di semplificare le numerose variabili che interagiscono nella fase in cui il consumatore decide di spendere il proprio reddito per acquistare uno o più beni. Nei corsi universitari di istituzioni di economia e di economia politica, solitamente, questo è il primo modello economico introdotto agli studenti.

Il capitolo quarto introduce il concetto di complessità e propone cambiamenti nella visione classica dell’economia. Si propone lo studio della complessità effettuato dal Santa Fe Institute (<http://www.santafe.edu>).

Il capitolo quinto esamina due modelli classici di apprendimento dell’economia: quello letterario-descrittivo, e quello matematico-statistico. E’ presenta un terzo modello: la simulazione, attraverso il progetto Swarm, supporto standardizzato per lo sviluppo di modelli di simulazione multi-agente.

Il capitolo sesto illustra il modello di simulazione del consumatore “Consumer”, scritto in linguaggio di programmazione Java, sup-

portato dalle librerie di Swarm; il modello costruito può essere utilizzato per la didattica perché concretizza la teoria del consumatore.

Il settimo capitolo ripercorre l'evoluzione della formazione a distanza, dal sistema postale all'aula virtuale e quale ruolo debba essere assegnato alle nuove tecnologie nell'ambito dei processi didattici tradizionali e nei modelli di apprendimento di tipo innovativo.

Il capitolo ottavo presenta alcuni software web-based: Future Learning Environment 3 (<http://FLE3.org>), Software Open Source web-based per la crescita della conoscenza e per l'apprendimento collaborativo, sviluppato dall'UIAH Media Lab, in cooperazione con il Center for Research on Networked Learning and Knowledge Building, Department of Psychology, University of Helsinki.

Real Presenter basic programma gratuito prodotto dalla Real (<http://www.real.com>), e Microsoft Presenter (<http://www.microsoft.com>) che permettono di realizzare presentazioni commentate prendendo l'avvio da file generati dal programma Power Point.

Seguono le conclusioni, la bibliografia e l'appendice contenete i listati del programma di simulazione "Consumer" realizzato come supporto alla didattica.

Capitolo 1

L'insegnamento dell'economia: problemi e prospettive

1.1 La riforma degli ordinamenti universitari

Secondo quanto si deduce dal “Regolamento recante norme concernenti l'autonomia didattica degli atenei” (in breve, RAU), pubblicato sulla Gazzetta Ufficiale del 4 gennaio 2000, le caratteristiche più innovative della riforma degli ordinamenti didattici universitari sembrano essere le tre seguenti:

1. articolazione “verticale” del percorso formativo in corsi di studio disposti in serie su tre livelli principali (“laurea”, “laurea specialistica” e “dottorato di ricerca”), aventi durata standard di tre, due e tre anni rispettivamente (in breve, modello organizzativo 3+2+3);
2. articolazione “orizzontale” dei corsi di studio dei primi due livelli in classi di appartenenza, ciascuna delle quali raggruppa insegnamenti del medesimo livello, caratterizzati dagli stessi obiettivi formativi;

3. individuazione di sei tipologie di attività formative, per ciascuna delle quali è fissato il numero minimo di crediti indispensabile per conseguire il titolo di studio relativo a ogni determinata classe.

1.2 Il modello organizzativo 3+2+3

L'adozione del modello organizzativo 3+2+3, l'introduzione del sistema delle classi di corsi di studio, la generalizzazione del meccanismo dei crediti formativi costituiscono innovazioni rilevanti che hanno portato a cambiamenti radicali nel modo di concepire e strutturare la didattica universitaria.

In particolare, l'adozione di tale modello organizzativo introduce una pluralità di livelli e di corrispondenti titoli disposti in serie in un sistema universitario, quale quello italiano, finora essenzialmente incentrato sull'unico titolo tradizionale, cui sono state affiancate, nel corso degli ultimi anni, il dottorato di ricerca ed il diploma universitario.

Lo schema 3+2+3 rappresenta quindi una novità nella struttura del sistema universitario italiano, che potrà portare i docenti di tutti gli ambiti disciplinari a ripensare le modalità didattiche delle rispettive discipline, identificando con chiarezza i contenuti minimi (conoscitivi, metodologici, tecnico-analitici) corrispondenti ai tre diversi livelli previsti dal legislatore.

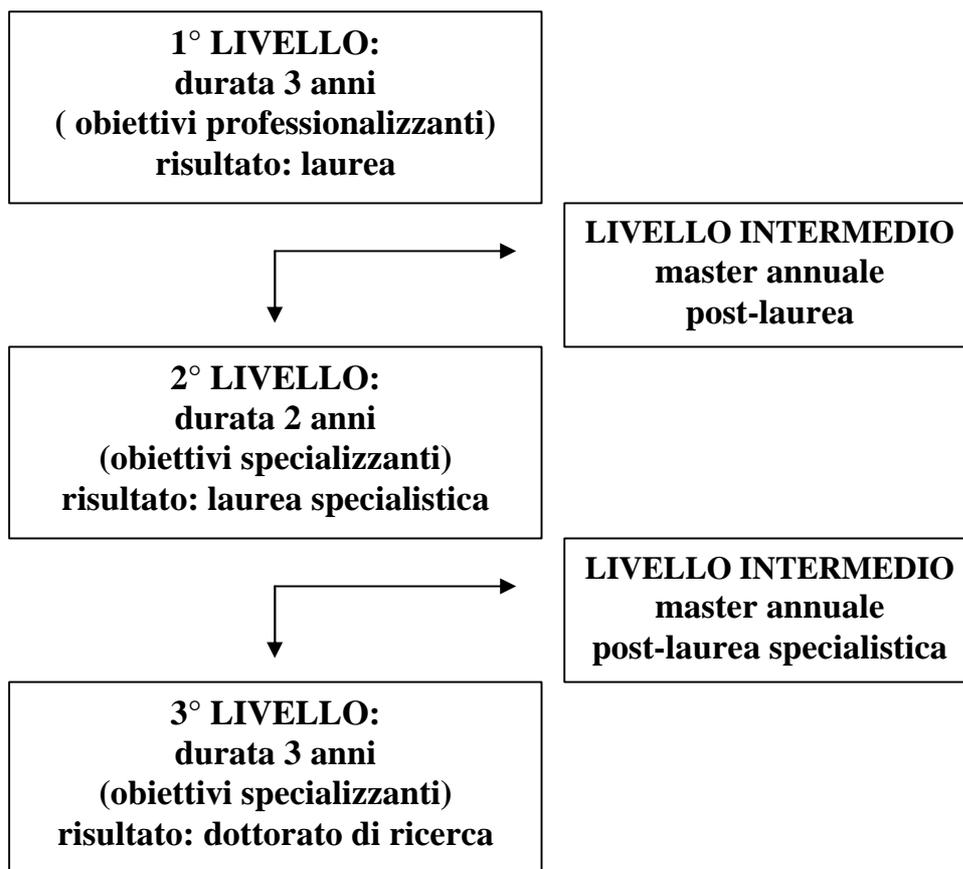


Figura 1. Modello organizzativo 3+2+3

1.3 Le tipologie di attività formative

Le sei tipologie di attività formative (a.f.) previste dall'Art. 10 del RAU sono le seguenti:

- 1) a.f. in uno o più ambiti disciplinari relativi alla formazione di base;
- 2) a.f. in uno o più ambiti disciplinari caratterizzanti;

- 3) a.f. in uno o più ambiti disciplinari affini o integrativi;
- 4) a.f. autonomamente scelte dallo studente;
- 5) a.f. relative alla preparazione della prova finale ecc.;
- 6) a.f. volte ad acquisire conoscenze linguistiche, abilità informatiche e telematiche, relazionali, anche mediante tirocini formativi ecc.

I primi tre tipi di attività formative riguardano le didattiche tradizionali, fondate sull'insegnamento diretto in aula (corsi, seminari, esercitazioni, laboratori, ecc.), mentre gli ultimi tre sono la vera novità.

1.4 Effetti specifici della riforma sull'insegnamento universitario nei corsi di primo livello

La riforma degli ordinamenti didattici esercita un'influenza rilevante sull'insegnamento universitario in tutti i livelli previsti dal legislatore (il RAU non prevede soltanto il primo, il secondo e il terzo livello, ma anche due ulteriori livelli intermedi "sui generis", rappresentati dai master annuali post-laurea e post-laurea specialistica).

Il problema più rilevante suscitato dalla riforma degli ordinamenti didattici, e più specificamente dall'adozione del modello seriale 3+2+3 (con le ulteriori complicazioni rappresentate dai master), ri-

sulta essere il seguente: come combinare in maniera armoniosa la formazione generale (di base, di tipo culturale, a orientamento teorico-metodologico), che deve pur sempre restare una caratteristica fondamentale dell'insegnamento universitario, con la formazione volta a obiettivi professionalizzanti, esplicitamente prevista dal legislatore come uno dei tratti distintivi della rinnovata didattica universitaria.

Il legislatore prevede da un lato che i corsi di primo livello siano da intendersi come corsi in se stessi completi e capaci di fornire una preparazione adeguata, anche ai fini professionali, agli studenti, presumibilmente la grande maggioranza degli immatricolati, che dovessero decidere di entrare nel mercato del lavoro dopo il conseguimento del titolo di primo livello. Dall'altro che alcuni studenti, presumibilmente una minoranza degli immatricolati, possano decidere di proseguire gli studi universitari dopo il conseguimento del titolo di primo livello, passando ai corsi di laurea specialistica. A questo proposito il RAU prescrive alle università che intendono attivare un corso di secondo livello la predisposizione di un curriculum formativo di primo livello che consenta di proseguire al livello successivo senza perdita di crediti.

In questo caso si pone lo specifico problema di come sia possibile disegnare, all'interno di una determinata classe o di una certa area disciplinare, percorsi formativi triennali nei quali possano convivere due esigenze diverse e contrapposte:

- da un lato l'esigenza posta dagli studenti orientati a entrare nel mercato del lavoro non appena completati gli studi di

primo livello, studenti che sono interessati a poter disporre di una formazione capace di garantire credibili sbocchi professionali;

- dall'altro l'esigenza posta dagli studenti orientati a proseguire al secondo livello, che sono interessati a poter disporre di una più approfondita formazione teorico-metodologica, che può consentire un passaggio non traumatico agli studi più avanzati di livello superiore.

1.5 L'insegnamento dell'economia nel nuovo modello didattico

Le materie economiche sono da più parti ritenute come teoriche, e poco utili nella formazione professionale dei laureati di primo livello, riservandole agli studenti che proseguono nel ciclo di studi.

Occorre dare un apporto formativo e tecnico alle materie economiche fondamentali, senza escludere l'importanza di un rigoroso impianto teorico. L'ausilio di nuove forme didattiche avanzate può apportare tale cambiamento.

Capitolo 2

L'Unione Europea e i nuovi modelli di didattica

2.1 Lo spirito europeo

Lo spirito di creare un nuovo modello di insegnamento trova grande stimolo nelle più recenti azioni della Commissione Europea; in occasione del Consiglio Europeo di Lisbona del marzo 2000, sono stati fissati ambiziosi obiettivi da raggiungere nell'Unione Europea entro i prossimi dieci anni.

L'istruzione è stata posta in cima alle priorità politiche, attraverso l'ampliamento dei sistemi di apprendimento e di formazione.

La disciplina basata sulle nuove tecnologie favorisce l'integrazione nell'istruzione, come fattore di cambiamento, e offre una buona occasione per migliorare la qualità, l'accessibilità, la varietà e l'efficacia dell'offerta di cultura.

Con la risoluzione del 13 luglio del 2001 il consiglio della comunità Europea ha ripreso lo spirito di Lisbona di fissare un'economia della conoscenza competitiva e dinamica con obiettivi specifici riguardo alle tecnologie dell'informazione e della comunicazione (di seguito denominate come TIC).

Tale risoluzione invita gli Stati membri dell'unione a:

1. Perseverare negli sforzi concernenti l'effettiva integrazione delle TIC nei sistemi di istruzione.
2. Sfruttare pienamente le potenzialità di internet, degli ambienti multimediali e di apprendimento virtuale per migliorare più rapide realizzazioni di educazione come principio educativo di base.
3. Promuovere le necessarie possibilità di apprendimento delle TIC nel contesto dei sistemi di istruzione e formazione accelerando le integrazioni delle TIC.
4. Perseverare negli sforzi concernenti la formazione iniziale e continua degli insegnanti e dei formatori in quanto all'utilizzo delle TIC.
5. Incoraggiare i responsabili degli istituti d'insegnamento e di formazione nonché coloro che decidono a livello locale, regionale e nazionale ed altri operatori interessati ad acquisire le necessarie competenze.
6. Accelerare l'acquisizione di attrezzature, di infrastrutture di qualità per l'istruzione e la formazione, tenendo conto dei progressi tecnici: hardware, software.

7. Stimolare lo sviluppo di materiale didattico di alta qualità per l'insegnamento e l'apprendimento per garantire qualità nell'offerta on-line.
8. Facilitare con la digitalizzazione e la standardizzazione documentale le risorse culturali pubbliche.
9. Promuovere nuove metodologie e software pedagogici innovativi.
10. Sfruttare il potenziale di comunicazione delle TIC favorendo un senso di unità all'interno della comunità Europa.
11. Sostenere e sviluppare luoghi di incontro virtuale.
12. Approfondire la ricerca nel contesto e-learning in particolare per stabilire come migliorare le prestazioni dell'apprendimento attraverso le nuove TIC, lo sviluppo pedagogico, le implicazioni dell'insegnamento e dell'apprendimento basati sulle TIC, stimolando la cooperazione internazionale al riguardo.
13. Promuovere integrazioni tra il settore pubblico e quello privato per contribuire allo sviluppo dell'e-learning.

In seguito la risoluzione del consiglio Europeo diverrà un vero piano di Azione per lo sviluppo della didattica di nuova generazione nell'Unione Europea.

La Commissione Europea ha negli ultimi anni lanciato alcune iniziative per incentivare lo sviluppo e l'uso di nuove forme didattiche, mettendo a disposizione di soggetti pubblici e privati ingenti somme con l'iniziativa "*e-Learning: Pensare l'istruzione di domani*" ed il corrispondente piano d'azione 2001-2004, la Commissione ha gettato le basi di un'azione concreta e duratura basata su una serie di misure specifiche, miranti a coordinare le diverse attività condotte in Europa nel settore dell'apprendimento elettronico, avvicinandoci all'economia della conoscenza ed alla visione impostata a Lisbona.

Dall'adozione dell'iniziativa e-learning nel maggio 2000 e del piano d'azione e-learning nel marzo 2001, l'interesse per l'apprendimento fondato sulle tecnologie ha continuato a progredire. Il suo utilizzo per migliorare la qualità e l'accesso all'istruzione ed alla formazione è generalmente percepito come uno degli elementi chiave.

Questa presa di posizione fa seguito alla relazione del Consiglio "istruzione" intitolata "Gli obiettivi concreti futuri dei sistemi d'istruzione e di formazione", in cui si ribadisce l'importanza del didattico a distanza. Questi sforzi stanno producendo risultati incoraggianti. Il *benchmarking* eEurope definito per la relazione elaborata in previsione del vertice di Barcellona dimostra che l'obiettivo iniziale eEurope (fornire a tutte le scuole un collegamento a Internet) è tutt'altro che raggiunto, e che occorre di conseguenza concentrarsi sul

miglioramento dei collegamenti e su una maggiore diffusione dell'utilizzo di Internet nell'istruzione.

A livello degli Stati membri, la maggior parte dei paesi dispone di un proprio piano d'azione mirante ad incoraggiare l'utilizzo delle nuove tecnologie nell'istruzione e nella formazione; spesso tali piani prevedono un sostegno diretto a progetti pilota locali in materia di apprendimento elettronico presso scuole ed istituti di insegnamento superiore. In occasione del vertice sull'apprendimento elettronico del 10 e 11 maggio 2001 è stata inoltre lanciata una collaborazione volta a favorire la nascita di forum di cooperazione tra autorità pubbliche e imprese del settore dell'istruzione elettronica e delle comunicazioni di massa.

Le università utilizzano l'apprendimento elettronico come una fonte di valore aggiunto per i propri studenti e per offrire una forma di insegnamento a distanza, virtuale e flessibile reso possibile dalla rete. L'e-learning si sta rivelando un'evoluzione generale piuttosto che una rivoluzione.

2.2 Piano d'azione eLearning

L'iniziativa "eLearning - pensare all'istruzione di domani" è stata adottata dalla Commissione europea il 24 maggio 2000. In seguito alle conclusioni del Consiglio europeo di Lisbona. Tale iniziativa ha presentato i principi, gli obiettivi e le linee d'azione di eLearning, definiti per l'utilizzo delle nuove tecnologie multimediali e di Internet con l'obiettivo di migliorare la qualità dell'apprendimen-

to agevolando l'accesso a risorse e servizi nonché gli scambi e la collaborazione a distanza.

L'iniziativa eLearning si inserisce nel contesto del *piano d'azione globale eEurope2*, che "mira a consentire all'Europa di sfruttare i propri punti di forza e di superare gli ostacoli che si frappongono a un aumento dell'integrazione e dell'impiego delle tecnologie digitali", nell'ambito della *relazione sugli obiettivi futuri e concreti dei sistemi d'istruzione*, e pone tra i propri obiettivi le tecnologie dell'informazione e della comunicazione. L'efficienza dei sistemi di istruzione si basa interamente sull'efficacia delle impostazioni didattiche. Per ottenere i risultati sperati, l'introduzione delle tecnologie dell'informazione e della comunicazione dovrà essere accompagnata da una profonda riorganizzazione delle strutture di apprendimento.

Il piano d'azione interessa il periodo 2001-2004. Esso mira a mobilitare i soggetti attivi nel campo dell'istruzione e della formazione nonché i protagonisti in ambito sociale, industriale ed economico per fare dell'apprendimento permanente il motore dello sviluppo della società.

Obbiettivi:

1. dotare tutte le scuole di un accesso a Internet e alle risorse multimediali entro la fine del 2001 (non realizzato) e tutte le aule scolastiche (Internet rapida) entro la fine del 2002;
2. collegare progressivamente le scuole alle reti di ricerca entro la fine del 2002;

3. raggiungere il rapporto di 5-15 allievi per computer multimediale nel 2004;
4. garantire la disponibilità di servizi di supporto e di risorse didattiche su Internet e predisporre piattaforme di apprendimento per via elettronica ad uso di insegnanti, studenti e genitori entro la fine del 2002;
5. sostenere l'evoluzione dei programmi scolastici per tener conto dei nuovi metodi di apprendimento e dell'uso delle tecnologie dell'informazione e della comunicazione entro la fine del 2002.

2.2.1 Uno spazio europeo per la ricerca sui nuovi ambienti di apprendimento

Questo intervento, primo del genere sviluppato dall'Unione Europea mira a potenziare la ricerca sia sul piano didattico, sia su quello socioeconomico, sia quello tecnologico nel campo dell'elearning e dell'utilizzo delle TIC nell'istruzione e nella formazione professionale.

Intende altresì sviluppare un "centro di eccellenza virtuale" che consenta, con l'ausilio delle strutture esistenti negli Stati membri, di valorizzare i risultati dei progetti in corso e di trarre profitto dalle conoscenze acquisite riguardo a nuovi modelli e ambienti di apprendimento. Tale laboratorio (una struttura informale sostenuta

dalla Commissione) fungerà da piattaforma europea di incontro e di scambio: sarà un ponte tra istruzione e ricerca.

Verranno esaminati in particolare i tre temi seguenti.

– **L'evoluzione dei sistemi.** Ricerca, sperimentazione e previsioni sui nuovi ambienti di apprendimento dal punto di vista sia didattico che tecnologico. Verrà dedicata particolare attenzione alle tecnologie emergenti (GRID, satellite, radio e televisione digitali ecc.) per elaborare applicazioni innovative per l'istruzione e la formazione. Sono fondamentali in questo contesto gli aspetti didattici, organizzativi (comunità, regioni e organizzazioni interessate all'apprendimento) e di gestione dei cambiamenti.

– **I modelli virtuali.** Il concetto di campus virtuale; le nuove prospettive per le università europee, la mobilità virtuale con accesso alle risorse didattiche senza vincoli di tempo o di spazio.

– Attenzione per le **differenze individuali** nell'apprendimento e l'istruzione per esigenze specifiche. Cercando di sfruttare il potenziale delle nuove tecnologie per rimediare a situazioni di handicap, di emarginazione, di difficoltà di accesso all'apprendimento o a disfunzioni dell'istruzione organizzata secondo i metodi tradizionali. Verrà dedicata un'attenzione particolare alle pari opportunità tra uomini e donne.

2.3 Strumenti per la realizzazione del piano d'azione eLearning

2.3.1 Socrates

Socrates è il programma europeo in materia di istruzione; vi partecipa una trentina di paesi europei. La durata dell'attuale fase, la seconda, va dal 1° gennaio 2000 al 31 dicembre 2006.

Il grande obiettivo di Socrates è quello di dar vita a un' Europa della conoscenza per affrontare meglio le grandi sfide del nuovo secolo: promuovere l'istruzione permanente, incoraggiare l'accesso di tutti all'istruzione e aiutare i cittadini ad acquisire qualifiche e competenze riconosciute. Più concretamente, Socrates mira a promuovere l'apprendimento delle lingue e a incoraggiare la mobilità e l'innovazione.

In tutti i settori dell'istruzione Socrates promuove la cooperazione europea. Quest'ultima assume varie forme: la mobilità (spostarsi in Europa), l'allestimento di progetti comuni, la costituzione di reti europee, e la realizzazione di studi e analisi comparative. In pratica, Socrates offre borse per studiare, insegnare, seguire un tirocinio o corsi di formazione all'estero; fornisce agli istituti d'insegnamento un sostegno per allestire progetti didattici e per lo scambio di esperienze; aiuta le associazioni e le Organizzazioni non governative ONG ad organizzare attività su temi riguardanti l'istruzione e così via. Sono sovvenzionate solo attività che presentano una dimensione europea fondata su una cooperazione di tipo transnazionale.

A prescindere dai destinatari e dal tipo di progetto, Socrates promuove il carattere multiculturale dell'Europa. Appoggia l'istruzione dei gruppi più sfavoriti; promuove le pari opportunità tra uomini e donne e combatte gli insuccessi scolastici. Socrates attribuisce notevole importanza alle nuove tecnologie dell'informazione e della comunicazione ed incoraggia l'apprendimento delle diverse lingue europee nonché le innovazioni nel campo dell'istruzione.

Socrates comprende otto azioni distinte:

1. *Comenius*: insegnamento scolastico;
2. *Erasmus*: insegnamento superiore;
3. *Grundtvig*: educazione degli adulti e altri percorsi educativi;
4. *Lingua*: apprendimento delle lingue europee;
5. *Minerva*: tecnologie dell'informazione e della comunicazione nel settore dell'istruzione;
6. *Osservazione e innovazione dei sistemi* e delle politiche in materia d'istruzione;
7. azioni congiunte con altri programmi europei;
8. misure di accompagnamento.

Nell'ambito di nuovi modelli di didattica, e diffusione della TIC tra questi otto sono obiettivi rilevanti:

1. *Comenius*

In linea generale Comenius punta a migliorare la qualità e a rafforzare la dimensione europea dell'istruzione scolastica, in particolare incoraggiando la collaborazione transnazionale tra scuole, contribuendo a un migliore sviluppo professionale del personale direttamente coinvolto nel settore dell'istruzione scolastica e promuovendo l'apprendimento delle lingue e la consapevolezza interculturale.

Ogni anno viene pubblicato un invito a presentare proposte, sia per le azioni centralizzate che per quelle decentrate. I progetti relativi alle azioni decentrate sono selezionati dalle agenzie nazionali. La Commissione (con l'aiuto di esperti del mondo dell'università), le agenzie nazionali e gli uffici di assistenza tecnica selezionano i progetti relativi alle azioni centralizzate.

Attività connesse a eLearning sono:

- *Alfabetizzazione digitale*

Quasi tutte le azioni di Comenius si avvalgono delle TIC a fini di comunicazione e di organizzazione interna dei progetti. La cooperazione transnazionale nell'ambito dei progetti offre ai partecipanti opportunità di migliorare le proprie capacità di utilizzare il computer (multimedia, videoconferenze ecc.)

- Formazione del personale docente scolastico

Una delle priorità consiste nel preparare "gli insegnanti di oggi e di domani ad utilizzare le TIC come strumento didattico, come mezzo per aprire la scuola al mondo esterno e per valutare le conseguenze delle nuove tecnologie sulla gestione dell'attività in classe".

- Reti di centri per l'acquisizione delle conoscenze

Comenius incoraggia la costituzione di reti di progetti su temi d'interesse comune, al fine di promuovere in Europa la collaborazione e l'innovazione nell'istruzione scolastica in vari settori tematici.

2. *Erasmus*

Erasmus persegue l'obiettivo di migliorare la qualità dell'istruzione e di promuovere la dimensione europea dell'insegnamento superiore accrescendo la mobilità tra paesi europei ed incoraggiando la collaborazione transnazionale tra università. Comprende tre sottoazioni collegate tra loro: "Cooperazione interuniversitaria europea", "Mobilità degli studenti e dei docenti universitari" e "Reti tematiche".

Attività connesse a eLearning sono nel riconoscimento della maggiore importanza delle TIC. I collegamenti con nuovi modelli di didattica sono numerosi, mantenendo l'aspetto transnazionale del programma.

- Mobilità virtuale

La mobilità virtuale può integrare la mobilità fisica per prepararla e per prolungarne i benefici attraverso un periodo di studio o di lezioni *online* presso l'università ospitante. Tutto ciò offre ai partecipanti provenienti da zone svantaggiate l'opportunità di un accesso più ampio alle informazioni riguardanti le università e le altre istituzioni partecipanti al programma.

Prima di affrontare le lezioni che si svolgono presso l'università ospitante ci si può avvalere di corsi di preparazione linguistica *online*.

- Le TIC nelle reti tematiche Erasmus

Le reti tematiche Erasmus fanno ampio uso delle TIC, sia per le attività di collegamento, sia per la messa a punto di prodotti e l'elaborazione di conclusioni. Alcune di esse valutano il ruolo delle TIC in ambiti disciplinari specifici o in ambiti interdisciplinari; altre esaminano concretamente l'impatto degli ambienti virtuali di apprendimento.

3. *Minerva*

Nell'ambito del programma Socrates, l'azione Minerva è quella che più di tutte promuove le TIC e nuove forme di didattica. Essa si prefigge di favorire la collaborazione a livello europeo in relazione all'utilizzo delle tecnologie dell'informazione e della comunicazione nel campo dell'istruzione e di promuovere l'insegnamento aperto e a distanza (ODL, Open and Distance Learning). Essa abbraccia tutti i

settori dell'istruzione, dall'educazione prescolastica a quella degli adulti. Minerva è stata avviata nel 2000 ma è già forte dei buoni risultati ottenuti dai 166 progetti finanziati dal 1995 al 1999 nel campo dell'ODL; tali risultati verranno valorizzati nell'ambito dell'iniziativa *eLearning*. La durata dell'azione Minerva è di 7 anni, dal 2000 al 2006.

Il finanziamento dei progetti viene effettuato tramite un invito annuale a presentare proposte nell'ambito delle disposizioni generali previste dal programma Socrates. Per l'invito a presentare proposte collegato alla selezione dei progetti nel 2001 occorreva ad esempio inviare una proposta preliminare entro il 1° novembre 2000 e presentare una proposta completa entro il 1° marzo 2001. I progetti sono stati avviati il 1° settembre 2001.

Il bilancio è di circa 10 milioni di euro l'anno ed è destinato a finanziamenti di medie dimensioni (dell'ordine di 100 000 euro l'anno in media per progetti della durata di due anni finanziati al 50%). Tutte le attività previste dall'azione Minerva concorrono a conseguire gli obiettivi dell'iniziativa *eLearning*:

- la previsione degli sviluppi futuri, l'analisi e la condivisione dei risultati sull'osservazione dell'uso e l'individuazione e la diffusione delle migliori prassi;
- gli aspetti organizzativi e socioeconomici della diffusione delle innovazioni collegate alle nuove tecnologie per l'apprendimento;

- la dimensione didattica dell'inserimento delle nuove tecnologie nella pratica dell'insegnamento e dell'apprendimento;
- l'istituzione di una rete che colleghi coloro che esercitano responsabilità in relazione alla formazione degli insegnanti oppure in campo dirigenziale o decisionale, per l'adozione e l'uso pertinente delle nuove tecnologie;
- il potenziamento e la diffusione delle competenze europee in questo campo e il confronto con altri modelli;
- lo sviluppo a livello europeo di portali Internet che rappresentino un punto d'incontro per chi appartiene al mondo dell'istruzione, nonché di servizi educativi multimediali di qualità.

2.3.2 Leonardo

Il programma Leonardo (2000-2006), sostiene e perfeziona gli interventi degli Stati membri, contribuisce all'attuazione di una politica comunitaria in tema di formazione professionale. Esso si prefigge di: migliorare le competenze delle persone, soprattutto dei giovani, che seguono una prima educazione alla professione, in particolare mediante la formazione alternata al lavoro e all'apprendistato; migliorare la qualità nonché le opportunità di accesso allo sviluppo professionale continuo e all'acquisizione permanente di nuove competenze; promuovere e rafforzare il contributo della forma-

zione professionale al processo d'innovazione allo scopo di migliorare la competitività e l'imprenditorialità.

Vi sono tre settori che possono ricevere finanziamenti: i progetti pilota, i progetti relativi ad azioni tematiche e le azioni congiunte.

- I progetti pilota esaminano l'impiego delle TIC in prodotti e attività di formazione professionale, sostenendo lo sviluppo di reti transnazionali per la formazione professionale aperta e a distanza tramite l'uso delle tecnologie dell'informazione (prodotti multimediali, siti web, trasmissione via rete ecc.); essi riguardano la concezione, il collaudo e la convalida di nuove impostazioni in tema di formazione professionale derivanti da nuove situazioni di lavoro (ad esempio il telelavoro).
- Progetti riguardanti azioni tematiche si occupano dell'innovazione resa possibile dall'impiego delle TIC nella metodologia di insegnamento e a fini organizzativi. Sono compresi progetti che elaborano nuovi metodi di certificazione e di convalida delle competenze acquisite nel corso dell'attività lavorativa e al di fuori dei sistemi di formazione convenzionali nonché progetti che sviluppano risorse e piani per la formazione di insegnanti e formatori.
- Viene contemplata la possibilità di svolgere azioni congiunte con programmi comunitari riguardanti lo stesso settore, in particolare con il programma Socrates.

2.4 Fondi Strutturali

2.4.1 FESR - Fondo europeo di sviluppo regionale

Il Fondo europeo di sviluppo regionale (FESR) è stato istituito nel 1975. La direzione generale della Politica regionale agisce da capofila per la gestione del FESR, che è inteso a ridurre le disparità regionali nell'Unione e a promuovere lo sviluppo e la riconversione delle regioni. Il FESR cofinanzia programmi pluriennali.

- Nelle regioni dell'obiettivo 1 (regioni in ritardo di sviluppo):
 - investimenti in infrastrutture, segnatamente nell'ambito delle reti transeuropee (trasporti, telecomunicazioni ed energia);
 - investimenti nei settori dell'istruzione e della salute;
 - tutela dell'ambiente.

- Nelle regioni dell'obiettivo 2 (regioni con difficoltà strutturali):
 - risanamento di siti industriali e di zone urbane degradate;
 - integrazione e rilancio delle zone rurali o dipendenti dalla pesca, con azioni di risanamento, ristrutturazione, costruzione, protezione dell'ambiente e investimenti in infrastrutture e attrezzature.

- In tutte le regioni degli obiettivi 1 e 2:

- investimenti diretti nella produzione per creare occupazione durevole;
 - aiuto alle PMI e allo sviluppo locale, con particolare attenzione per i patti territoriali per l'occupazione;
 - potenziamento delle capacità regionali di ricerca e di sviluppo.
- Nelle regioni comprendenti una zona dell'obiettivo 1 o dell'obiettivo 2:
 - sostegno all'emergere di strategie innovative a favore della competitività regionale (azioni innovative).
 - In tutto il territorio dell'Unione:
 - risanamento economico e sociale delle zone urbane in difficoltà mediante l'iniziativa Urban;
 - sostegno alla cooperazione transfrontaliera, transnazionale e interregionale mediante l'iniziativa Interreg III.

La gestione del FESR è decentrata ed è generalmente affidata agli enti regionali delle regioni ammissibili.

L'eLearning e, in maniera più generale, la società dell'informazione, sono tra le direttrici consigliate dalla Commissione per favorire lo sviluppo delle regioni.

Più in genere i fondi strutturali della comunità Europea visto le molte possibilità di utilizzo permettono numerosi sviluppi in capo al progetto eLearning ciò perché il legislatore Europeo ha scelto di lasciare ampio margine di scelta nella destinazione delle risorse.

Nei singoli contesti nazionali si utilizzano i fondi strutturali europei.

2.5 Situazione nazionale

A livello nazionale le più importanti iniziative vanno ricercate attraverso il ministero della pubblica istruzione, il quale integra e sviluppa in modo analitico le direttive nazionali e comunitarie in campo di didattica:

Il Programma Operativo Nazionale "La Scuola per lo Sviluppo" è un programma integrato - Utilizza due fondi strutturali: Fondo Sociale Europeo (FSE) e Fondo Europeo di Sviluppo Regionale (FESR) - con un piano finanziario complessivo pari a 472,557 milioni di EURO di fondi comunitari.

Il programma mira a costituire un forte strumento per sostenere lo sviluppo e l'innovazione del sistema scolastico in funzione di migliorare l'occupabilità dei giovani e degli adulti e promuovere lo sviluppo economico e sociale del mezzogiorno mediante l'ampliamento delle competenze delle sue risorse umane.

Le linee strategiche e di intervento propongono un forte raccordo fra politiche nazionali di sviluppo del sistema di istruzione e Strategie europee per la crescita delle risorse umane a livello comunitario.

Gli stimoli per nuove proposte in campo universitario a livello nazionale trovano spunto a partire dal Ministero dell'Istruzione dell'Università e della Ricerca il quale coordina le iniziative formative.

Capitolo 3

La teoria del consumatore

1.1 Introduzione

La teoria del consumatore si pone l'obiettivo di semplificare le molte variabili che interagiscono nella fase in cui il consumatore decide di spendere il suo reddito per l'acquisto di uno o più beni. Nella formulazione più classica, è priva di dinamicità intertemporale, la condizione iniziale è che tutti i consumatori scelgono delle combinazioni ottimali tra quelle che possono effettuare, a queste ne seguono numerose altre necessarie per modellizzare i comportamenti del consumatore nella cd. "fase della scelta". Nei corsi di istituzioni di economia, e di economia politica in genere questo è il primo esempio di modello economico introdotto agli studenti.

La base della teoria è che il vincolo di bilancio sia la combinazione delle possibili scelte che un consumatore può compiere tenuto presente il suo reddito e la sua dotazione di moneta.

La teoria cerca di spiegare il fenomeno, semplificando quanto accade nel mondo reale: si supponga che esista una certa dotazione di beni tra i quali il consumatore può scegliere, con i rispettivi prezzi, ed una corrispondente dotazione di reddito spendibile, si cercherà di definire come e quanto spenderà il consumatore del suo reddito.

Per semplificazione la teoria viene normalmente introdotta con la possibilità di scelta tra soli due beni.

3.2 Il vincolo di bilancio

Si definisca con M il reddito disponibile del consumatore, con X e Y due diversi beni, con P_x il prezzo del bene X e con P_y il prezzo del bene Y .

Con *insieme di bilancio* (o insieme ammissibile) si definisce l'insieme di tutti i panieri che il consumatore può acquistare, anche se non spende tutto il proprio reddito, ed è pari a:

$$\{(x,y) : P_x x + P_y y \leq M\}$$

Il *vincolo di bilancio* è invece l'insieme di tutti i panieri che il consumatore può acquistare se spende tutto il proprio reddito ed è pari a:

$$\{(x,y) : P_x x + P_y y = M\}$$

L'equazione del vincolo di bilancio è $M = P_x x + P_y y$ che in forma esplicita diverrà:

$$y = (M/P_y) - (P_x/P_y) x$$

Concettualmente il vincolo di bilancio esprime come il consumatore, con una certa disponibilità di denaro, andrà a spendere tutto il

suo reddito nell'acquisto di due beni. La somma di quanto spende in un primo bene e quanto spende in un secondo bene andrà ad esaurire completamente la sua disponibilità di moneta.

Dall'analisi dell'equazione in forma esplicita si ricava che il rapporto tra i prezzi dei due beni (P_x/P_y) dal punto di vista matematico rappresenta la pendenza della retta del vincolo di bilancio, dal punto di vista economico evidenzia il prezzo relativo di x in termini di y inteso a quanto il consumatore deve rinunciare del bene y per comprare un'unità aggiuntiva del bene x, tecnicamente lo si può esprimere come costo opportunità, espresso in termini di y, di una unità in più di x.

Nel vincolo di bilancio uniche grandezze a prendere parte al modello sono il reddito disponibile, ed il prezzo dei due beni, non vi è quindi nessuna nozione di localizzazione, tempo, quantità.

La retta rappresenta le possibili scelte che il consumatore con la sua dotazione di fattori è in grado di compiere.

L'inclinazione della retta è data quindi dal rapporto tra i prezzi dei due beni P_x/P_y

L'intersezione della retta del vincolo di bilancio con gli assi evidenzia quante unità di un determinato bene si andranno a comprare nel momento in cui si rinuncia all'acquisto del secondo bene.

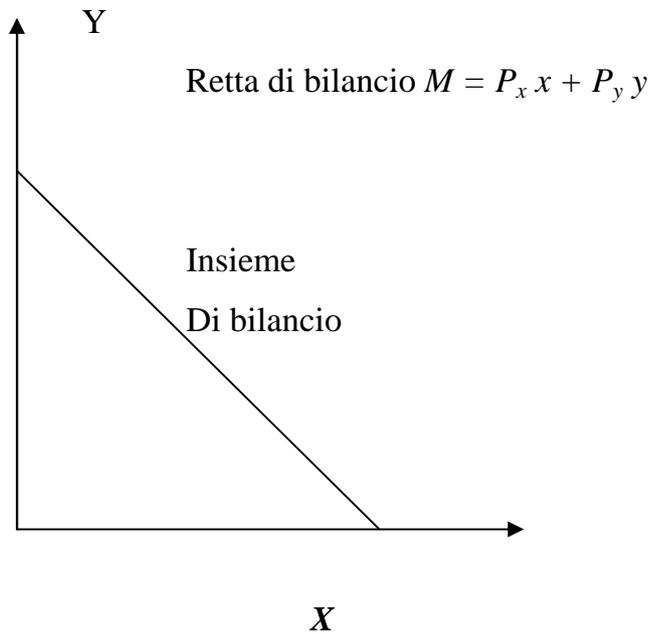


Figura 1. Il vincolo di bilancio

3.3 Variazioni della retta bilancio

Analizziamo, da prima, una variazione del reddito del consumatore, ciò comporterà traslazioni analoghe a quelle dei fasci propri.

I fasci di rette propri sono caratterizzati da traslazioni parallele delle curve, nei fasci impropri, invece, la traslazione è frutto della rotazione attorno ad un punto della curva.

Una variazione positiva, del reddito del consumatore, comporterà una traslazione parallela verso l'alto della retta, una diminuzione del reddito comporterà, invece, una traslazione verso il basso della retta del vincolo di bilancio.

Qualora si verifichi un aumento del reddito disponibile le quantità acquistabili di entrambi i beni saranno naturalmente più elevate, a fronte della maggiore disponibilità finanziaria.

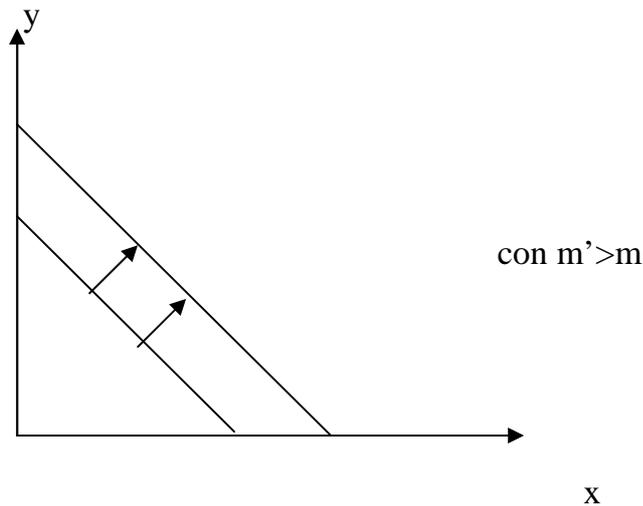


Figura 2. Variazioni del vincolo di Bilancio, aumento del reddito

Il medesimo effetto è riscontrabile qualora si verifichi una variazione del prezzo dei due beni, purché tale variazione risulti di eguale proporzione su entrambi i beni.

Variazioni di prezzo dei beni, tra loro proporzionali, e variazioni di reddito generano quindi una traslazione verso l'alto o verso il basso della retta del vincolo di bilancio.

Cambiamenti di prezzo di un singolo bene, od anche variazioni di prezzo di entrambi, ma non in misura proporzionale, comportano una rotazione della retta del vincolo di bilancio come rappresentato in figura 3.

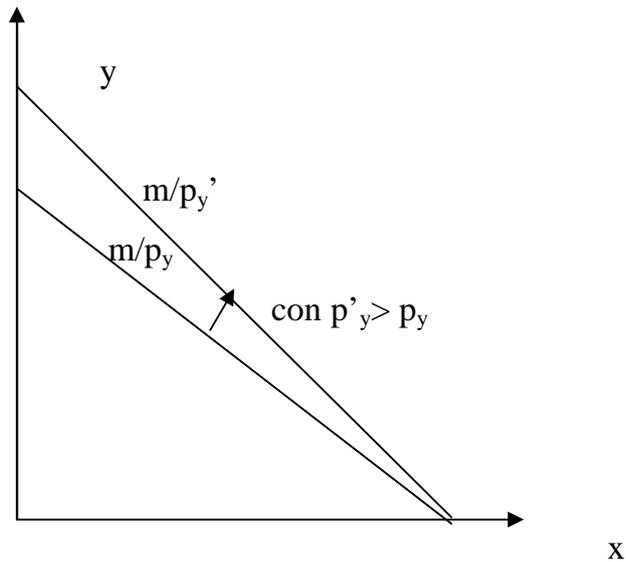


Figura 3. Variazioni del prezzo di un bene, aumento di p_y

3.4 Preferenze

Il vincolo di bilancio definisce gli aspetti oggettivi del modello: prezzi dei beni, reddito disponibile, in sostanza evidenzia il “poter acquistare, non il che cosa acquistare”.

Per introdurre la scelta del consumatore occorre introdurre i concetti di “preferenza del consumatore” nel significato di scelta preferita, e di “paniere di consumo” da intendersi come coppia di beni (x_i, y_i) scelti del consumatore; possiamo dire che il consumatore preferirà un certo paniere ad un altro se: $(x_1, y_1) \succ (x_2, y_2)$.

La preferenza espressa dal consumatore nella scelta del paniere è basata sul comportamento stesso del consumatore, il quale ordina, e quindi preferisce i diversi paniere acquistabili.

Se tra le possibili scelte due panieri risultano tra loro egualmente graditi al consumatore, cioè la scelta dell'uno o dell'altro è indifferente, avremo che $(x_1, y_1) \approx (x_2, y_2)$.

3.5 Assunzioni

Definiti i caratteri generali del modello occorre, per dare più coerenza, inserire una serie di assunzioni, dette assiomatiche, sulle relazioni di preferenza:

1. Il reddito del consumatore viene sempre speso interamente perché mai sazio.
2. Nei beni normali un aumento di consumo è sempre associato ad un aumento di utilitàⁱ. Tale assioma detto di *non sazietà* afferma che una quantità maggiore di un bene è sempre preferita ad una quantità minore.
3. *Completezza*, la quale permette il confronto tra due panieri tale per cui avremo:

$(x_1, y_1) \succ (x_2, y_2)$, paniere uno preferito al paniere due oppure,

$(x_1, y_1) \approx (x_2, y_2)$, panieri indifferenti oppure,

$(x_1, y_1) \prec (x_2, y_2)$, paniere uno meno preferito del paniere due;

questo principio garantisce un'ordinalità nelle scelte del consumatore, il quale definisce una scala di valori, in base alla quale decide i propri acquisti.

4. *Riflessività*, la quale assume che ogni paniere sia desiderabile almeno quanto se stesso, per cui avremo: $(x_1, y_1) \succ (x_1, y_1)$, a garanzia della razionalità delle scelte.
5. *Transitività*, la quale afferma che se $(x_1, y_1) \succ (x_2, y_2)$ e $(x_2, y_2) \succ (x_3, y_3)$ allora $(x_1, y_1) \succ (x_3, y_3)$. Se un primo paniere è preferito, o al più indifferente, ad un secondo e questo è preferito, o al più indifferente, ad un terzo allora il primo paniere sarà preferito, o al più indifferente, al terzo.

3.6 Le curve di indifferenza

Il concetto di preferenza viene graficamente rilevato dalle curve di indifferenza, le quali riportano nel piano cartesiano il concetto di scelta preferita dal consumatore.

La loro costruzione procede dall'individuazione di un primo paniere, e dall'insieme di panieri, che procurano una medesima soddisfazione al consumatore, definito dei "panieri indifferenti".

La curva passante per l'insieme così definito forma la "curva di indifferenza".

In base agli assiomi sopra elencati si evidenzia una proprietà delle curve di indifferenza: due curve di indifferenza non si possono mai intersecare.

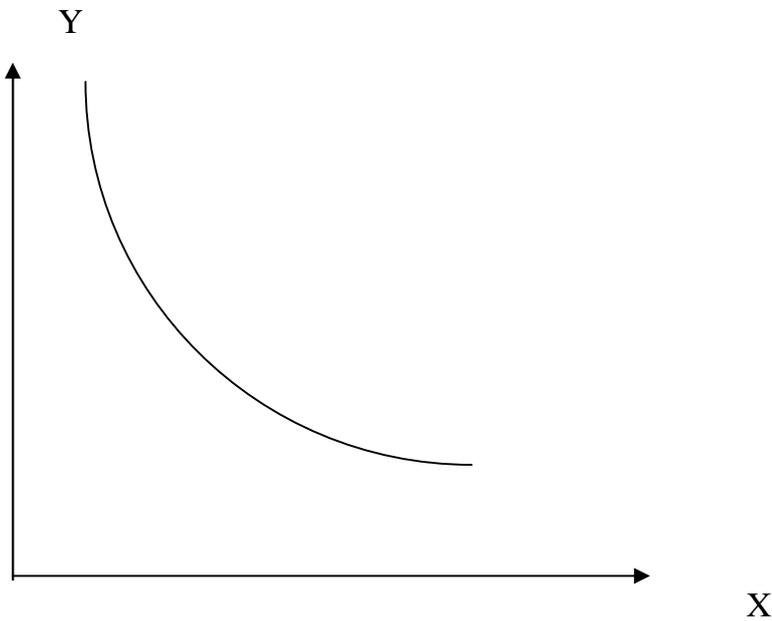


Figura 4. Preferenze del consumatori

Qualora si scelgano tre panieri di beni, X , Y , Z , tali che X giaccia soltanto su una curva di indifferenza, Y giaccia soltanto sull'altra, e Z sulla loro intersezione.

Per ipotesi le curve di indifferenza rappresentano livelli distinti di preferenza, cosicché uno dei panieri, ad esempio X , sarà strettamente preferito all'altro paniere Y . Sia noto che $X \approx Z$ e $Z \approx Y$, l'assioma di transitività implica che $X \approx Y$.

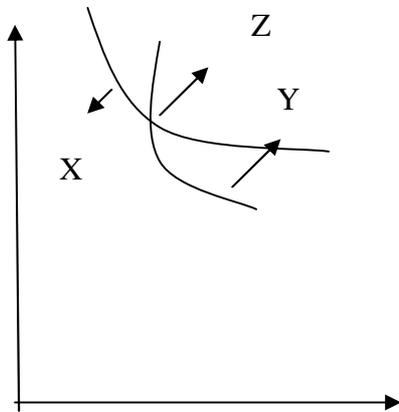


Figura 5. Intersezione di due curve di indifferenza

Questo contraddice l'ipotesi che $X \succ Y$, e di conseguenza curve di indifferenza a cui corrispondono diversi livelli di preferenza non possono intersecarsi.

Con *mappa di indifferenza* si va a definire l'insieme delle curve di indifferenza.

La teoria delle preferenze evidenzia differenti tipi di curve, associati a beni con caratteristiche diversi:

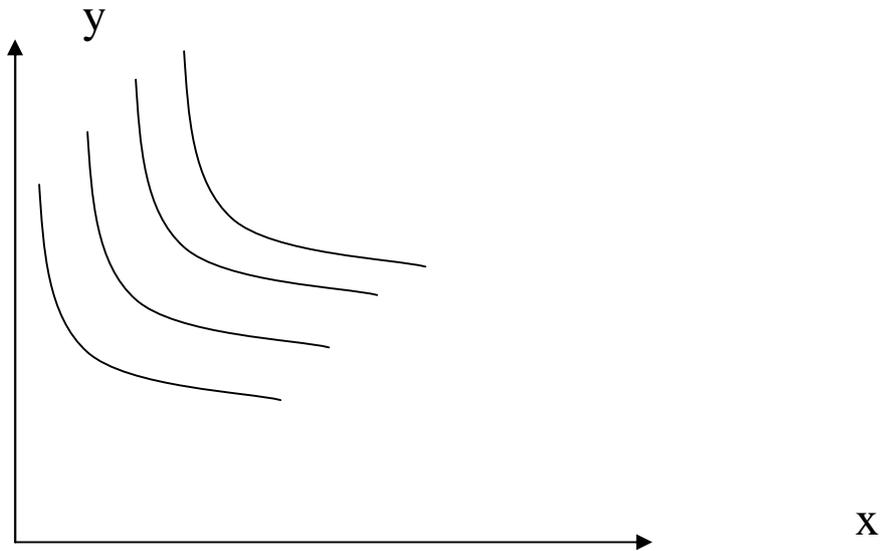


Figura 6. Mappa di indifferenza

- *Perfetti sostituti*, beni in cui il consumatore è disposto a sostituire un bene con l'altro ad un saggio di sostituzione costante, per il livello di soddisfazione non è rilevante la differenziazione tra i due beni ma l'ammontare totale delle quantità degli stessi. (fig. 7A)
- *Perfetti complementi*, sono quei beni che vengono sempre comprati congiuntamente, in proporzioni definite.(fig. 7B)
- *Mali*, sono invece quei beni che il consumatore non apprezza, e che quindi non vuole acquistare. (fig. 7C)
- *Beni di Giffen* quei beni per cui ad una riduzione del prezzo segue una riduzione dei consumi.

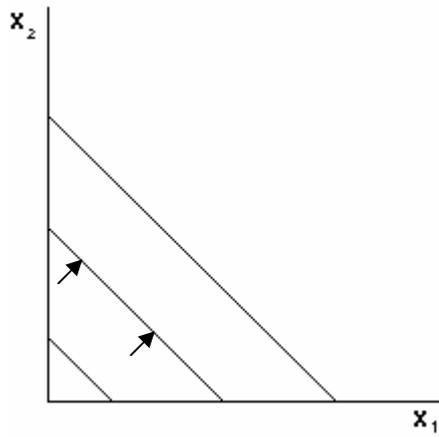


Figura 7a. Perfetti sostituti

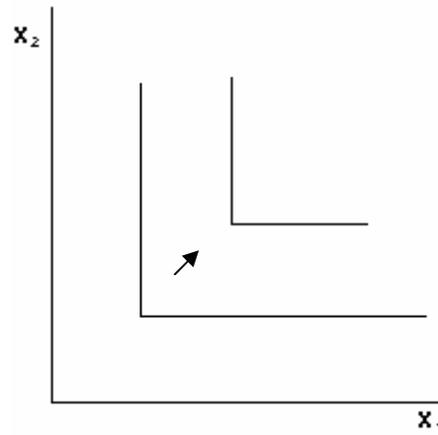


Figura 7b. Perfetti complementi

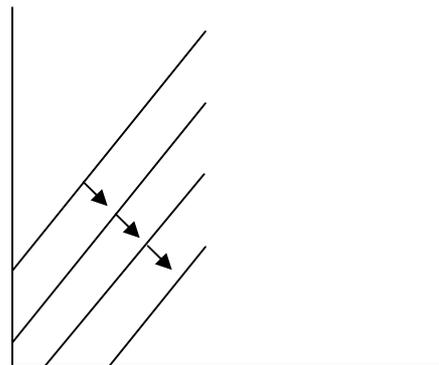


Figura 7c. Mali

Ulteriore classificazione definisce come *neutrale* il bene verso cui il consumatore prova indifferenza tra l'acquistare o non acquistare.

All'interno dei panieri acquistabili può esserci un punto definito "di *sazietà*", che rappresenta quella coppia di beni preferita ad ogni altra in termini di soddisfazione.

Elencati i diversi tipi delle curve di indifferenza, si definisce la classe delle preferenze regolari, a cui occorre fare riferimento nello studio della teoria del consumatore.

3.7 Preferenze well-behaved

Nella categoria di beni appartenenti alle preferenze regolari o well-behaved, si pongono una serie di condizioni restrittive al fine di rendere più facilmente studiabili le diverse curve di indifferenza.

Innanzitutto si impone il rispetto dei vincoli posti dalla teoria assiomatica, che comporta una serie di semplificazioni.

Trattare esclusivamente beni, e non mali, rispondenti al principio di non sazietà, implica l'avere curve di indifferenze monotone. Sempre l'assioma di non sazietà garantisce che, quantitativi maggiori di un determinato bene, comportino un livello maggiore di soddisfazione, a parità di quantità dell'altro bene:

$$(x_1 + \delta x_1, y_1) \succ (x_2, y_2).$$

Dall'ipotesi di monotonicità deriva che le curve di indifferenza hanno inclinazione sempre negativa, per rimanere sul medesimo livello di indifferenza dovremo infatti, qualora si vada ad aggiungere un'unità di un determinato bene, rinunciare ad una quota dell'altro bene.

Ulteriore conseguenza delle precedenti ipotesi è che presi due panieri, collocati su di una medesima curva di indifferenza, il paniere medio tra i due, sarà su una curva di indifferenza superiore alla pre-

cedente, ciò implica che l'insieme dei panieri preferibili a (x_n, y_m) è convesso.

Qualora le preferenze siano strettamente convesseⁱⁱ, non solo il paniere medio, ma anche una qualunque media ponderata di due panieri indifferenti, è un paniere strettamente preferito ai due panieri estremi:

$(x_1, y_1) \approx (x_2, y_2)$ avremo

$$\left(\frac{(x_1 + x_2)}{n}, \frac{(y_1 + y_2)}{n} \right) \succ (x_1, y_1)$$

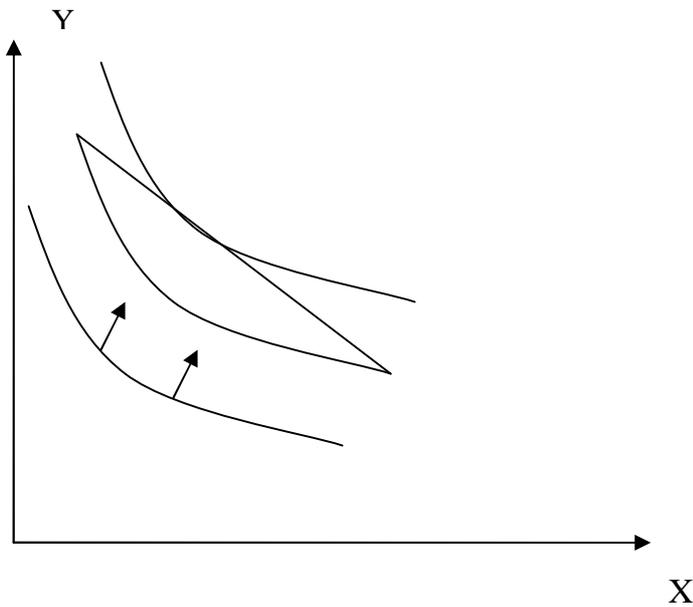


Figura 8. Preferenze well-behaved

3.8 Saggio marginale di sostituzione MRS

Graficamente il saggio marginale di sostituzione rappresenta l'inclinazione della retta tangente alla curva di indifferenza in un determinato punto, economicamente è la disponibilità, a parità di soddisfazione, di rinunciare ad una marginale quantità di un bene in cambio dell'altro.

Matematicamente è quindi la derivata prima nel punto considerato. Nelle curve di indifferenza con preferenze well-behaved si avrà un saggio marginale decrescente a cui soggiace un semplice concetto economico: maggiore è la quantità a disposizione di un determinato bene, tante più unità si è disposti a scambiare con il secondo bene.

Il *Saggio marginale di sostituzione* (o Marginal Rate of Substitution, MRS) è, in qualsiasi punto di una curva di indifferenza, il valore assoluto della pendenza della curva in quel punto.

La figura n.9 evidenzia che, se il consumatore vuole rimanere sulla stessa curva di indifferenza, dovrà rinunciare a Δy unità di y solo se in cambio riceve Δx unità di x , passando così da A a B, quindi:

$$MRS_{xy} = \Delta y / \Delta x$$

La formula esprime quante unità di y si è disposti a rinunciare in cambio di una unità di x , a parità di soddisfazione, ciò è detto *beneficio marginale di 1 unità di x in termini di y* .

Dall'ipotesi di convessità, deriva che il MRS è decrescente in valore assoluto

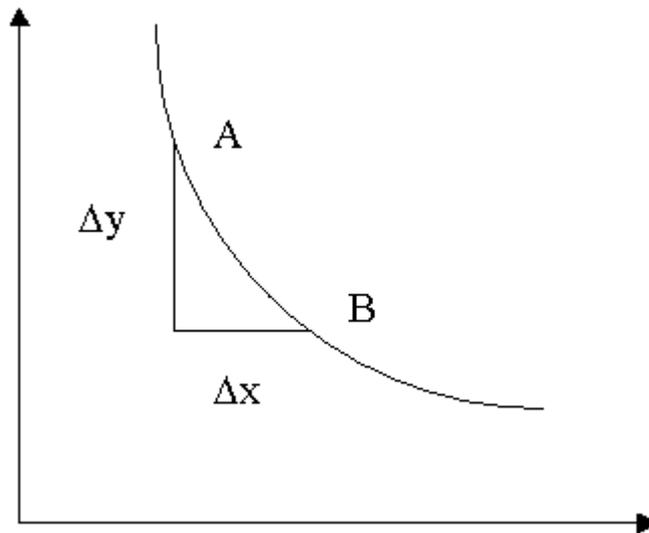


Figura 9. Saggio Marginale di Sostituzione

3.9 La scelta del paniere ottimo

Il vincolo di bilancio specifica quali panieri possono essere acquistati, l'assioma di non sazietà assicura che il paniere preferito deve stare sulla linea di bilancio. Il paniere preferito non può trovarsi su una curva di indifferenza, che giace almeno in parte all'interno del vincolo di bilancio, perché la non sazietà e la transitività sarebbero violate. Il paniere preferito deve trovarsi in una curva di indifferenza che ha un solo punto di contatto col vincolo di bilancio, cioè che è ad esso tangente.

Nel punto di ottimo avremo quindi che la pendenza del vincolo di bilancio è uguale alla pendenza della curva di indifferenza:

$$MRS_{xy} = P_x/P_y$$

Se $MRS_{xy} \neq P_x/P_y$, il consumatore non è in un punto tale da massimizzare la propria utilità.

Per esempio qualora il consumatore si fermi in un punto in cui il suo $MRS_{xy} = 1$ e $P_x/P_y = 2$, il consumatore può essere compensato per la rinuncia ad una unità di x con una unità di y , mentre se vende sul mercato una unità di x , riceve in cambio due unità di y . Acquistando meno x e più y , egli sarà sicuramente più soddisfatto.

Il concetto di scelta ottima pone la necessità di inserire nel modello una nuova variabile: l'utilità o meglio la funzione di utilità, che andrà ad esprimere i benefici delle diverse scelte per un consumatore.

3.10 L'utilità

L'indice di preferenza ci aiuta a dare un significato alle varie scelte del consumatore, occorre però attribuire un ordine alle possibili scelte.

L'aver definito l'insieme di beni tra loro indifferenti non è sufficiente a quantificare quanto un bene sia realmente preferito ad un altro, e quindi quantificare la soddisfazione del consumatore.

Definire l'utilità è da sempre in economia, e non solo, motivo di grandi discussioni, stabilire quanto è preferito un oggetto ad un altro, in termini rigorosi e razionali, è un compito arduo se non addirittura impossibile.

Già in era vittoriana, filosofi ed economisti parlavano dell'utilità come dell'indicatore del benessere complessivo: si riteneva cioè che l'utilità fosse la misura numerica della felicità della persona. Partendo da questo concetto era naturale pensare che il consumatore

complesse scelte per massimizzare la sua utilità, tali da renderlo il più felice possibile.

Gli economisti classici però non hanno mai descritto effettivamente come l'utilità possa essere misurata.

Come si può infatti misurare la quantità di utilità associata a scelte diverse?

A causa di questi problemi concettuali, nell'ultimo secolo, si è affermata la teoria comportamentale del consumatore. La stessa è formulata in termini di preferenze del consumatore e l'utilità è interpretata come un modo di descrivere le preferenze stesse.

3.11 Utilità cardinale e utilità ordinale

Le teorie che attribuiscono un significato alla grandezza dell'utilità, e che cercano di misurarne la differenza, sono dette dell'utilità cardinale: si fondano sull'ipotesi che la differenza tra l'utilità di due panieri diversifichi il livello di preferenza di uno rispetto all'altro e quindi la scelta del consumatore verso un certo paniere; ad ogni singola scelta associano un livello di utilità cardinale, definendo i diversi livelli di soddisfazione per il consumatore. A questa categoria appartengono le teorie dell'utilità dell'era vittoriana.

Le teorie cardinali trovano grandi difficoltà nel definire l'utilità dei singoli beni; per questo le teorie più recenti evitano di associare un peso all'utilità, ma danno solo una graduatoria.

A queste appartengono le teorie dell'utilità ordinale.

3.11.1 Utilità ordinale

La funzione di utilità associa un numero ad ogni singolo paniere di consumo, tale che ai panieri preferiti siano associati numeri più elevati. La caratteristica fondamentale della funzione di utilità è il modo in cui ordina i panieri di beni. Ai singoli valori non è dato un significato cardinale, ma solo un valore ordinale.

La mappa d'indifferenza è ricavata dalla relazione che esiste fra le quantità dei beni compresi nei vari panieri acquistabili e la soddisfazione che il consumatore ne ricava.

La relazione fra l'utilità e le quantità dei beni del paniere è la funzione di utilità.

Nel caso di due beni F e S , indicando l'utilità con U , la funzione di utilità si esprime con:

$$U(F, S)$$

U indica l'utilità, cioè il grado di soddisfazione del consumatore.

Una visione cardinale dell'utilità non è necessaria per descrivere la scelta razionale del consumatore; occorre poter stabilire che un dato grado di utilità è maggiore per un paniere rispetto ad un altro (il primo verrà preferito) o uguale fra i due (sono indifferenti).

Con U indichiamo, per qualunque paniere composto da F e da S , la posizione in graduatoria rispetto agli altri possibili panieri acquistabili.

Nella fase di scelta il consumatore è posto di fronte a due possibili panieri, se preferisce il primo al secondo sarà poiché il grado di utilità è maggiore.

La teoria dell'utilità prevede alcuni enunciati:

1. se $u(x_A, x_B)$ rappresenta alcune particolari preferenze allora sarà $u(x_A, x_B) > u(y_A, y_B)$, se e solo se $(x_A, x_B) \succ (y_A, y_B)$.
2. Se $f(u)$ è una trasformazione monotona, allora $u(x_A, x_B) > u(y_A, y_B)$, se e solo se $f(u(x_A, x_B)) > f(u(y_A, y_B))$.
3. $f(u(x_A, x_B)) > f(u(y_A, y_B))$ se e solo se $(x_A, x_B) \succ (y_A, y_B)$, e quindi la funzione $f(u)$ rappresenta le preferenze esattamente come la funzione di utilità di partenza $u(x_A, x_B)$. Una trasformazione monotona di una funzione di utilità corrisponde ad una funzione di utilità che rappresenta le stesse preferenze della funzione di utilità di partenza.

La funzione di utilità è matematicamente un fascio di curve di indifferenza, per ogni singola curva di indifferenza è associata l'utilità ordinale.

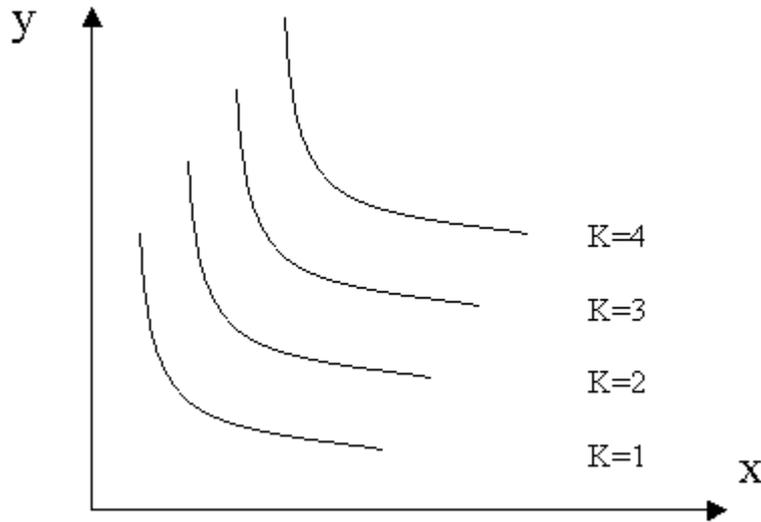


Figura 10. Funzione di utilità

La funzione di utilità è un modo per associare un numero ad ogni possibile paniere di consumo, tale che ai panieri preferiti siano assegnati numeri più elevati.

Formalmente, se il soggetto consuma il paniere (x_A, x_B) , la funzione di utilità $u(x_A, x_B)$ indica l'utilità al variare della quantità dei due beni, in modo tale che (x_A, x_B) è preferito a (x'_A, x'_B) se e solo se $u(x_A, x_B) > u(x'_A, x'_B)$.

Segue che a tutti i panieri, posti su una stessa curva di indifferenza, è associato lo stesso valore e che, a curve di indifferenza diverse, sono associate utilità differenti.

L'interpretazione dei valori di utilità comporta che, ad esempio, se $u(x_A, x_B) = 600$ mentre $u(x'_A, x'_B) = 60$ non significa che (x_A, x_B) è 10 volte preferito a (x'_A, x'_B) . Ciò per la natura ordinale dell'utilità di cui abbiamo ampiamente trattato.

3.12 Utilità marginale

Definito un paniere, per il consumatore, misuriamo la funzione di utilità cui associare un valore di preferenza. L'analisi dei valori di utilità, prossimi al paniere di partenza, permettono di verificare a quale livello di soddisfazione giunge il consumatore rispetto alle altre possibili scelte.

Siamo interessati a misurare la variazione dell'utilità di un consumatore che consuma un paniere di beni (x_A, x_B) , quando aumentiamo di poco il consumo del bene A. Formalmente, tale saggio di variazione è dato dal rapporto:

$$\frac{\Delta U}{\Delta x_A} = \frac{U(x_A + \Delta x_A, x_B) - U(x_A, x_B)}{\Delta x_A} \quad (1)$$

Si tratta della variazione dell'utilità, quando si varia di una piccola quantità, la disponibilità del bene A. Si noti che la quantità del bene B è mantenuta costante nel calcolo del saggio di variazione dell'utilità del bene A. Il rapporto del membro di destra, per Δ che tende a 0, è l'utilità marginale del bene A ed esprime il saggio al quale varia l'utilità del consumatore quando aumentiamo al margine la quantità del bene A mantenendo invariata la quantità del bene B. Indichiamo con MU_A l'utilità marginale del bene A.

La variazione dell'utilità associata ad una variazione infinitesima del consumo del bene A è semplicemente il prodotto della variazione del consumo (Δx_A) per l'utilità marginale del bene:

$$MU_A \Delta x_A.$$

In modo analogo si definisce il saggio di variazione dell'utilità generato da una variazione del consumo del bene B :

$$\frac{\Delta U}{\Delta x_B} = \frac{U(x_A, x_B + \Delta x_B) - U(x_A, x_B)}{\Delta x_B} \quad (2)$$

Quando Δ tende a 0, troviamo l'utilità marginale del bene B che indichiamo con MU_B . La variazione dell'utilità associata alla variazione marginale del consumo del bene B è:

$$MU_B \Delta x_B. \quad (3)$$

Il saggio marginale di sostituzione (RMS) è l'inclinazione delle curve di indifferenza e può essere espresso in termini di utilità marginali.

Consideriamo una variazione del consumo di ciascun bene (Δx_A , Δx_B) che mantenga costante l'utilità del consumatore consentendogli di spostarsi lungo una curva di indifferenza. Deve quindi essere:

$$\Delta U = MU_A \Delta x_A + MU_B \Delta x_B = 0 \quad (4)$$

Per Δx_A e Δx_B molto piccoli, otteniamo la pendenza della curva di indifferenza:

$$dx_A/dx_B = - MU_A / MU_B \quad (5)$$

la quale, essendo uguale al saggio marginale di sostituzione, ci permette di esprimerlo come utilità marginale.

$$\text{MRS} = - \text{MU}_A / \text{MU}_B \quad (6)$$

La formula fornisce un metodo per calcolare il saggio marginale di sostituzione indicando come deve avvenire lo scambio tra il bene A ed il bene B, in modo tale da mantenere costante l'utilità del consumatore, cioè rimanendo sulla stessa curva di indifferenza.

Esiste un altro modo per interpretare il saggio marginale di sostituzione, senza far ricorso al concetto di utilità marginale. Abbiamo detto che il MRS rappresenta il saggio al quale il consumatore è al limite disposto a sostituire un bene all'altro, ad esempio cedendo A per B. Questo equivale ad affermare che il consumatore è al margine disposto a pagare una parte del bene A per poter acquistare una determinata quantità del bene B, mantenendo inalterata la sua utilità. Il MRS può essere considerato come la disponibilità marginale a pagare.

3.13 Scelta ottima

Se la scelta del consumatore non fosse soggetta al vincolo di bilancio, probabilmente sceglierebbe il paniere che gli consente di consumare la maggior quantità possibile di entrambi i beni. Invece, dobbiamo determinare la scelta ottima (x_A^*, x_B^*) che massimizza il

benessere del consumatore compatibilmente con il vincolo di bilancio.

Il paniere ottimo sarà quello che si trova sulla curva di indifferenza più alta possibile dato il vincolo di bilancio.

Segue che, in corrispondenza del paniere ottimo (x_A^*, x_B^*) , si verifica la condizione di tangenza tra la retta di bilancio e la curva di indifferenza (Figura 11). Nel caso di preferenze strettamente convesse questa è una condizione necessaria e sufficiente affinché un paniere (x_A, x_B) sia ottimo.

Da un punto di vista geometrico, in corrispondenza di (x_A^*, x_B^*) , l'inclinazione della curva di indifferenza (MRS) deve essere uguale alla inclinazione della retta di bilancio, quindi:

$$\text{RMS} = -\frac{P_A}{P_B} \quad (7)$$

In altri termini, il saggio marginale di sostituzione del consumatore deve essere uguale al saggio di scambio tra il bene B e il bene A proposto dal mercato.

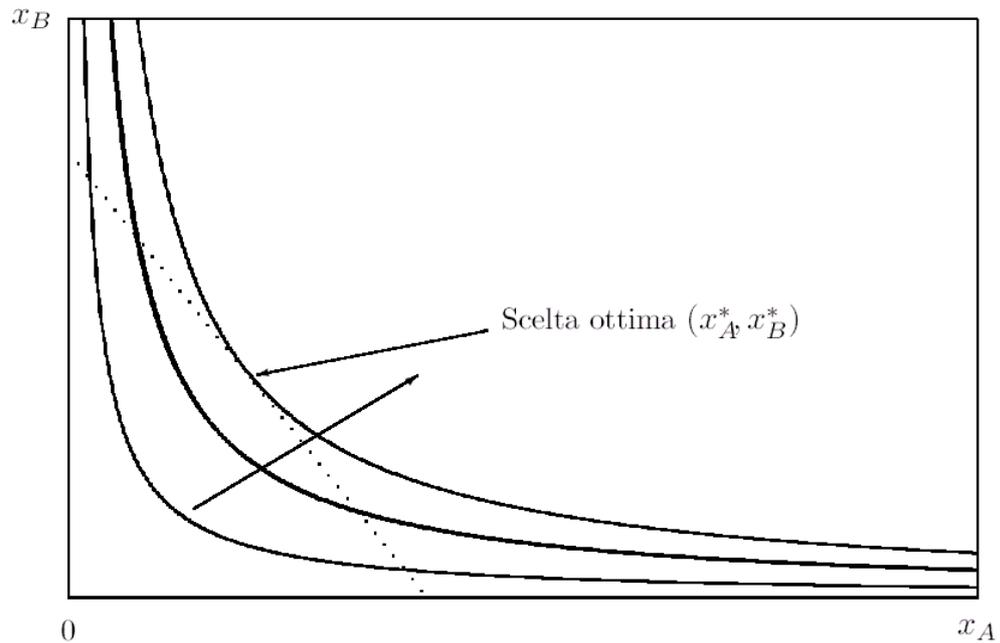


figura 11. Scelta ottima

Dalla (6), con le opportune semplificazioni, la condizione di ottimo diventa:

$$\frac{MU_A}{MU_B} = \frac{P_A}{P_B} \quad (8)$$

Il problema del consumatore può essere risolto anche algebricamente.

3.14 La funzione di utilità Cobb-Douglas

Una funzione di utilità comunemente usata è la funzione di utilità Cobb-Douglas:

$$u(x_A, x_B) = x_A^a x_B^b \quad (9)$$

dove a e b sono numeri positivi. Le curve di indifferenza Cobb-Douglas godono delle proprietà delle preferenze accennate nelle sezioni precedenti, in particolare sono decrescenti e convesse.

Mostriamo due metodi alternativi per studiare il problema del consumatore: il primo è la condizione di tangenza e il secondo è la risoluzione del problema della massimizzazione.

Primo metodo, soluzione del problema del consumatore attraverso la condizione di tangenza:

calcolando il tasso marginale di sostituzione in termini di rapporto tra utilità marginali, l'RMS della funzione di utilità Cobb-Douglas è:

$$RMS = -\frac{MU_A}{MU_B} = -\frac{ax_A^{a-1}x_B^b}{bx_B^{b-1}x_A^a} = -\frac{a}{b} \frac{x_B}{x_A} \quad (10)$$

Sappiamo che la scelta ottima del consumatore deve verificare la seguente condizione:

RMS = - P_A/P_B e quindi la condizione di ottimo diventa:

$$\frac{a}{b} \frac{x_B}{x_A} = \frac{p_A}{p_B} \quad (11)$$

Inoltre, nel punto di ottimo (x_A^*, x_B^*) , è soddisfatto il vincolo di bilancio dato da:

$$p_A x_A + p_B x_B = m. \quad (12)$$

Abbiamo quindi un sistema di due equazioni, (11) e (12), in due incognite, x_A e x_B , risolvibile con il metodo di sostituzione. Da cui otteniamo:

$$\begin{aligned} x_A^* &= \frac{a}{a+b} \frac{m}{p_A} \\ x_B^* &= \frac{b}{a+b} \frac{m}{p_B} \end{aligned} \quad (13)$$

La (13) rappresenta il paniere ottimo.

Secondo metodo, massimizzazione vincolata dell' utilità:

Risolviamo il problema di massimizzazione vincolata usando le condizioni per trovare un punto di massimo. Il problema della massimizzazione dell' utilità vincolata nel caso di funzioni di utilità Cobb-Douglas diventa:

$$\max_{x_A, x_B} x_A^a x_B^b \quad (14)$$

tale che $p_A x_A + p_B x_B = m$.

Vogliamo quindi trovare quei valori di x_A e x_B che diano il valore di $u(x_A, x_B)$ più alto possibile e che soddisfino contemporaneamente il vincolo di bilancio del consumatore.

Uno dei modi per risolvere un problema di massimizzazione vincolata è quello di esplicitare una delle due variabili in termini dell'altra nel vincolo di bilancio e poi sostituirla nella funzione di utilità. Si tratterà poi di risolvere un problema di massimizzazione libera in una variabile.

Esplicitiamo il vincolo di bilancio rispetto a x_B

$$x_B(x_A) = \frac{m}{p_B} - \frac{p_A}{p_B} x_A \quad (15)$$

Utilizziamo il vincolo di bilancio così come è espresso nella (15) e sostituiamolo nella funzione di utilità. Il problema diventa:

$$\max_{x_A} x_A^a \left(\frac{m}{p_B} - \frac{p_A}{p_B} x_A \right)^b \quad (16)$$

Questo è un problema di massimizzazione non vincolata nella sola variabile x_A . La condizione del primo ordine per il caso Cobb-Douglas è ottenuta differenziando la (16) rispetto a x_A e uguagliando a 0 l'espressione ottenuta.

Risulta quindi:

$$ax_A^{a-1} \left(\frac{m}{p_B} - \frac{p_A}{p_B} x_A \right)^b - x_A^a b \left(\frac{m}{p_B} - \frac{p_A}{p_B} x_A \right)^{b-1} \frac{p_A}{p_B} = 0 \quad (17)$$

che con opportune semplificazioni diventa:

$$\frac{a}{x_A} - b \frac{p_B p_A}{m - p_A x_A p_B} = 0 \quad (18)$$

esplicitiamo rispetto a x_A :

$$x_A^* = \frac{a}{a+b} \frac{m}{p_A}$$

Sostituiamo nel vincolo di bilancio il valore x_A^* appena trovato ed otteniamo:

$$x_B^* = \frac{b}{a+b} \frac{m}{p_B}$$

x_A^* e x_B^* sono i punti di ottimo ricercati, e sono uguali a quelli ottenuti con il primo metodo (13).

3.15 Critiche al modello del consumatore

Secondo la definizione di vincolo di bilancio il consumatore acquista due soli beni spendendo tutto il reddito disponibile, ciò appare contraddittorio con il comportamento reale.

Per rimediare a questa semplice critica si è cercato di estendere il significato dei due beni: il primo rappresenta ciò che effettivamente si vuole acquistare, il secondo tutti i beni rimanenti che il consumatore compra abitualmente.

Da cui:

X è il bene che si vuole acquistare,

Y è la somma di n beni che si vogliono comunque acquistare:

$$P_y Y = \sum_{i=1}^n P_{y_i} y_i$$

Avremo che P_x è il prezzo del bene X e P_y la sommatoria dei prezzi dei restanti beni che si desidera acquistare.

Una restrizione simile comporta che il prezzo del bene x è notevolmente inferiore di tutto quanto si desidera acquistare abitualmente, e la stessa quantità del bene è ridotta rispetto al totale delle quantità acquistata dal bene Y.

La retta del vincolo di bilancio è quindi molto schiacciata sull'asse delle X.

Si potrebbe arrivare a ritenere il vincolo di bilancio quasi piatto, specie se si considerano alcuni beni il cui costo rapportato alla somma degli altri beni è praticamente trascurabile.

Già questa semplice critica riduce la capacità esplicativa del modello.

Nella realtà difficilmente le variazioni di reddito creano un proporzionale aumento di acquisto della quantità dei singoli beni.

L'elasticità del consumatore è strettamente legata alla ricchezza, al reddito, ed al tipo di bene che si vuole acquistare.

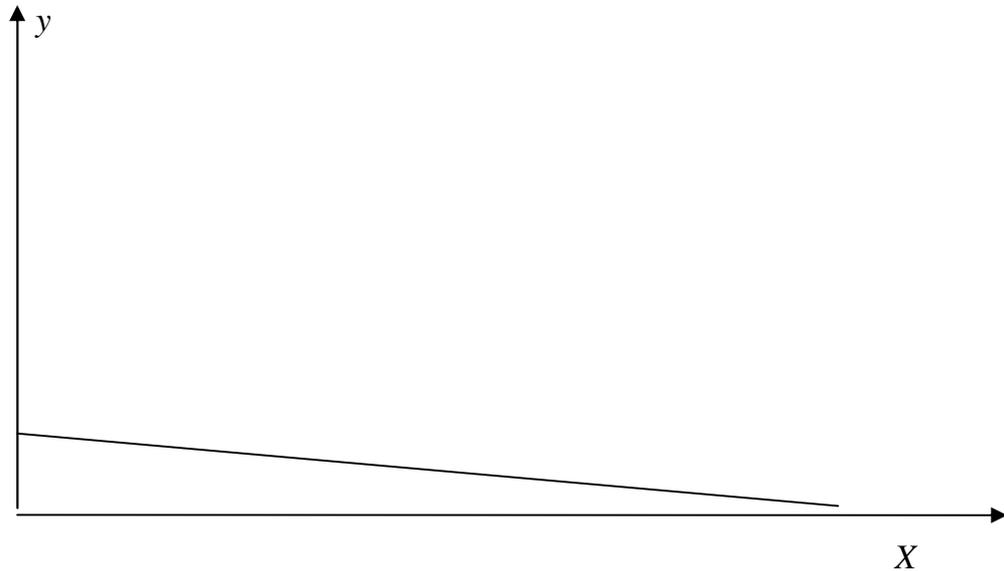


Figura 12. Vincolo di bilancio: un bene restanti beni

Altrettanto ragionevole è ritenere che l'aumento di prezzo di un singolo bene non generi automaticamente variazioni nella quantità acquistata dell'altro bene.

Il concetto di preferenza introdotto nella teoria è in linea generale condivisibile e riscontrabile, è ragionevole ritenere che il consumatore nello scegliere i beni da acquistare si comporti in base a certe preferenze e gusti.

Di più difficile comprensione il voler legare le preferenze all'acquisto non di un bene ma di due o più. Nella società moderna il concetto di preferenza ha acquistato sempre più un forte senso di precarietà, legato sempre più a mode, pubblicità, innovazione tecnica, distanza. La staticità del modello sempre più si scontra con la realtà del mercato.

La teoria assiomatica delle preferenze e diretta conseguenza dell'impostazione neoclassica della economia la quale merita una accurata riflessione.

L'impianto neoclassico pone una serie di vincoli comportamentali all'agente tra cui :

il concetto di aspettativa razionale ovvero effettuare le scelte più ragionevoli tra quelli potenziali, e la ragionevolezza discende a sua volta da un comportamento responsabile e maturo del consumatore e dalla piene conoscenza del mercato.

E' ragionevole ritenere un comportamento simile?

Sicuramente no, l'agente economico non può avere, piena conoscenza del mercato oltre che per ovvie barriere informative anche per questioni di convenienza economica, la ricerca costa e spesso termina non con il risultato migliore ma con quello più soddisfacente.

La teoria assiomatica segue lo stesso filone logico imponendo condizioni indispensabili per semplificare il modello. L'ipotesi di piena spendibilità del reddito, comporta grandi semplificazioni matematico concettuali, ma anche di verifica empirica. Lo stesso si evidenzia nell'assioma di sazietà per cui più è sempre meglio, comportamento in forte contrasto con il comportamento reale.

Gli assiomi di riflessibilità e di transitività trovano fondamento anche essi nel concetto di razionalità: il consumatore essendo razionale non potrà che essere coerente, ma ciò è nuovamente troppo semplicistico. Le interazioni con l'esterno creano troppa volatilità nei gusti, il fenomeno delle mode bene evidenzia tali comportamenti.

Nella definizione di quantità ottima quale punto di tangenza tra vincolo di bilancio e curve di utilità la necessità di semplificare il calcolo matematico del massimo vincolato ha comportato l'introduzione di numerose limitazioni alle curve di utilità, indispensabili per rendere risolvibile il vincolo.

Il modello del consumatore ha numerose lacune, le quali discendono direttamente dalla rigorosa impostazione matematico statica. Affiancare al modello una simulazione che illustri il comportamento del consumatore permette di superare molti dei limiti concettuali esposti precedentemente.

Note al capitolo

ⁱ Condizione strettamente legata alla legge di Gossel secondo la quale: di nessun bisogno devono essere soddisfatte fasi meno intense quando esistono di un altro bisogno fasi più intense.

ⁱⁱ Curva strettamente convessa implica oltre alla convessità un andamento strettamente curvilineo. Le curve di indifferenza di due beni perfetti sostituti sono convesse ma non strettamente.

Capitolo 4

L'economia e la complessità

4.1 La complessità

Il concetto di complessità appartiene a pieno titolo al vocabolario degli economisti. Al termine possono essere assegnati diversi significati: da sinonimo di complicato e difficile da risolvere, fino all'interpretazione in termini matematici di complessità delle dinamiche caotiche.

Da alcuni anni una maggiore apertura al dialogo interdisciplinare ha permesso di chiarire quale sia il corretto significato da attribuire al concetto di complessità.

E' utile, innanzitutto, differenziare complicato da complesso (Terna 2000).

Complicato è qualcosa che può essere ricondotto ad una somma di azioni semplici, per esempio il motore a scoppio; il suo funzionamento è complicato, ma di per sé comprensibile perché scomponibile in varie fasi più semplici.

Il fenomeno complesso è, invece, non scomponibile: per esempio il principio di autoregolazione termica dei formicai; noi non conosciamo quali siano i fenomeni che lo generano ed in quale sequenza vadano ad agire.

Si pensi anche secondo Per Back (1996) ad un cumulo di sabbia, che è versato piano attraverso un imbuto. L'ordine con cui cadono i granelli di sabbia nell'imbuto è imprevedibile, ed è quindi un fenomeno complesso.

Si può affermare che, in via del tutto generale, un sistema complesso è composto da un numero elevato di parti distinte che interagiscono tra loro, ma che non possono essere analizzate separatamente. L'analisi separata non permetterebbe di spiegare, ma soprattutto di prevedere, i fenomeni che si osservano dall'esterno.

Una definizione di complessità è data da Waldrop (1992)

"[...] At a kind of abstract phase transition called "the edge of Chaos", you also find complexity: a class of behaviours in which the components of the system never quite lock into place, yet never quite dissolve into turbulence, either. These are systems that are both stable enough to store information, and yet evanescent enough to transmit it. These are the system that can be organized to perform complex computations, to react to the world, to be spontaneous, adaptive, and alive."

Day (1994) analizza la complessità sotto l'aspetto esclusivamente matematico definendola come un sistema dinamico, che dal punto di vista endogeno non tende asintoticamente ad un punto fisso. Tale fenomeno può mostrare un comportamento discontinuo ed è descritto da un sistema di equazioni differenziali o di equazioni alle differenze finite, con elementi stocastici.

Una definizione di questo tipo, osserva Rosser (1999), non tiene conto del fatto che in economia emergono delle complicazioni non previste causate dall'interazione di calcoli umani nelle decisioni.

Kauffman (1995) sostiene:

"[...] Just near this phase transition, just at the edge of chaos, the most complex behaviour can occur-orderly enough to ensure stability, yet full of flexibility and surprise. Indeed, this is what we mean by complexity."

Come si può osservare esistono numerose definizioni di complessità, ciò per la difficoltà di delimitare un preciso ambito di applicazione della medesima.

Tuttavia la definizione di Day è l'unica che permette di includere le teorie sui sistemi dinamici non lineari: la teoria della cibernetica, la teoria sulle catastrofi, la teoria del caos e la teoria della complessità. Queste costituiscono le quattro correnti di studio sulla non-linearità:

- La teoria della cibernetica con Forrester (1960) evidenzia il fatto che nei sistemi multipli di equazioni non lineari emergano risultati sorprendenti e controintuitivi. Ciò ha formato il principio fondamentale dallo studio sui modelli dinamici fino a quelli più moderni sulla complessità.
- La teoria sulle catastrofi elaborata da Thom (1975) identifica sette modelli geometrici di catastrofi di base. Una catastrofe è un particolare tipo di discontinuità in un certo sistema. Le

discontinuità dipendono da diversi punti di equilibrio, comportando il passaggio da un punto all'altro al variare dei parametri.

- La teoria del caos studia le dinamiche caotiche generate da un processo deterministico. I sistemi caotici sono estremamente sensibili alle condizioni iniziali e ad ogni più piccola perturbazione. Anche prendendo il via da condizioni iniziali vicinissime nello spazio si può dar luogo a traiettorie completamente diverse ed ogni instabilità viene amplificata. È questo il fenomeno che Lorenz (1979) definisce: “effetto farfalla”ⁱ. Diversamente dalle strutture dissipative, che “reggono” le perturbazioni ristabilendo la loro struttura e la loro autonomia, i sistemi caotici si “impadroniscono” immediatamente di una fluttuazione e l'amplificano fino ad estenderla all'interno del sistema, provocando comportamenti nuovi ed assolutamente inaspettati.
- La teoria della complessità racchiude tutte le premesse teoriche delle precedenti correnti. L'istituto per gli studi sulla complessità, di Santa Fe nel New Mexico, il Santa Fe Institute, suggerisce che un sistema complesso si basa su sei caratteristiche:
 1. interazione diffusa tra agenti eterogenei, che agiscono localmente ad uno spazio;

2. nessun controllore centrale del modello, sebbene siano consentiti deboli interazioni a livello globale;
3. organizzazione gerarchica multi-livello, con interazioni distribuite a diversi livelli;
4. adattamento continuo degli agenti che sono in grado di imparare ed evolvere;
5. continue novità come nuovi mercati, tecnologie, compromenti creano nuove nicchie nell'“ecologia” del sistema;
6. dinamica priva di un equilibrio globale, ma con molti punti di equilibrio instabili.

Il risultato è un ambiente caratterizzato da razionalità limitata e da aspettative non razionali.

Una volta chiariti gli aspetti definatori, risulta evidente che adottare il criterio della complessità nell'ambito dell'analisi economica, presuppone la necessità di porre una giusta attenzione al modo nel quale gli individui entrano in contatto e interagiscono tra loro.

4.2 Lo scenario di Santa Fe e le discipline economiche

La teoria sulla complessità tenta di cambiare la visione classica che si ha dell'economista, concentrando l'attenzione non sulla spiegazione del fenomeno economico, ma sulla comprensione della natura dello stesso.

L'analisi proposta dagli studiosi prende il via dall'assunzione di complessità, ed intorno a tale concetto fa ruotare più che una teoria, una tecnica di avvicinamento all'economia. Non si vuole spiegare un particolare evento, con un definito modello teorico; piuttosto si cerca di vedere i fatti da un punto di vista diverso.

Lo scenario della complessità è affrontato e studiato dal Santa Fe Institute (<http://www.santafe.edu>), l'approccio sopra esposto per inquadrare i fenomeni complessi è definito Santa Fe Scenary. Gli studiosi di complessità non cercano una definizione del concetto, ma solo la sua esemplificazione: definire la complessità implicherebbe perdere il significato intrinseco del termine, se definita diverrebbe non più complessa, ma solo complicata.

Il concetto di complessità va quindi ribaltato nella modellizzazione; obiettivo dell'economista sarà quello di spiegare i fenomeni senza definire, capire e controllare ogni singola variabile, ma cercando di comprendere, avvalendosi di modelli anche solo descrittivi, i significati dei singoli fatti.

Nell'ambiente economico si assunse consapevolezza del concetto di complessità al Santa Fe Institute sul finire degli anni ottanta:

*“[...] September 1987, twenty people came together at the Santa Fe Institute to talk about "the economy as an evolving, complex system." Ten were theoretical economists, invited by Kenneth J. Arrow, and ten were physicists, biologists, and computer scientists, invited by Philip W. Anderson. The meeting was motivated by the hope that new ideas bubbling in the natural sciences, loosely tied together under the rubric of "the sciences of complexity," might stimulate new ways of thinking about economic problems. For ten days, economists and natural scientists took turns talking about their respective worlds and methodologies. While physicists grappled with general equilibrium analysis and non cooperative game theory, economists tried to make sense of spin glass models, boolean networks, and genetic algorithms. The meeting left two legacies. (The first was a volume of essays, *The Economy as an Evolving Complex System*, edited by Arrow, Anderson, and David Pines).*

The other was the founding, in 1988, of the Economics Program at the Santa Fe Institute, the Institute's first resident research program. The Program's mission was to encourage the understanding of economic phenomena from a complexity perspective, which involved the development of theory as well as tools for modeling and for empirical analysis”.

4.3 Esempi di sistemi complessi.

Le immagini di seguito riportate richiamano modi diversi di rappresentare la complessità.

La complessità può presentarsi alla nostra osservazione sotto molteplici forme accomunate, però, da uno stesso meccanismo di fondo: l'interazione.

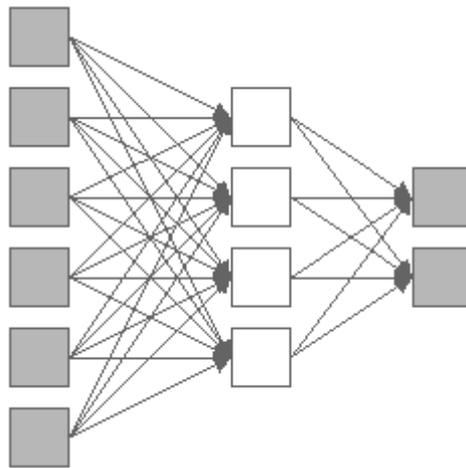


Figura 1. Schema di una rete neurale.

Le reti neurali artificiali sono utilizzate per comprendere e riprodurre formalmente i meccanismi fondamentali dei processi di apprendimento propri della complessa struttura di percezione sensoriale umana. Le reti neurali sono impiegate con successo nei compiti di classificazione di forme incerte, nel riconoscimento delle immagini, nella descrizione delle interazioni nelle strutture organizzative aziendali, nella previsione di fenomeni complessi evolutivi.

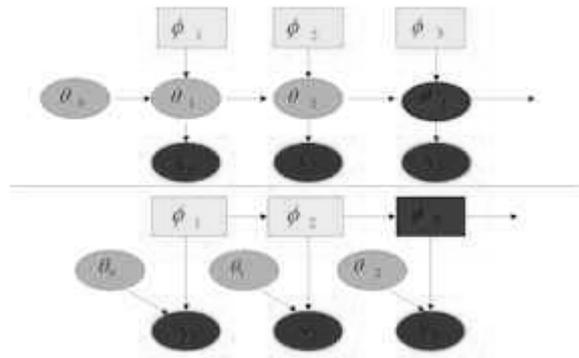


Figura 2. Schema di un sistema stocastico dinamico.

I sistemi stocastici dinamici paralleli interagenti descrivono con le probabilità le leggi evolutive di fenomeni economici, sociali e ambientali complessi. Essi permettono di far interagire l'evoluzione di variabili osservabili che vorremmo prevedere con la parallela evoluzione di variabili non osservabili esplicative.

		Giocatore 2	
		L ₂	R ₂
Giocatore 1	L ₁	1 , 1	5 , 0
	R ₁	0 , 5	4 , 4

Figura 3. Gioco del dilemma del prigioniero

Nel gioco del dilemma del prigioniero due giocatori, che non possono comunicare tra loro, hanno a disposizione due strategie, L e R.

Dalla tabella dei guadagni, rispettivamente del primo e del secondo giocatore, si vede chiaramente come per ciascun giocatore, qualunque sia la scelta dell'avversario, sia razionale giocare la strategia L. Ma il guadagno atteso della strategia L risulta nettamente minore di quello che si otterrebbe se i giocatori scegliessero entrambi R. Nella teoria dei giochi questo schema è stato intensamente utilizzato per discutere la complessità dell'interazione umana quando essa presenta una divaricazione tra razionalità individuale e razionalità collettiva.

4.4 La sfida della complessità.

Spesso avvertiamo la complessità di una situazione di fronte ad un evento sorprendente, ad un fatto che smentisce le nostre previsioni, anche quelle apparentemente più sicure e più competenti.

- La previsione dell'andamento dei prezzi di mercato dei titoli azionari. Quale sarà l'evoluzione dell'indice Nasdaq nelle prossime settimane?
- La previsione dei tempi di diffusione di una nuova tecnologia. Si pensi al commercio elettronico: quanto inciderà sul volume complessivo di scambi? Quali settori ne subiranno maggiormente l'impatto? Quali sono le forze che ne ostacoleranno l'espansione?

- La previsione della dinamica della concorrenza in certi mercati. Ad esempio, come si svilupperà la competizione nel mercato automobilistico europeo nei prossimi 5 anni? Quale effetto avranno le nuove tecnologie non inquinanti?
- La previsione delle nuove forme di organizzazione del lavoro. La quota del lavoro dipendente sul totale degli occupati continuerà a decrescere? Se sì, con quali ritmi?

In tutti questi casi si manifesta la nostra difficoltà a capire il comportamento di certi sistemi (la borsa valori, le conoscenze scientifiche, organizzative e pratiche sottostanti le applicazioni di una tecnologia, il sistema delle imprese e delle istituzioni che definiscono un certo ambiente concorrenziale, ecc.) e dunque a prevederne le traiettorie evolutive. Questi difetti di comprensione dipendono dal fatto che i sistemi di cui abbiamo parlato, sono sistemi complessi, vale a dire, come abbiamo già detto in precedenza, sistemi composti da parti reciprocamente interagenti di modo che il comportamento del sistema non può essere ricavato dalla semplice somma dei comportamenti delle singole parti. Le oscillazioni delle quotazioni del Nasdaq sono proprietà emergenti dei rispettivi sistemi perché esse dipendono dal modo specifico con cui si svolgono le interazioni tra le parti del sistema. Queste interazioni non si sviluppano secondo semplici sequenze lineari, ma danno luogo a complessi circuiti di feedback che possono ramificarsi in maniera imprevedibile, amplificando l'intensità di eventi apparentemente modesti che in tal modo possono agire con una forza insospettata su aree del sistema

distanti dal punto di origine. L'opacità dei sistemi complessi può generare comportamenti difforni da quelli previsti e può creare situazioni caratterizzate da notevoli ambiguità circa i corsi d'azione più razionali da assumere per cercare di influenzare la dinamica dei sistemi. Uno degli aspetti sorprendenti della moderna teoria dei sistemi dinamici è che anche sistemi strutturalmente semplici, cioè descrivibili con poche equazioni non lineari, possono dar luogo a dinamiche notevolmente complicate.

Metodologie in pieno sviluppo come quelle delle reti neurali, degli algoritmi genetici, degli automi cellulari e di reti probabilistiche, modelli grafici e sistemi esperti stanno contribuendo in maniera rilevante ad ampliare la nostra capacità di rappresentazione dei sistemi complessi, ed è vero che queste metodologie hanno già trovato concreta applicazione in diversi ambiti operativi. Ma, è anche vero che le tecniche di calcolo possono essere applicate intelligentemente solo contestualmente alla negoziazione sociale del consenso sul significato e sui possibili modi di soluzione dei problemi.

Per Simon la complessità del comportamento umano è in gran parte un riflesso della complessità dell'ambiente in cui l'uomo si trova. È fuor di dubbio che la rivoluzione, in pieno svolgimento, dei mezzi di calcolo e di comunicazione sta portando ad un aumento senza precedenti dell'ampiezza e della intensità delle interazioni tra persone, organizzazioni, istituzioni.

I contesti in cui si svolge l'azione umana, di conseguenza, diventano via via più complessi e, seguendo Simon, ciò obbliga le persone, le organizzazioni, le istituzioni a disegnare strategie più ampie ed articolate. Queste strategie diventano anche più rischiose perché at-

traverso la rete delle interazioni ogni intervento può portare a conseguenze emergenti di vasta portata. È dunque augurabile che la nuova mappa di categorie, di modelli concettuali e di strumenti che si sta formando attorno all'idea di complessità contribuisca a rendere l'azione degli imprenditori, dei manager e dei politici più ingegnosa e più responsabile.

Note al capitolo

ⁱ 29 dicembre 1979, il fisico Edward Lorenz presentò alla Conferenza annuale della *American Association for the Advancement of Science*, una relazione in cui ipotizzava come il battito delle ali di una farfalla in Brasile, a séguito di una catena di eventi, potesse provocare una tromba d'aria nel Texas. L'insolita quanto suggestiva relazione, diede il nome al cosiddetto *butterfly effect*, effetto farfalla. A questo punto, il lettore si chiederà se "l'effetto farfalla" è solo una suggestiva speculazione, oppure ha un riscontro reale...Nel corso di un programma di simulazione del clima, Lorenz fece un'inaspettata quanto importante scoperta. Una delle simulazioni climatiche si basava su dodici variabili, incluse relazioni non lineari. Lorenz scoprì che, ripetendo la stessa simulazione con valori leggermente diversi (una serie di dati veniva prima arrotondata a sei cifre decimali, e successivamente a tre), l'evoluzione del "clima" elaborata dal computer si discostava nettamente dai risultati precedenti: a quella che si configurava appena una perturbazione, dopo una effimera somiglianza iniziale, si sostituiva un modello climatico completamente diverso. Queste osservazioni hanno portato allo sviluppo della *Teoria del Caos* che pone limiti definiti alla prevedibilità dell'evoluzione di sistemi complessi non lineari. Nei sistemi lineari, una piccola variazione nello stato iniziale di un sistema (fisico, chimico, biologico, economico) provoca una variazione corrispondentemente piccola nel suo stato finale: per esempio, colpendo leggermente più forte una palla da biliardo, questa andrà più lontano. Al contrario, sono non lineari le situazioni di un sistema in cui piccole differenze nelle condizioni iniziali producono differenze non prevedibili nel comportamento successivo. Un sistema può anche comportarsi in modo caotico in certi casi e in modo non caotico in altri. Per esempio, da un rubinetto non chiuso le gocce cadono in una sequenza regolare; variando leggermente l'apertura del rubinetto, si può far sì che le gocce cadano invece in modo irregolare, appunto caotico. Ancora, il movimento regolare di un pendolo fissato ad un appoggio elastico, diventa caotico.

Capitolo 5

La Simulazione e

l'apprendimento dell'economia

5.1 Modelli didattici per lo studio dell'economia

La maggior parte degli economisti si avvale di due modelli classici per lo studio dell'economia:

- il modello letterario-descrittivo, con le finalità di spiegare i fenomeni economici,
- il modello matematico-statistico, con le finalità di analizzare e dimostrare.

L'analisi economica utilizzando i due modelli sopra citati, non tiene conto di un elemento essenziale, vale a dire la visione della realtà del fenomeno economico. A questo scopo è necessaria l'introduzione di un terzo tipo di modello:

- il modello della simulazione.

L'uso della simulazione in economia è ripartito in due settori:

1. il primo si rifà alla conduzione di esperimenti mentali difficili da portare avanti senza l'aiuto del calcolo ordinato, che un computer riesce a fornire;
2. il secondo, invece, si avvale della simulazione come strumento per una rappresentazione formale della realtà, soprattutto se la realtà da rappresentare risulta molto complicata.

La simulazione è uno strumento che serve a “calcolare” la realtà, ma se si cambiano i parametri di calcolo il risultato sarà l’“irrealtà”, la visione di un qualcosa che non esiste, o meglio che non esiste ancora, ovvero sarà una “previsione”.

In questo modello si capisce come realtà e previsione siano facilmente identificabili.

Burton (2001) individua nella simulazione lo strumento per creare “mondi” che esistono o che potrebbero esistere. Si può riprodurre un mondo nella sua realtà, ovvero “com’è”; inoltre, modificando alcuni parametri, si creano “mondi come potrebbero essere”, disegnando situazioni artificiali.

Un’analisi di questo tipo è garantita dalla versatilità degli strumenti della simulazione; una volta creato un modello è possibile effettuare diverse prove, esplorare aspetti nuovi, generare dati, ovvero effettuare un gran numero di ricerche.

Gli studi economici sono stati rivolti per circa due secoli all’approfondimento della visione di realtà dei fenomeni dell’economia, si sono susseguite innumerevoli ricerche.

L'economia classica si è sempre preoccupata di costruire degli schemi generali da cui far dipendere ogni conseguenza. La realtà è che questa scienza non deve tenere conto solo delle conseguenze che si verificano in seguito a certi fenomeni economici, ma deve studiare innanzitutto i soggetti fondamentali che operano in essa: gli individui.

Walras, percepisce l'idea che gli individui abbiano una parte attiva nei processi economici tramite un'analisi della specificità del metodo matematico (o metodo razionale). Le affermazioni walrasiane si riferiscono alla necessità di applicare il metodo matematico all'economia, ciò nell'ambito delle sue riflessioni epistemologiche. L'esposizione del metodo matematico come definito da Walras, permette dapprima di sgomberare il campo da un malinteso che lo vorrebbe “assiomatico” e, più in generale, di distinguerlo dalla “matematizzazione” nella sua accezione corrente. Si evidenziano due fasi nella costruzione scientifica walrasiana:

- 1) la definizione dei tipi ideali sulla base dei tipi reali;
- 2) la deduzione dai tipi ideali delle proposizioni costituenti la teoria del valore di scambio.

In questo contesto il compito della scienza è, esplicitamente indicato da Walras, nella formulazione di un ideale verso cui orientare la pratica. Secondo una tale visione i fatti economici sono percepiti dai soggetti, che li identificano come: prezzi, domanda, offerta. Il tutto è sottoposto ad un processo di astrazione, gli individui in que-

stione altro non sono che soggetti ideali sui quali si applica l'economia politica pura attraverso proposizioni matematiche.

Dagli studi di Walras fino ad oggi sono stati compiuti grandi passi, ma nonostante ciò permangono difficoltà teoriche e pratiche nella rappresentazione della realtà in economia.

5.2 Il progetto Swarm

Una via da percorrere per eliminare tali difficoltà è quella di basarsi sulla simulazione, riconoscendo l'economia come un sistema complesso.

La ricerca principale da effettuare è quella sui soggetti economici in relazione alla complessità. Si tratta di studiare l'azione dei soggetti economici verificando le conseguenze di ciò che si riesce a comprendere degli stessi, attraverso la simulazione.

La costruzione di questi modelli è facilitata dall'uso di particolari programmi cd. "ad oggetti". Un esempio fra tutti è fornito dal protocollo di programmazione denominato Swarm (<http://www.swarm.org>).

Il progetto Swarm, nome che significa sciame, nasce a metà degli anni '90, presso il Santa Fe Institute (<http://www.santafe.edu>) di Santa Fe nel New Mexico. Si tratta di una biblioteca di funzioni, sviluppate attraverso classi e metodi, modelli che reagiscono ad ordini. Il linguaggio di programmazione utilizzato è l'Objective C, e più recentemente il Java.

La scelta del nome non è stata casuale: Swarm gestisce attraverso le sue librerie ed attraverso la programmazione ad oggetti in Java o in

Objective C “sciame” di oggetti indispensabili per costruire modelli complessi utili per la comprensione dei fatti economici.

Con l'utilizzo di tale strumento è possibile spiegare i fenomeni economici, simulandoli. Si procede alla ricostruzione della realtà, in relazione ad un dato fenomeno economico, attraverso la compilazione di classi astratte di soggetti-oggetti che obbediscono a particolari ordini detti metodi. Il risultato sarà la creazione di un “mondo” all'interno del quale operano soggetti-oggetti che seguono le indicazioni fornite dallo sperimentatore o emergenti dall'esperimento.

5.3 Dal progetto Swarm, all'applicazione

Nel 1994 un gruppo di ricercatori, guidati da Chris Langton, dà inizio al progetto Swarm, con lo scopo di creare un supporto standardizzato per lo sviluppo di modelli di simulazione multiagente.

Nel 1995 si ottiene una prima versione, la seconda è del 1997, entrambe basate sulle architetture di sistema operativo UNIX/Solaris e Linux.

Nuove versioni sono state create per poter operare su diversi sistemi operativi quali DEC Alpha e Microsoft Windows; nell'ultimo caso Swarm si appoggia a Cygwin, che riproduce un ambiente Unix all'interno di Windows.

Swarm è utilizzato per la costruzione di modelli in cui interagiscono agenti elementari. Analizzando questi modelli è possibile far emergere quali siano i comportamenti degli individui, sotto condizioni determinate.

Nella fase iniziale della programmazione vengono riconosciute delle librerie che si riferiscono ad oggetti. Inoltre occorre costruire la classe che rappresenta le caratteristiche ed i comportamenti tipici dei singoli oggetti. Le applicazioni create in Swarm hanno carattere fortemente gerarchico. Si parte dal livello più elevato, rappresentato dall'observer Swarm che a sua volta definisce il model Swarm da cui sono creati e gestiti gli agenti individuali. Il model Swarm può analizzare, diffondere ed interpretare i dati forniti dai movimenti dei singoli individui, che hanno eseguito compiti ed attività da noi pianificate.

Ogni oggetto è combinato in sistemi strutturati ed ognuno è caratterizzato da uno comportamento, che è rappresentato da funzioni in grado di operare sui dati, attraverso i *metodi*, che sono le regole di comportamento dei singoli agenti. Gli oggetti e le relative modalità di interazione costituiscono gli elementi di base del programma. I singoli oggetti hanno specifici compiti da svolgere all'interno della struttura; gli stessi hanno facoltà di comunicare con gli altri oggetti attraverso delle chiamate e tramite la richiesta di esecuzione dei *metodi*.

Gli oggetti possono essere di diverso tipo, generati da classi differenti, ovvero dello stesso tipo e appartenere a classi uguali.

Fondamentale è l'eventuale carattere ereditario delle classi: un oggetto e la classe che lo contiene potranno ereditare caratteristiche e *metodi* propri di altre classi.

Gli oggetti di una classe vengono definiti una sola volta, ciò crea una "tipizzazione" dell'oggetto che potrà servire per la costruzione, successiva, di oggetti simili, identificabili con le istanze della clas-

se. Queste istanze hanno variabili e metodi comuni, ma hanno un proprio spazio di memoria, che contiene il set di variabili.

5.4 Lo schema ERA, Environment-Rules-Agent

Lo schema ERA (Terna 2000) è una guida per la costruzione degli oggetti che rappresentano gli agenti all'interno delle simulazioni.

Nella comunità scientifica, che si avvale di strumenti informatici avanzati, vi è spesso la difficoltà di rendere comprensibile i codici di programmazione. Utilizzando Swarm, avvalendosi della programmazione ad oggetti, è vantaggioso l'uso di un protocollo di lavoro.

Con lo schema ERA si separano e ordinano all'interno della programmazione i differenti passaggi logici con cui si costruisce il modello.

Lo schema ERA, riportato nelle figura 1, permette di gestire quattro diversi strati nella costruzione del modello e degli agenti.

- Un primo strato rappresenta l'ambiente in cui gli elementi sono chiamati ad interagire, questo corrisponde al modello Swarm e rappresenta il contesto all'interno del quale si definiscono gli agenti, si strutturano le liste, si individuano gli eventi nel tempo, si chiariscono i metodi, cioè le regole di interazione tra gli agenti interpretabili come messaggi che gli agenti sono in grado di gestire reagendo con azioni ed informazioni.

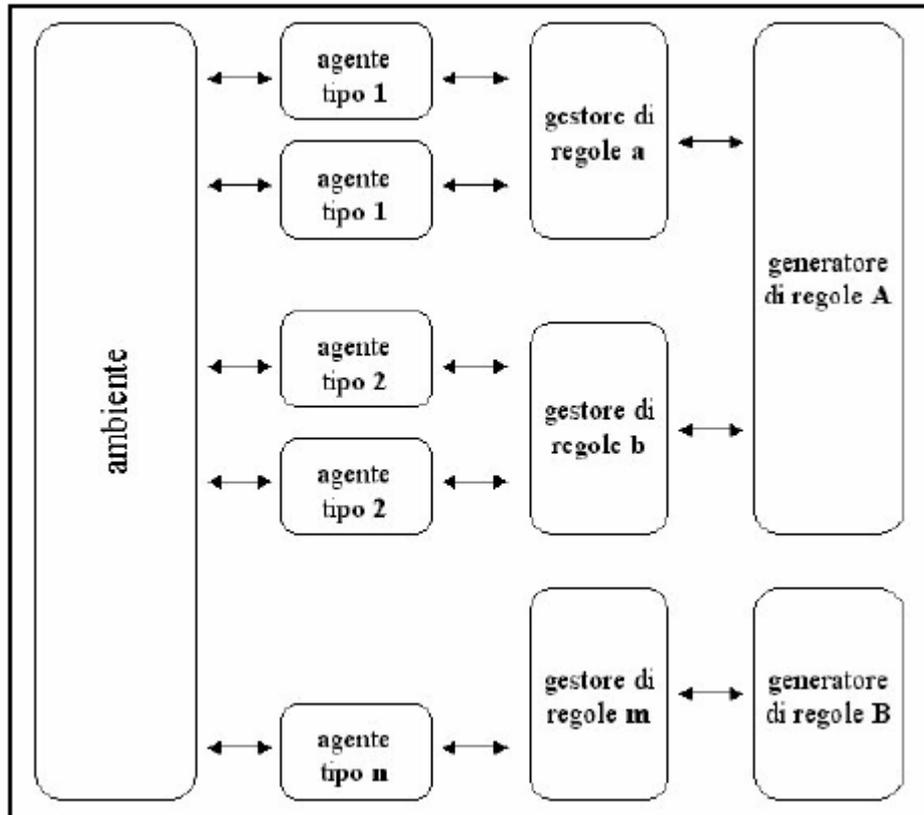


Figura 1. Schema ERA

- Un secondo strato è quello degli agenti, che possono essere costruiti attingendo alle proprietà, ai caratteri, ai dati ed ai metodi da classi più generali.
- Il terzo strato si occupa delle modalità attraverso cui gli agenti decidono il proprio comportamento. Ad ogni scelta, l'agente interroga un oggetto sovraordinato, definito "gestore di regole" (Rule Master), comunicandogli i dati necessari ed ottenendo le indicazioni di azione.

- Il quarto strato tratta la costruzione delle regole. Come gli agenti interrogano i gestori di regole, così i gestori di regole interrogano i generatori di regole (Rule Maker) per modificare la propria linea di azione.

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

Se il codice è scritto secondo tali modalità, risulterà più agevole compiere modifiche, si potrà sostituire le caratteristiche dei gestori di regole modificando semplicemente gli oggetti introdotti nel modello.

5.5 Java

Il linguaggio di programmazione e l'ambiente Java sono stati sviluppati dai ricercatori della Sun Microsystems al fine di risolvere i problemi di semplicità e di portabilità, comuni a molti linguaggi di programmazione moderni.

Java era parte di un progetto più generale il cui scopo consisteva nel portare software avanzato anche su macchine piccole, ma molto diffuse, come i PC. Inizialmente, i ricercatori della Sun avevano pensato di usare linguaggi già in uso, poi però problemi di complessità condussero all'implementazione di un nuovo linguaggio di programmazione.

Java è indipendente dalla piattaforma, completamente orientato all'oggetto e multi-threaded. Oltre a queste moderne caratteristiche,

esso dà la possibilità di creare applicazioni (applet) in grado di essere richiamate ed eseguite da apposite applicazioni client del World Wide Web.

In un programma Java tutte le istruzioni sono contenute nelle classi, le quali, una volta compilate, costituiscono ognuna un modulo a sé stante. In Java non esiste la possibilità di accedere direttamente alla memoria. Non sono previsti né puntatori, né funzioni di allocazione e deallocazione della memoria. Questa caratteristica, se da una parte fa perdere flessibilità al linguaggio, rende Java particolarmente sicuro ed affidabile.

Java è un linguaggio interpretato, e la velocità di esecuzione delle applicazioni risulta essere nettamente inferiore a quella dei programmi scritti e compilati con il C++. Per rendere Java effettivamente competitivo rispetto al C++ ed ad altri linguaggi di programmazione, i ricercatori è stato sviluppato un compilatore che permetta di generare anche gli eseguibili di sorgenti Java.

Il compilatore Java che crea i programmi eseguibili da codici sorgente, compila per una macchina che non esiste: la *macchina virtuale Java*. Si potrebbe pensare alla macchina virtuale come a un microprocessore che sia implementato via software e funzioni utilizzando le capacità fornite dal sistema operativo e dall'hardware del computer. Poiché la macchina virtuale di Java non è un microprocessore reale, il bytecode di Java viene interpretato, anziché eseguito direttamente nelle istruzioni macchina native del computer host.

I problemi tradizionali degli interpreti sono sempre stati le loro scarse prestazioni: Java tenta di superarli compilando ad uno stadio

intermedio convertendo il codice sorgente nel bytecode, che può poi essere a sua volta convertito (in modo più efficiente) nel codice nativo di un particolare processore. Inoltre, creando un solo standard per il formato dei dati, Java garantisce che i programmi siano indipendenti dall'hardware.

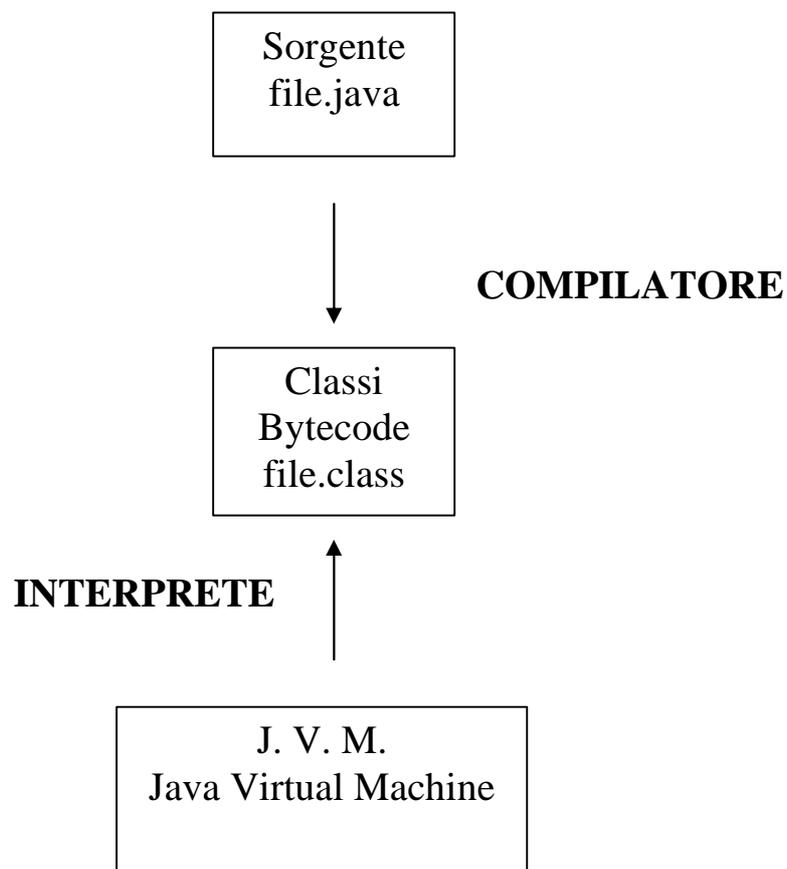


Figura 2. L'ambiente Java

La principale caratteristica di Java è che si tratta di un linguaggio realmente orientato agli oggetti. Questo fornisce diversi benefici, compreso: la riusabilità del codice, l'estensibilità, le applicazioni dinamiche.

La classe è una raccolta di variabili e metodi che incapsulano funzionalità in un oggetto riusabile e caricabile dinamicamente. Questo significa che una volta che la classe è stata creata, essa può venire utilizzata come modello per la creazione di altre classi che forniscono funzionalità diverse. Dopo la creazione di una classe, l'ambiente run-time di Java consente il caricamento dinamico delle classi. Questo significa che le applicazioni esistenti possono aggiungere funzionalità collegando nuove classi che incapsulano i metodi necessari seguendo un principio gerarchico di ereditarietà .

Un modo con cui Java supera i problemi di prestazioni dei tradizionali interpreti è l'inclusione dei multithreading.

E' possibile implementare più thread contemporaneamente, permettendo ad un singolo programma di eseguire più compiti separati nello stesso tempo.

Per un programma è insolito utilizzare costantemente i cicli della CPU. Il programma, invece, deve attendere l'input dell'utente e l'accesso ai file o alla rete. Queste azioni lasciano il processore inattivo nelle applicazioni single-threaded; Java utilizza questo tempo morto per eseguire la manutenzione generale del sistema.

Inoltre, i bytecode compilati di Java sono molto vicini ai codici macchina, quindi la loro interpretazione su una piattaforma specifica può essere molto efficiente.

5.6 Le librerie in Swarm

Swarm utilizza Java e la comunità degli sviluppatori e parte degli utilizzatori Swarm rendono pubbliche le classi già realizzate, in questo modo i programmatori di modelli in Swarm possono creare oggetti avvalendosi di queste informazioni che sono contenute in librerie.

Queste hanno due funzioni principali:

1. rappresentano una serie di classi che i costruttori di modelli possono utilizzare direttamente, creando oggetti derivati da esse;
2. è possibile impiegarle per creare delle sottoclassi con funzioni particolari, in base al tipo di modello che si vuole creare.

In Swarm si possono identificare specifiche librerie adatte all'uso simulativo che rappresentano un elemento fondamentale per lo sviluppo di modelli, perché ne abbreviano i tempi di realizzazione; esse sono:

- *activity*: questa libreria è il cuore del meccanismo simulativi, contiene il nucleo delle funzioni che gestiscono le strutture di dati necessarie a definire la sequenza degli eventi, le strutture dello schedule, il supporto per l'esecuzione e la gestione del tempo;

- *collections*: mette a disposizione le classi necessarie per la creazione di programmi Object Oriented di tipo generico, inoltre permette di rintracciare oggetti in un sistema complesso attraverso la gestione di liste, vettori e mappe;
- *defobj* : nella gerarchia delle librerie SWARM questa rappresenta la classe root, contiene le funzioni che permettono la definizione dei metodi di base per la creazione e l'eliminazione degli oggetti e le classi, che consentono di archiviare le istanze degli oggetti;
- *gui*: assegna le classi che permettono la creazione di oggetti grafici ed immagini e gestisce l'interazione tra utilizzatore ed interfaccia grafica;
- *objetbase*: libreria che contiene le classi fondamentali su cui si basano: gli oggetti presenti nelle simulazioni di Swarm ed il funzionamento degli agenti. Le classi presenti nella libreria sono: *SwarmObject*, che è la classe radice (*root*) da cui ereditano tutti gli agenti simulati e che definisce l'interfaccia base per la gestione della memoria, e *Swarm*, che fornisce le funzionalità richieste per la definizione di *modelSwarm* ed *observerSwarm*;
- *probe*: contiene le classi utili alla definizione degli oggetti sonda, utili per osservare e registrare variabili e parametri degli agenti della simulazione;

- *random*: offre all'utilizzatore un'ampia gamma di numeri casuali, essenziali per avere diversi tipi di generatori in uno stesso programma, in modo che si possa simulare i differenti comportamenti di tipo causale degli agenti.
- *simtools*: è un insieme di classi indispensabili per la costruzione del processo simulativo e per il controllo di tutto il modello. Queste classi possono generare insiemi di dati statistici, disegnare grafici ed altro.
- *tkobjc*: è un'interfaccia grafica basata sul toolkit Tcl/Tk, che a sua volta si appoggia su X-Windows. Con questa classe si possono creare istogrammi, grafici, schemi e bottoni.

Oltre ai pacchetti appena illustrati esistono numerosi software addizionali, creati dalla comunità degli utilizzatori di Swarm, creati per specifiche simulazioni: per esempio vi sono oggetti che permettono di gestire gli spazi in due dimensioni, algoritmi genetici e reti neurali.

Capitolo 6

Didattica e Simulazione, il modello Consumer

6.1 Premesse sul software

Il programma di simulazione Consumer, da me realizzato risponde ai vincoli del Software Aperto e Libero della GNU General Public License. (<http://www.gpl.org>)

6.1.1 Free software e Open Source

Il software libero è distribuito in modo che chiunque ne abbia il permesso di uso, di copia e di distribuzione, in forma modificata o meno, gratuitamente o a pagamento. Inoltre il codice sorgente deve essere disponibile.

L'espressione "Software libero" si riferisce alla libertà dell'utente di eseguire, copiare, distribuire, studiare, cambiare e migliorare il software. Più precisamente, esso si riferisce a tre tipi di libertà per gli utenti del software:

1. Libertà di eseguire il programma per qualsiasi scopo.
2. Libertà di studiare come funziona il programma, e adattarlo alle proprie necessità.
3. Libertà di migliorare il programma, e distribuirne pubblicamente i miglioramenti, in modo tale che tutta la comunità ne tragga beneficio.

Il modello proposto è stato scritto utilizzando il linguaggio di programmazione Java e si appoggia alle librerie che si trovano in Swarm. Per la realizzazione si è partiti dallo schema di un precedente lavoro denominato: “jSimpleBug” realizzato da Chris Lanton & Swarm development team Santa Fe Institute; sono stati utilizzati anche i successivi aggiornamenti apportati da Charles P. Staelin per convertire il codice dal linguaggio C++ al linguaggio Java.

6.2 Consumer

“Consumer” è un software di supporto alla didattica, che facilita la comprensione dell’economica.

La teoria economica classica modella il comportamento del consumatore attraverso strumenti matematico-statistici. Il software Consumer modella il medesimo comportamento utilizzando strumenti informatici in grado di simulare la realtà.

Una didattica che si appoggia ad entrambe le impostazioni otterrà risultati migliori.

Il programma simula il comportamento dei consumatori nel momento in cui decido se acquistare un bene. Essi valutano il prezzo cercano le imprese che possano soddisfare le loro richieste.

6.2.1 Avvio del programma

Il modello è composto da due soggetti fondamentali:

- i consumatori,
- le imprese.

Questi attraverso l'ambiente comunicano, interagiscono e commerciano.

Per avviare la simulazione, se si opera in ambiente Windows, occorre lanciare da Cygwin il comando `javaSwarm StartConsumer`.

6.2.2 I menù di scelta

All'avvio del software si visualizzano tre menù:

1. Il primo è comune a tutte le applicazioni Swarm e contiene i comandi che regolano l'avvio, e il blocco della simulazione, nel dettaglio:
 - Il bottone *Start* avvia la simulazione.

- Il comando *Stop* sospende il programma, che sarà possibile riavviare con *Start*.



Figura 1. ProcCtrl

- Il tasto *Next* permette di compiere un singolo passo con sentendo una visualizzazione *step by step*.
 - *Save* fissa la posizione delle finestre per le successive simulazioni. Il tasto *Quit* termina il programma e chiude tutte le finestre attive.
2. La seconda finestra, Observer Swarm, contiene informazioni sullo scorrere della simulazione a video e sulle caratteristiche delle finestre grafiche: grandezza del mondo, frequenza di aggiornamento dati, scorrere del tempo.



Figura 2. Observer Swarm

Qualora il personal computer su cui si lavora avesse difficoltà nel gestire la simulazione, riducendo la *display Frequency* e lo *zoom Factor* il carico di lavoro per il processore si ridurrebbe sensibilmente. La voce *simulate Time* visualizza lo scorrere del tempo della simulazione.

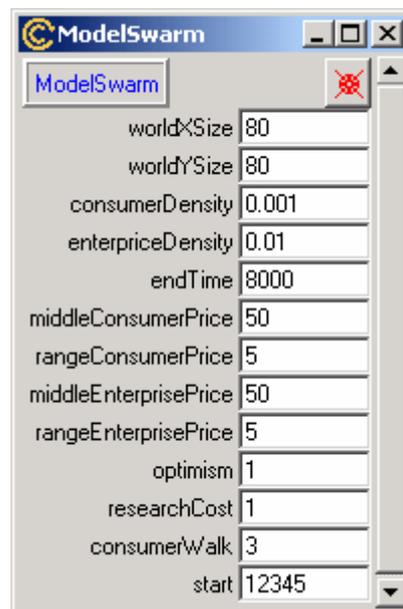


Figura 3. Model Swarm

3. Nella finestra Model Swarm sono definite le caratteristiche dei consumatori e delle imprese coinvolte nella simulazione. I singoli parametri possono essere modificati semplicemente sostituendo il valore visualizzato; il discente potrà verificare come, al variare delle singole caratteristiche di consumatori e imprese, reagisca il mercato.
- *WorldXSize e WorldYSize*: definiscono la grandezza del mondo in cui sono sparse le imprese e i consumatori.
 - *Consumer Density*: specifica il numero di consumatori presenti sul mercato. Il valore è strettamente legato alla superficie del mondo ed è definito come percentuale di mondo occupato da consumatori. Nell'impostazione standard il numero massimo di consumatori è pari a seimilaquattrocento unità, ampliabile estendendo la superficie del mondo.
 - *Enterprise Density*: specifica il numero di imprese presenti sul mercato ed è definita con le stesse regole della Consumer Density. Anche in questo caso nell'impostazione standard il valore massimo è pari a seimilaquattrocento unità.
 - *End Time*: definisce la durata del tempo e quindi della simulazione. Consumer associa lo scorrere di un'unità di tempo al passo dei consumatori. Ogni qual volta i consumatori avranno compiuto un passo sarà trascorsa una giornata.

- *Middle Consumer Price*: visualizza il prezzo medio a cui i consumatori sono disposti ad acquistare un bene. Ogni singolo consumatore avrà a disposizione un prezzo, calcolato come *Middle Consumer Price*, più o meno una quantità compresa tra zero e *Range Consumer Price*. Nel corso della simulazione il prezzo dei singoli consumatori cambierà in base a regole di comportamento definite nelle variabili *Optimism e Research Cost*. La prima interviene quando un consumatore incontra un'impresa disposta a vendere ad un prezzo minore rispetto a quello a cui il consumatore sarebbe disposto a comprare, la seconda quando lo stesso incontra imprese che vendono ad un prezzo più elevato.
- *Middle Enterprise Price*: definisce il prezzo medio delle imprese. Il prezzo di vendita di ogni singola impresa è definito come prezzo medio sommato o sottratto di una valore compreso tra zero e *Range Enterprise Price*.
- *Consumer Walk*: definisce il passo dei consumatori. Ad ogni ciclo si sposteranno di un passo pari a *Consumer Walk*.
- *Start*: modifica il generatore di numeri casuali. Per garantire la casualità nella simulazione occorre che il programma si avvii con valori nuovi ad ogni simulazione. Qualora non si modificano i parametri la simulazione ripercorrerà la medesima evoluzione fintanto che non avremo cambiato il valore dello *start*. Un cambiamento di tale valore assegna in modo

casuale nuovi prezzi e posizioni a tutti i consumatori e a tutte le imprese.

6.2.3 Avvio della simulazione

All'avvio della simulazione il programma genera i consumatori e le imprese in base ai valori impostati nel menù Model Swarm.

Le due categorie di agenti sono numerate e collocate in posizioni casuali all'interno del mondo. Le prime ad essere generate sono le imprese che comunicano il loro nome, corrispondente ad un numero, ed il loro indirizzo, rappresentato dalle coordinate x ed y che individuano la loro posizione nel mondo.

A conclusione del procedimento sul video è possibile leggere il numero delle imprese presenti sul mercato.

A seguire si formano i consumatori i quali, come le imprese, avvertono della loro presenza comunicando nome e posizione. Al termine di questa seconda procedura sul video è segnato il numero totale dei consumatori presenti sul mercato, e se alcuni di essi occupano una posizione già utilizzata da un'impresa.

```

~/Applicazioni/modello/Vtest/10/10
saverio@VEUDH ~/Applicazioni/modello/Utest/10/10
$ javaSwarm StartConsumer
L'impresa numero 1 è nella posizione 29, 14 con un prezzo di 53.0
L'impresa numero 2 è nella posizione 65, 16 con un prezzo di 48.0
L'impresa numero 3 è nella posizione 11, 18 con un prezzo di 55.0
L'impresa numero 4 è nella posizione 21, 28 con un prezzo di 50.0
L'impresa numero 5 è nella posizione 18, 63 con un prezzo di 55.0
5 Imprese sul mercato
Il consumatore numero 1 parte dalla posizione 7, 13 con prezzo 48.0
Il consumatore numero 2 parte dalla posizione 28, 16 con prezzo 54.0
Il consumatore numero 3 parte dalla posizione 9, 25 con prezzo 45.0
Il consumatore numero 4 parte dalla posizione 38, 26 con prezzo 45.0
Il consumatore numero 5 parte dalla posizione 70, 30 con prezzo 46.0
Il consumatore numero 6 parte dalla posizione 15, 48 con prezzo 51.0
Il consumatore numero 7 parte dalla posizione 34, 49 con prezzo 48.0
Il consumatore numero 8 parte dalla posizione 65, 55 con prezzo 51.0
Il consumatore numero 9 parte dalla posizione 3, 62 con prezzo 51.0
9 consumatori sul mercato, di cui su posizione di imprese 0
    
```

Figura 4. Avvio della simulazione

Contestualmente alle informazioni su posizione e nome degli agenti si avviano due finestre grafiche, la prima definita *Market Space* la quale rappresenta il mondo in cui operano i consumatori e le imprese.

All'avvio della simulazione il *Market Space* visualizza la posizione delle imprese con puntini di colore giallo ed i consumatori con puntini di colore blu.

Le imprese durante la simulazione non cambiano posizione e colore, i consumatori vagano casualmente nel mondo e a seconda del loro livello di soddisfazione modificano il colore diventando rossi se insoddisfatti delle offerte del mercato, verdi se soddisfatti, se il colore resta blu significa che i consumatori non hanno mai incontrato un'impresa .

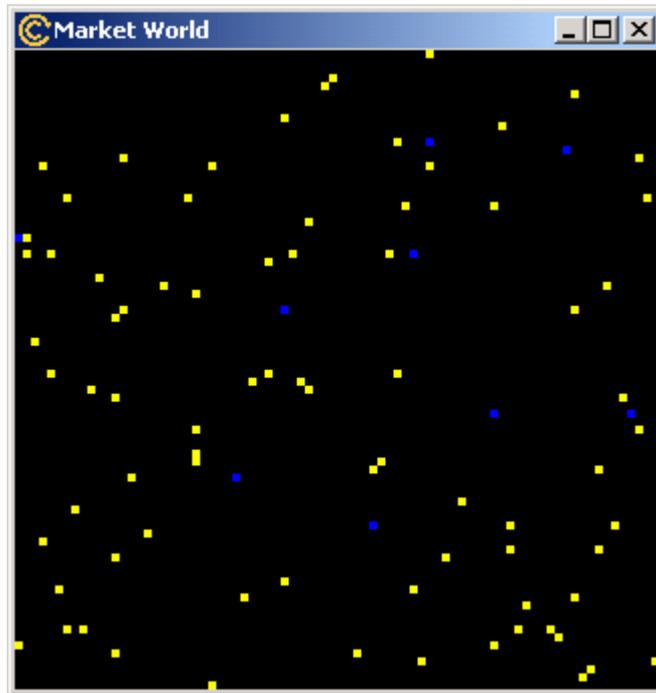


Figura 5. Avvio del Market Space

Contestualmente al Market Space compare il grafico Middle Consumer Price, il quale visualizza nello scorrere del tempo la media aritmetica del prezzo a cui sono disposti a comprare i consumatori. All'avvio e fintanto che nessun consumatore ha incontrato un'impresa il prezzo medio di riserva dei consumatori è un valore prossimo a quello impostato nella variabile *Middle Consumer Price*.

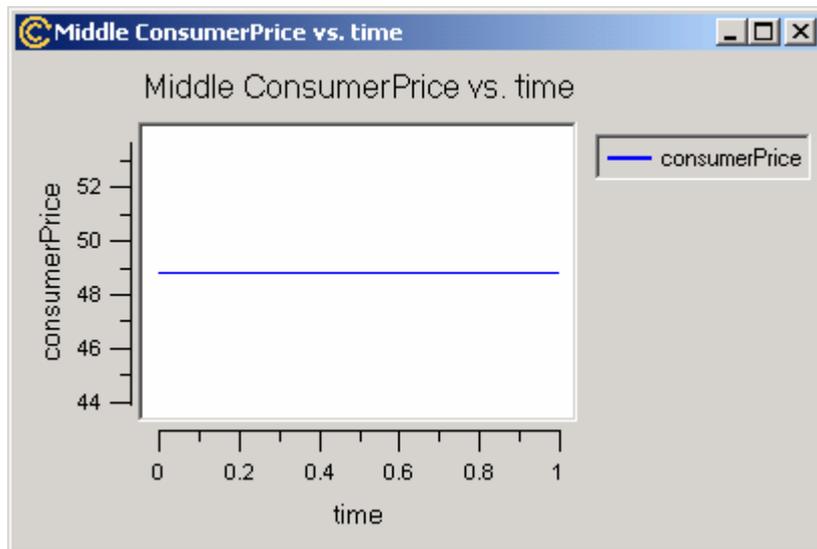


figura 6. Middle Consumer Price

6.2.4 L'ambiente

I consumatori e le imprese sono in un mondo bidimensionale, la loro posizione nello spazio è indicata dalle coordinate x ed y. Quando i consumatori giungono sui confini dello spazio e lo oltrepassano si ripresentano dalla parte opposta; il mondo è lo sviluppo bidimensionale di un toroide

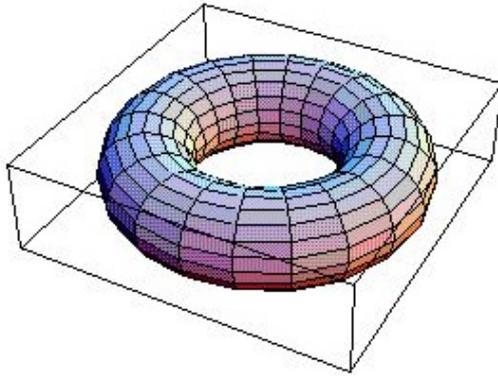


Figura 7a. Toroide

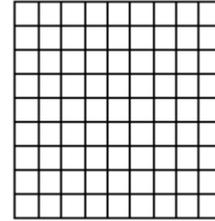


Figura 7b. Sviluppo bidimensionale di un toroide

6.2.5 Le imprese

Il comportamento delle imprese è inelastico alla domanda del mercato, quando si presenta un compratore forniscono il prezzo di vendita. Le imprese non modificano il loro comportamento sia che il consumatore accetti o rifiuti di acquistare.

I beni venduti si differenziano solo per il prezzo di vendita definito con *Enterprise Price* e *Middle Enterprise Price*.

6.2.6 I consumatori

All'avvio della simulazione i consumatori comunicano nome e posizione. Iniziano a camminare, con passo definito dall'utente con la variabile *Consumer Walk*. L'impostazione di default dispone tre cellette per passo. In partenza il colore è blu. Con il progredire del "random walk" ad ogni passo controllano se incontrano un'impresa. Al verificarsi dell'evento dialogano chiedendo il nome dell'impresa e il prezzo di vendita. Si potranno riscontrare due condizioni:

1. Il consumatore compra

Qualora il prezzo di vendita risulti inferiore o uguale al prezzo di riserva del consumatore l'acquisto avrà successo. Nello stato del singolo consumatore ciò comporterà due cambiamenti:

- il colore passerà da blu a verde,
- il prezzo a cui era disposto a concludere l'acquisto verrà aggiornato al prezzo di vendita dell'impresa, ulteriormente ridotto o incrementato di una componente definita a priori dall'utente con la variabile *optimism*

Si intende dimostrare che un consumatore quando conclude un acquisto adegua il proprio prezzo opportunità a quello dell'impresa,

incrementato o diminuito di una componente predefinita nella variabile *Optimism*.

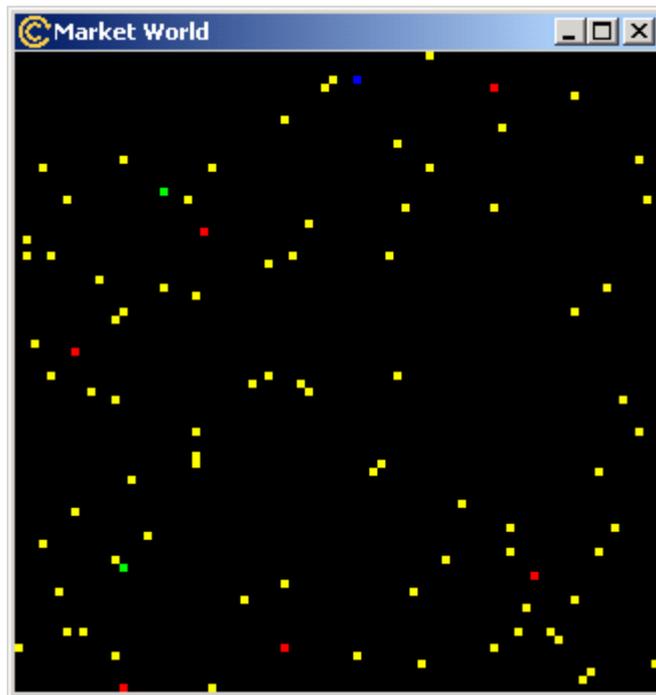


Figura 8. I tre stati del consumatore: blu, rosso e verde

I valori positivi sono associati a comportamenti ottimistici, il consumatore spera di spendere sempre meno, i negativi si riferiscono a comportamento pessimistico, il consumatore pensa di concludere i successivi acquisti a condizioni non così favorevoli.

I differenti valori impostati comportano dinamiche dei prezzi diverse. La componente ottimistica abbassa il prezzo medio, quella pessimistica lo alza.

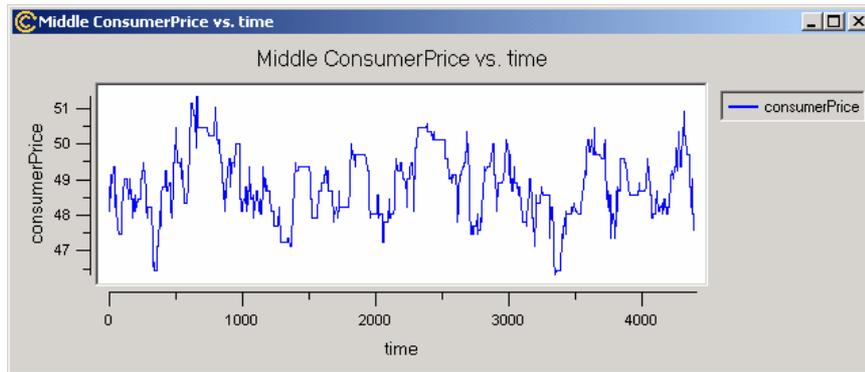


Figura 9. Dinamica dei prezzi

2. il consumatore non compra

Qualora il prezzo di vendita dell'impresa risulti maggiore di quello opportunità del consumare non si concluderà l'acquisto.

Anche questo caso comporterà due cambiamenti nello stato del consumatore.

- Cambierà colore diventando rosso
- aggiornerà il prezzo opportunità.

Il nuovo prezzo a cui sarà disposto a concludere l'acquisto sarà pari al prezzo di vendita dell'impresa ridotto di una quota definita dall'utente con la variabile *Research Cost*. Si rivelano due fenomeni:

- costi di ricerca

- l'adeguamento al prezzo di mercato.

6.3 Applicazioni

Il software è stato pensato e realizzato per consentire al discente di ripetere la simulazione più volte verificando come il mercato si modifichi al variare delle condizioni iniziali.

Ciò è garantito dal poter variare tutte le variabili direttamente dal video, senza alcun intervento nel codice del programma .

Gli eventuali scenari prefigurabili sono illimitati: spetterà al docente ed al discente esplorare le molte applicazioni.

Qui di seguito presentiamo alcuni esempi fondamentali:

6.3.1 Nessuno acquisto

Qualora si impongano valori irragionevoli il modello evidenzia tali errori. Se imponiamo un prezzo di vendita per le imprese alto, uno per i consumatori basso ed inseriamo una componente di ottimismo elevata e una componente riduttiva del costo di ricerca elevata, nel mercato non si andrà a commerciare. Dopo alcuni acquisti i consumatori diverranno di colore rosso, ed il prezzo medio visualizzato dal grafico *Middle Consumer Price* sarà pari ad una linea per la mancanza di vendite. Ciò poiché il basso costo di ricerca e il troppo ottimismo porta i consumatori a rifiutare gli acquisti spingendoli a continuare la ricerca.

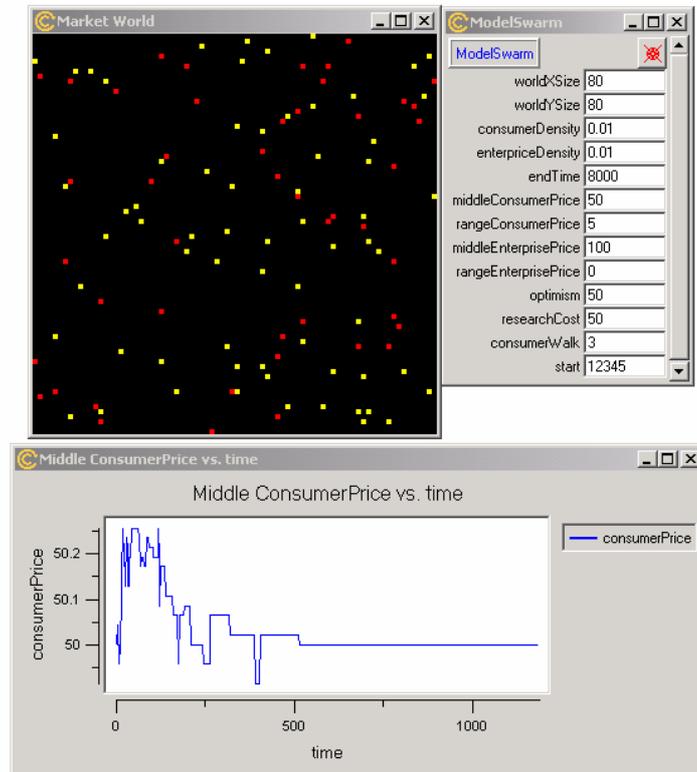


Figura 10. Nessun acquisto

6.3.2 Tutti acquistano

I consumatori con componente di ottimismo pari a zero ritengono di concludere i prossimi acquisti alle medesime condizioni. Se contestualmente hanno la componente attenuata dei costi ricerca pari a zero qualora non concludano l'acquisto aggiorneranno il prezzo opportunità a quello delle imprese.

I consumatori sono quindi molto elastici al mercato e sono pronti ad adeguarsi rapidamente ai prezzi delle imprese. Dopo un avvio in cui rifiuteranno l'acquisto cambieranno i loro comportamenti iniziando

ad acquistare sempre e comunque. La dinamica del prezzo avrà un andamento crescente che si appiattirà al prezzo di mercato delle imprese.

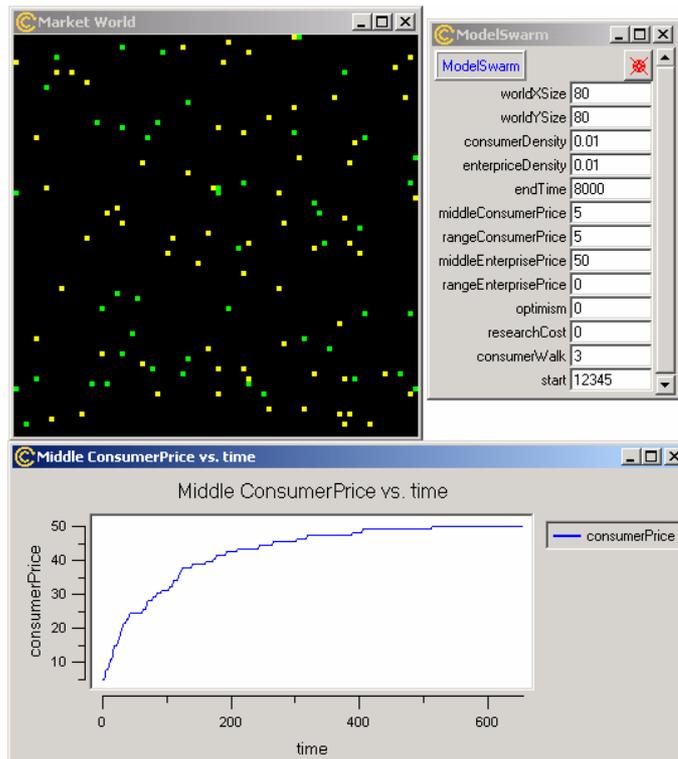


Figura 11. solo acquisti

6.3.3 Prezzi oscillanti

Qualora vi siano consumatori con prezzi potenziali di acquisto differenziati, costi di ricerca e componente ottimistica, la presenza di una impresa con alti o bassi prezzi nelle vicinanze definirà l'andamento del prezzo del singolo consumatore. La variabile *Consumer Walk* permetterà di affievolire la componente localizzativa

consentendo al consumatore grossi spostamenti. L'aumentare del range di prezzo delle imprese aumenterà la variabilità dei prezzi medi di acquisto.

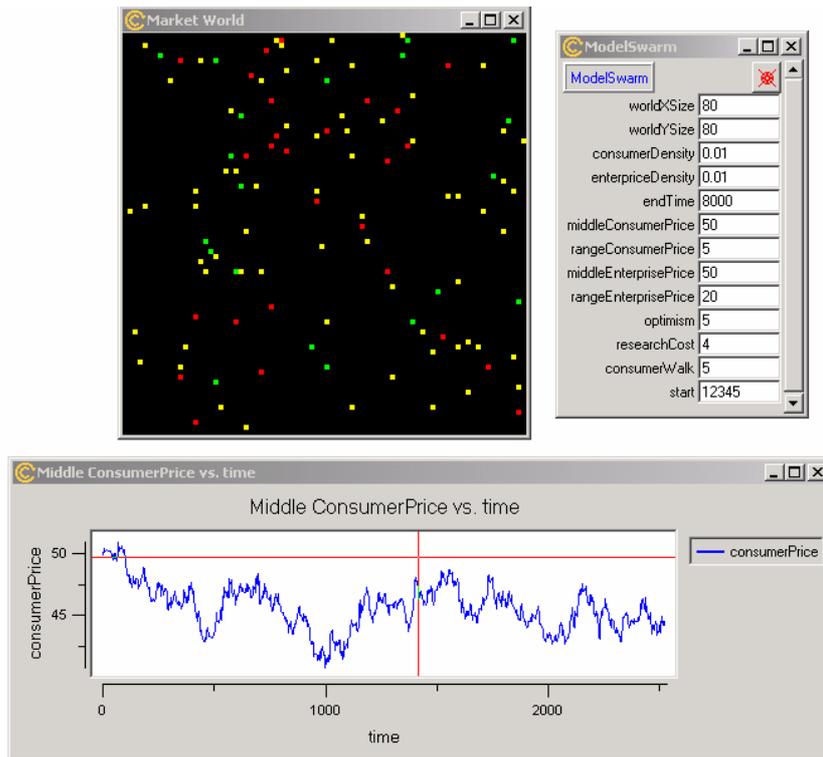


Figura 12. Variabilità del prezzo

6.4 Numerosità

Il programma Consumer permette anche la gestione di molti agenti. Potremo alzare il numero di consumatori e di imprese simulando i diversi tipi di mercato.

Per una migliore visualizzazione occorre modificare le impostazioni di superficie del mondo *World X/Y size* il tutto agendo sulle variabili della finestra model Swarm. Creare molti oggetti rallenta però i tempi di simulazione.

```

~/Applicazioni/modello/vtest/10/12
Il consumatore numero 2541 parte dalla posizione 16, 79 con prezzo 55,0
Il consumatore numero 2542 parte dalla posizione 18, 79 con prezzo 45,0
Il consumatore numero 2543 parte dalla posizione 21, 79 con prezzo 47,0
Il consumatore numero 2544 parte dalla posizione 27, 79 con prezzo 52,0
Il consumatore numero 2545 parte dalla posizione 28, 79 con prezzo 48,0
Il consumatore numero 2546 parte dalla posizione 30, 79 con prezzo 53,0
Il consumatore numero 2547 parte dalla posizione 34, 79 con prezzo 52,0
Il consumatore numero 2548 parte dalla posizione 40, 79 con prezzo 55,0
Il consumatore numero 2549 parte dalla posizione 42, 79 con prezzo 45,0
Il consumatore numero 2550 parte dalla posizione 45, 79 con prezzo 45,0
Il consumatore numero 2551 parte dalla posizione 46, 79 con prezzo 50,0
Il consumatore numero 2552 parte dalla posizione 47, 79 con prezzo 51,0
Il consumatore numero 2553 parte dalla posizione 54, 79 con prezzo 49,0
Il consumatore numero 2554 parte dalla posizione 55, 79 con prezzo 49,0
Il consumatore numero 2555 parte dalla posizione 58, 79 con prezzo 47,0
Il consumatore numero 2556 parte dalla posizione 61, 79 con prezzo 49,0
Il consumatore numero 2557 parte dalla posizione 62, 79 con prezzo 48,0
Il consumatore numero 2558 parte dalla posizione 68, 79 con prezzo 52,0
Il consumatore numero 2559 parte dalla posizione 70, 79 con prezzo 51,0
Il consumatore numero 2560 parte dalla posizione 73, 79 con prezzo 54,0
Il consumatore numero 2561 parte dalla posizione 76, 79 con prezzo 50,0
Il consumatore numero 2562 parte dalla posizione 77, 79 con prezzo 55,0
Il consumatore numero 2563 parte dalla posizione 78, 79 con prezzo 52,0
Il consumatore numero 2564 parte dalla posizione 79, 79 con prezzo 48,0
2564 consumatori sul mercato, di cui su posizione di imprese 33
    
```

Figura 13. 2564 consumatori presenti sul mercato

6.5 Conclusioni

Gli equilibri di mercato raggiungibili dal modello Consumer sono molteplici, sarà compito del docente e del discente esplorarli e scoprirli. In ciò si concretizza il terzo modello per la didattica: oltre al modello descrittivo, e a quello matematico-statistico, il modello di simulazione come strumento per capire per imparare e per sperimentare.

6.6 Evoluzioni del modello

Consumer è un esempio di come la simulazione sia strumento didattico. Future evoluzioni ed ampliamenti potranno accrescere le capacità esplicative del modello. Prossimi passi dovranno riguardare il settore delle imprese che per ora non sono dotate di contabilità, capacità di investire, conoscenza dei concorrenti ciò permetterebbe l'apertura e la chiusura di nuove imprese.

Occorrerà differenziare il comportamento dei consumatori. In questa prima formulazione hanno tutti medesimo passo, ottimismo, costo di ricerca. Sarebbe interessante dare una forma di memoria non solo di prezzo ma anche di localizzazione ai consumatori.

Tali evoluzioni permetteranno al modello di spiegare non solo il comportamento del consumatore ma anche numerosi altri eventi economici quali l'inflazione, i cicli economici, le diverse forme di mercato.

Nel complesso Consumer è un primo piccolo mattone per la costruzione di un nuovo impianto didattico basato sul modello sperimentale simulativo.

.

Capitolo 7

Formazione a distanza

7.1 Cenni storici

Per formazione a distanza si intende la situazione che si realizza quando tra docente e discente c'è una distanza di spazio e di tempo. Lo sviluppo della società dell'informazione ha fatto sì che questa modalità di formazione venisse negli ultimi anni ampliata e modificata.

L'insegnamento tradizionale, centrato sulla lezione frontale, ha perso l'esclusività delle azioni formative, a vantaggio di sistemi a base telematica per la condivisione delle informazioni.

Questo passaggio ha consentito lo spostamento dell'attenzione dal docente al discente, focalizzando i sistemi di più sull'apprendimento e di meno sull'insegnamento. La rivoluzione digitale e telematica ha portato una innovazione tale da non poter essere agevolmente assimilata dal sistema formativo attuale; molti sono stati i tentativi di unire le nuove tecnologie con la didattica, non sempre coronati dal successo sperato.

La realizzazione di sistemi per l'insegnamento e per l'apprendimento ha spesso privilegiato l'aspetto tecnico su quello

didattico; con questi presupposti, la continua evoluzione tecnologica ha fornito un terreno instabile sul quale costruire i sistemi didattici che nel tempo si sono susseguiti.

Alla luce di queste considerazioni si può dare uno sguardo ai metodi che sono stati utilizzati fino ad oggi ed ipotizzare una futura evoluzione dei sistemi didattici.

1. *Sistema postale*: si possono annoverare tra i primi sistemi di formazione a distanza quelli che facevano uso delle poste per il recapito dei materiali didattici: ad esempio la *Scuola Radio Elettra* di Torino e la *Open University* inglese; entrambe hanno utilizzato questo mezzo per collegare docenti, discenti e materiali didattici.

Si pensi anche ai sistemi di insegnamento adottati in Australia, paese in cui la dislocazione territoriale ha favorito l'instaurarsi di tali modelli didattici, soprattutto per la necessità di mettere in contatto soggetti distanti centinaia di chilometri l'uno dall'altro.

Il sistema non permette un'interattività elevata, ma può già configurarsi come "aperto" e "a distanza": infatti ogni discente è in grado di scegliere in quale momento e con quale modalità affrontare lo studio in un luogo diverso da quello dell'aula.

2. *Anni '80*: in Italia questo è il periodo dei corsi disponibili su nastro magnetico, un esempio, che sicuramente tutti cono-

scono, è rappresentato dai tanto pubblicizzati corsi di lingua straniera editi dalle grandi case editrici.

In seguito l'informatica personale ha consentito di racchiudere su supporti relativamente piccoli grandi quantità di dati: l'epoca del CBT, *Computer Based Training*, ebbe il massimo sviluppo parallelamente alla tecnologia del CD-ROM, consentendo di inserire i corsi su supporto ottico. Rispetto alla posta, il nuovo mezzo consentiva la riproduzione di materiali didattici più complessi del semplice testo; in questo periodo la cosiddetta "multimedialità" fu utilizzata come "panacea" per tutti i tipi di insegnamento e culminò con la realizzazione di numerosi corsi su CD-Rom sostitutivi della docenza "frontale". Si è verificato il fatto che, nonostante si fosse riscontrata una notevole diffusione dei supporti nelle scuole e nei sistemi formativi, in realtà questi non provocassero l'apprendimento desiderato.

Le motivazioni vanno ricercate nel metodo utilizzato; si trattava di sistemi per l'auto-apprendimento con scarsa o esigua comunicazione con il docente, in queste condizioni l'utente non motivato tendeva ad abbandonare il corso.

3. *Anni '90*: la diffusione delle connessioni internet verificatasi a partire dalla metà degli anni '90, ha permesso il superamento delle difficoltà che avevano incontrato i corsi a distanza degli anni '80, dovute soprattutto all'assenza di interattività con il docente; che può essere superata solo dalla possibilità di collegamento in tempo reale.

Sarebbe semplicistico considerare che è solo cambiato il mezzo di collegamento, passando dal cd-rom al sito web; questo corrisponde ad una prima fase dell'uso della telematica a fini didattici. Il vero cambiamento avviene quando il centro del sistema di apprendimento diventa la comunità di apprendimento, costituito dall'insieme *docente – discenti – materiali didattici* dinamicamente collegato tramite le reti telematiche, che permettono l'accesso sincrono e a-sincrono al modello di insegnamento.

La tecnologia diventa uno strumento da utilizzare ai fini didattici, ma non si arroga il diritto di sostituire la funzione del docente.

7.2 Prospettive future della didattica a distanza

La confluenza delle opportunità offerte dalla tecnologia e le istanze provenienti dal campo della didattica hanno portato alla nascita dell'area di studio denominata: *Technology mediated interactive learning* detta anche *Computer supported cooperative learning*. Essa scaturisce dalla sintesi di due filoni di ricerca:

- il filone noto come *Computer supported cooperative work*, sviluppatosi verso la metà degli anni '80, che considera le nuove tecnologie come un ausilio al lavoro di gruppo;
- il filone pedagogico afferente al *Cooperative learning*, che considera l'acquisizione di conoscenze, abilità e atteggiamenti da parte di un singolo come il risultato di

un'interazione di gruppo e quindi richiama l'attenzione sull'importanza delle relazioni interpersonali tra docente ed allievi, ma soprattutto tra allievi, nel processo di apprendimento.

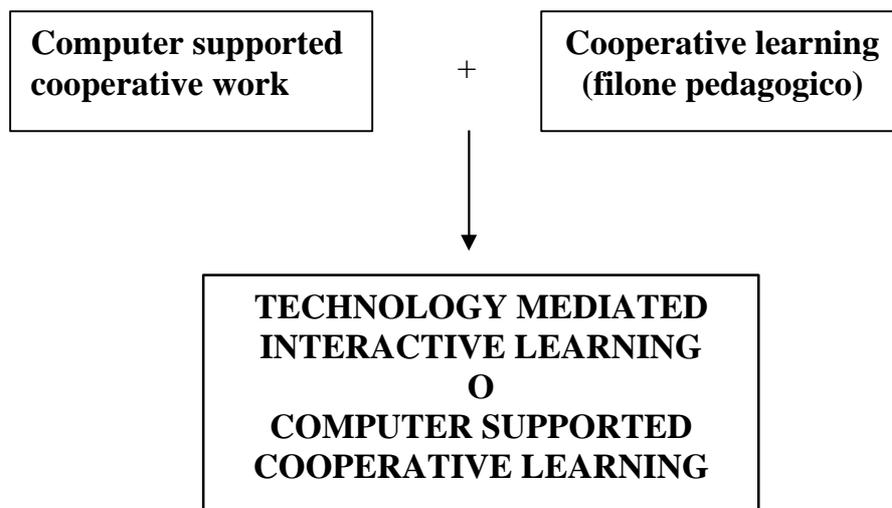


Figura 1. Schematizzazione della nascita dell'area di studio:
Technology mediated interactive learning

Quest'area di studio mette in evidenza il fatto che le nuove tecnologie offrono reti ed applicazioni tali da permettere una transizione dall'approccio individualistico alla metafora della rete elettronica intesa come "aula virtuale", in cui persone spazialmente e temporalmente distribuite, sono coinvolte in un processo di apprendimento da una parte e di insegnamento dall'altra attraverso l'ausilio di un sistema tecnologico. Tale sistema tecnologico ha il ruolo di:

1. mediare l'interazione persona-persona;
2. concorrere alla creazione di un "ambiente condiviso";
3. fornire gli strumenti per agevolare l'apprendimento cooperativo oltre a quello individuale.

La conseguenza immediata dell'impiego di questi sistemi è quella di consentire la progressiva diminuzione delle barriere spazio-temporali e la possibilità di realizzare modelli di apprendimento di tipo flessibile per diversi fattori:

- la sede di studio (in casa, al lavoro, in classe, ovunque sia conveniente),
- il tempo (ciascuno può apprendere in base ai propri ritmi),
- il percorso formativo (ciascuno lo sceglie secondo le proprie necessità e i propri interessi).

7.3 L'aula virtuale

A seconda delle varie esigenze di formazione, di istruzione e del tipo di utenza coinvolto è possibile realizzare differenti configurazioni di aula virtuale: da quelle più semplici, basate su sistemi di posta elettronica che consentono la comunicazione asincrona attra-

verso il testo, a quelle più complesse, dette a “banda larga”, di tipo più interattivo, basate sull’impiego dei sistemi multimediali, che permettono al docente e agli allievi di comunicare in tempo reale con l’ausilio di suoni, testi, immagini fisse ed in movimento.

Le esperienze significative finora realizzate sulle “aule virtuali” riguardano soprattutto interventi formativi attuati con mezzi di comunicazione semplici come: la posta elettronica e la *computer conference*.

Esistono pertanto scarse esperienze riguardo l’impiego di sistemi multimediali a “larga banda”, le cui potenzialità sono:

- maggiore comunicazione in tempo reale tra docente e discente (interattività);
- utilizzazione della comunicazione non verbale (attraverso gesti, tono della voce, postura, contatto visivo), l’ottenimento di un feedback visivo in tempo reale;
- l’elaborazione e la modifica di un documento comune (*cooperative editing*) e la trasmissione anche di comunicazioni informali tra docente ed allievi e tra allievi stessi;
- la creazione della cosiddetta telepresenza ossia di un senso di presenza condivisa tra persone geograficamente sparse con possibili effetti positivi sul clima didattico e sulle capacità di apprendimento degli allievi;

- l'utilizzazione da parte del docente della cosiddetta "cattedra multimediale", che permette la presentazione delle informazioni relative ai contenuti dell'intervento formativo sotto varie forme (testi, immagini fisse o in movimento, suono ecc) attraverso l'uso di diversi supporti: slides, videocassette, software didattici, collegamento a banche dati.

In sintesi, i sistemi multimediali a "larga banda" consentono di realizzare una crescente interattività tra docente ed allievi, l'uso della comunicazione non verbale e l'impiego dei supporti didattici multimediali.

L'interattività del mezzo tecnologico consente un immediato feedback tra i partecipanti e può incrementare la partecipazione attiva dei discenti alla lezione, non soltanto attraverso interazioni con il docente, ma anche con gli altri allievi. Questo tipo di sistemi permette di realizzare modelli di apprendimento cooperativo.

La grande varietà di servizi e mezzi di comunicazione messi a disposizione da tali sistemi tecnologici (come la posta elettronica, la computer conference, i sistemi groupware, la videoconferenza) facilita ricorso a soluzioni per l'apprendimento di tipo integrato in cui si alternano, nell'ambito di uno stesso intervento formativo, momenti di apprendimento "in presenza" e momenti di apprendimento "a distanza".

Da ciò emerge che l'aula virtuale non è pura riproduzione o duplicazione di una situazione "in presenza", ma è un ambiente al quale la tecnologia conferisce una maggiore flessibilità rispetto agli am-

bienti fisici, che si può riconfigurare ed adattare anche ai cambiamenti organizzativi.

7.4 Punti critici e accorgimenti

Nonostante questa ampia presentazione di opportunità, è largamente condiviso che le principali barriere alla diffusione delle “aule virtuali” di tipo interattivo sono legate ad aspetti quali le difficoltà di interazione dell’utente (docente e allievo) con il sistema multimediale e le resistenze dei docenti e degli allievi a cambiare i metodi tradizionali di insegnare, per gli uni, e di apprendere, per gli altri. Ad esempio le difficoltà di interazione con il sistema multimediale possono derivare da interruzioni nel collegamento, ovvero da un’interfaccia software troppo complessa per gli utenti, o dalla mancanza di un adeguato “training” all’uso delle apparecchiature, oppure dalla mancanza di un adeguato ausilio umano durante l’uso. Le resistenze dei docenti sono legate soprattutto al fatto che, le loro abitudini si rifanno all’utilizzo di metodologie didattiche tradizionali, come le lezioni di tipo “top down”, cioè basate su un flusso informativo diretto dal docente al discente. E’ raro l’utilizzo di modelli basati sull’apprendimento cooperativo.

D’altra parte i discenti solitamente svolgono le loro attività di apprendimento in modo individuale o in competizione con gli altri allievi, per questo si possono manifestare resistenze anche da parte dei discenti nell’adottare un approccio cooperativo.

L’uso delle tecnologie multimediali in “aula virtuale” implica da parte degli utenti (docente ed allievi) la possibilità di svolgere un

numero di compiti maggiore, avendo a disposizione un'ampia gamma di funzionalità. Questo sottolinea che il successo dell'aula virtuale dipende dalla misura in cui la tecnologia viene incontro a reali esigenze degli utenti e ribadisce la necessità della progettazione tecnologica centrata sull'utente.

L'uso di tecnologie multimediali obbliga il docente a cambiare le modalità abitualmente utilizzate per progettare e svolgere l'intervento formativo e richiede nuove competenze per poterlo realizzare in "aula virtuale". Ad esempio il docente dovrà tenere la lezione di fronte ad una telecamera, dovrà saper utilizzare i media a disposizione in modo appropriato, avrà il nuovo compito di gestire la comunicazione tra partecipanti dislocati in diverse postazioni remote.

Un altro punto critico è connesso al ruolo da assegnare alle nuove tecnologie. Le funzionalità offerte possono essere utilizzate sia come risorse nell'ambito di processi didattici tradizionali sia come strumenti per realizzare situazioni di apprendimento di tipo innovativo. E' oramai da più parti riconosciuto che finora, nelle esperienze di apprendimento a distanza, ci si è limitati ad utilizzare le tecnologie per riprodurre situazioni di apprendimento tradizionali. Non sono state sfruttate al massimo le potenzialità tecnologiche per introdurre innovazione nel processo di apprendimento.

Mettendo a confronto modelli di insegnamento tradizionali quali i "top down" e modelli all'avanguardia che si avvalgono di sistemi multimediali, si può dimostrare che questi ultimi sono in grado di garantire una migliore comprensione degli argomenti trattati, ciò è dovuto all'apporto di strumenti fatti per svolgere attività in paralle-

lo, di sperimentazione, di simulazione per immagazzinare, organizzare e recuperare informazioni.

Il successo dell'apprendimento in rete e la creazione delle "aule virtuali" dipende soprattutto dal supporto che deve essere dato all'utenza, ossia occorre garantire un uso corretto dello strumento di insegnamento per i docenti e di apprendimento per gli studenti.

7.5 Il ruolo dell' "human factor"

Nell'affrontare i temi della didattica a distanza un contributo particolarmente utile è fornito dall'analisi dell' "human factor" . Esso si colloca in tutte le fasi di sviluppo delle nuove tecnologie didattiche. Il concetto chiave della disciplina dei fattori umani o "human factor" è l'usabilità. L'usabilità è un concetto generale e complesso che, in estrema sintesi, fa riferimento alla facilità di uso degli apparati, in termini di efficacia e di soddisfazione nell'uso da parte degli utenti.

Il concetto di usabilità applicato alle situazioni di "aula virtuale" basate su collegamenti a "larga banda" permette di affrontare alcuni aspetti specifici:

- l'interazione del docente con la "cattedra multimediale",
- i requisiti del sistema multimediale,
- le procedure di "training" di docenti ed allievi,

- i protocolli di comunicazione da adottare in “aula virtuale”.

Il sistema deve consentire la creazione di uno spazio con la possibilità di condividere documenti, oggetti ed applicazioni software.

La formazione, o “training”, dei docenti è fondamentale per creare le nuove competenze richieste al fine di condurre le lezioni in “aula virtuale”.

Deve essere finalizzata al massimo sfruttamento delle opportunità offerte dalla tecnologia, soprattutto per realizzare modelli di apprendimento cooperativo, in cui il docente ha il ruolo fondamentale di promuovere e gestire la comunicazione tra allievi. Occorre considerare i problemi riguardanti la produzione dei materiali didattici multimediali ed il riadattamento di quelli già esistenti.

Problematiche di tipo analogo esistono riguardo alle procedure di addestramento dei discenti. Esse devono essere orientate all’acquisizione, da parte degli allievi, degli strumenti che compongono il sistema, affinché una volta apprese le modalità di funzionamento, riescano a servirsene in modo efficace per l’apprendimento didattico. Il “training” deve essere orientato, non solo, a fornire agli allievi le istruzioni necessarie per interagire correttamente con le apparecchiature, ma anche a fornire indicazioni sul comportamento da tenere nell’ “aula virtuale”.

Per quanto riguarda i protocolli di comunicazione e di comportamento per gli allievi, a seconda della metodologia didattica adottata, è necessario definire con quali modalità è opportuno gestire gli interventi, le richieste di chiarimenti e le domande durante l’intervento formativo.

In “aula virtuale” è necessario definire le procedure da adottare nell’eventualità di guasti alle apparecchiature che richiedano, ad esempio, l’intervento di un tecnico o il ricorso a mezzi di comunicazione alternativi quali: l’audioconferenza o la computer conferenza. Da quanto detto risulta evidente che la disciplina dell’“human factor” permette di studiare ogni aspetto del complesso fenomeno dell’insegnamento e dell’apprendimento a distanza rendendone possibile un’analisi scientifica interdisciplinare ricca di suggerimenti operativi in campo tecnologico, psicologico e didattico.

Capitolo 8

Esempi di software web-based

8.1 Introduzione

Prima di analizzare i singoli software idonei a sviluppare le nuove tecniche di insegnamento per il docente e di apprendimento per lo studente attraverso l'internet, occorre distinguere due categorie di strumenti che adottano “modus operandi” differenti.

La prima si fonda su un'idea più generale e prevede una totale applicazione del modello virtuale. Attraverso particolari software si creano piattaforme di comunicazione che interessano la totalità dei rapporti interpersonali che intercorrono tra docente e studente e tra studenti stessi.

È riprodotta un'aula virtuale che sopperisce completamente al metodo d'insegnamento tradizionale; tutto si può svolgere quindi attraverso il Web: passaggi di documenti, colloqui, visione di slides, proposte di quesiti.

La seconda categoria invece corrisponde ad una visione mista, che non abbandona completamente il metodo didattico tradizionale. Il docente è aiutato nella sua tecnica di insegnamento da programmi che facilitano il passaggio d'informazioni.

8.2 Future Learning Environment 3

Software Open Source web-based per la crescita della conoscenza, sviluppato dall'UIAH media lab, (<http://FLE3.org>) in cooperazione con il Center for Research on Networked Learning and Knowledge Building, Department of Psychology, University of Helsinki.

FLE3 (Future Learning Environment) è un software server per supportare l'apprendimento collaborativo. Il programma si appoggia ad un server Zope.

L'ambiente Zope, software anche esso Open Source, mette a disposizione una architettura ad oggetti per la costruzione di portali Web modulari e dinamici. Scritto per lo più in Python, incorpora un server Web ed un database object oriented.

Il prodotto si amministra e si utilizza da un browser Web, attraverso una interfaccia intuitiva che aiuta a gestire l'elevata complessità del prodotto.

FLE3, appoggiandosi alla gestione dinamica delle pagine garantite dall'ambiente Zope, coordina la crescita della conoscenza favorendo l'interazione tra docente e studenti.

Ogni utente di FLE3, a seguito del log-in nel sistema, ha un desktop via web personale e personalizzabile, attraverso cui dialoga con gli altri utenti inserendo documenti, collegamenti ipertestuali, appunti, commenti, messaggi. Il tutto in relazione o meno con quanto inserito dai docenti e dagli altri partecipanti.

Il doppio flusso di informazioni modifica il ruolo del docente che diventa non solo più quello di inserire automaticamente materiale

didattico, ma anche di gestire e stimolare la nascita di discussioni all'interno del gruppo.

Il flusso informativo in FLE3 si articola in modo radicalmente diverso rispetto ai forum via web.

Nei forum la conoscenza tende a distribuirsi in modo verticale ovvero lo studente pone dei quesiti, a cui il docente risponde, seguono eventualmente precisazioni e richieste di ulteriori chiarimenti. Contestualmente, con un medesimo rapporto domanda risposta il forum evolve e nasconde i set di domanda/risposta da più tempo formulati. In FLE3 invece la conoscenza si diffonde in modo orizzontale creando una specie di ragnatela del sapere in cui i singoli contributi generano un risultato a priori non prevedibile e previsto.

L'interfaccia del programma può essere impostata in diverse lingue tra cui quella italiana. In FLE3 si possono avere tre diversi livelli di utenza:

1. l'amministratore,
2. lo staff,
3. l'utente.

Ai tre livelli di utenza sono associati privilegi diversi: l'amministratore genera i corsi e gestisce i partecipanti, lo staff gestisce i corsi e stimola le discussioni, l'utente usufruisce del corso e sotto la guida del docente costruisce la conoscenza.

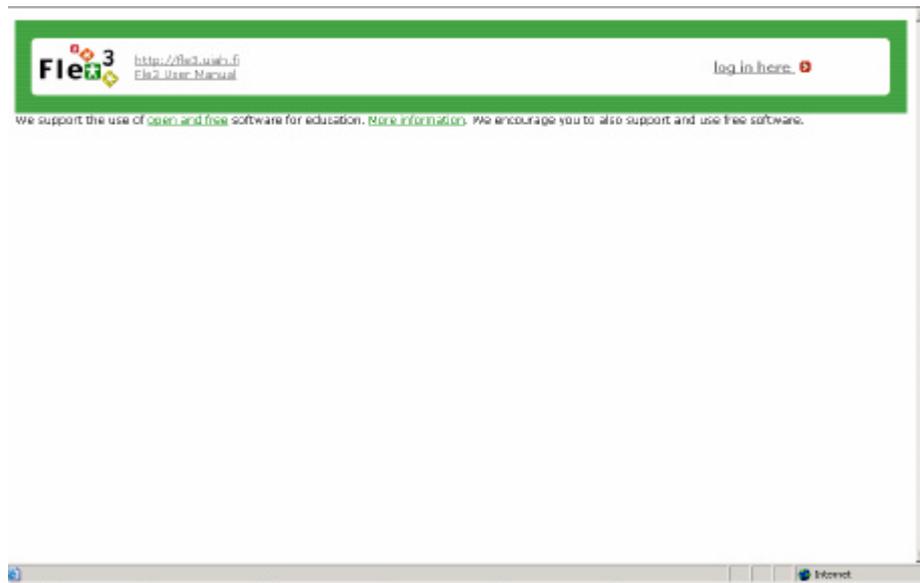
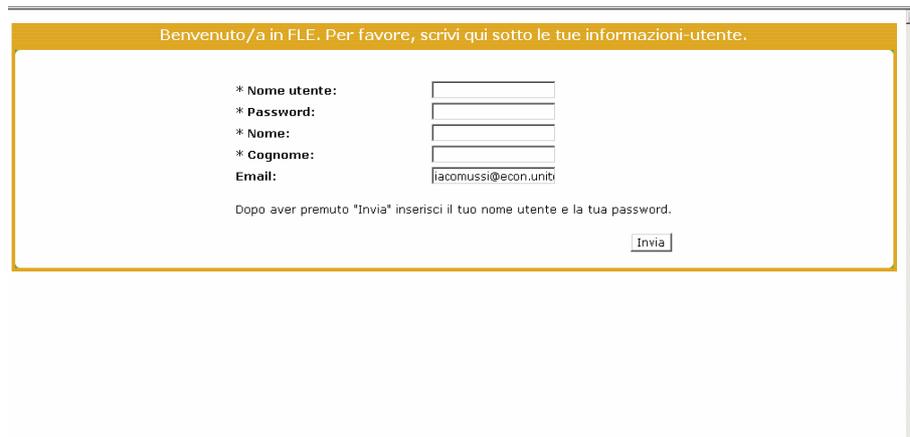


Figura 1. Log-in di FLE3

Ruolo principale dell'amministratore è quello di:

1. creare i corsi,
2. iscrivere gli utenti, tale procedura avviene direttamente o con invio via mail,
3. impostare il livello di accesso: amministratore, staff, o utenti,
4. garantire "ordine" bloccando gli utenti indisciplinati.



Benvenuto/a in FLE. Per favore, scrivi qui sotto le tue informazioni-utente.

* Nome utente:

* Password:

* Nome:

* Cognome:

Email:

Dopo aver premuto "Invia" inserisci il tuo nome utente e la tua password.

Figura 2. Iscrizione in FLE3

A guidare il propagarsi della conoscenza è lo staff che stimola e modera la discussione.

L'utente a seguito del log-in visualizza la scrivania ed i tools di sviluppo. In basso in posizione centrata l'elenco dei corsi a cui è iscritto.

L'utente potrà navigare all'interno dei documenti dei corsi a cui partecipa, ricevendo informazioni continue su eventuali modifiche intercorse dal precedente accesso.

L'amministratore e lo staff di FLE3 avranno invece un menù con molte più scelte tra cui quella per la gestione utenti e per la gestione dei corsi:



Figura 3. Scrivania dell'amministratore di FLE3

8.2 FLE3 Tool

Desktop

Primo tool che compare dopo il log-in è la scrivania personale, la quale (fig. 4) contiene i collegamenti ai corsi e al materiale didattico.

Ogni utente entrando nella cartella "corso" visualizza il materiale didattico, i commenti, ed i link inseriti dallo staff e dagli altri studenti.

Potrà inoltre accedere alla scrivania di ognuno dei partecipanti visualizzando , ma non modificando i documenti inseriti.



Figura 4. Scrivania dell'utente

Knowledge Building

La sezione costruzione della conoscenza è un ambiente strutturato per discussioni di gruppo; sulla sinistra (fig. 5) gli utenti visualizzano chi è connesso in quel momento, sulla destra chi non lo è.

I contesti, o temi della discussione, sono creati dagli amministratori e dai membri dello staff.

I singoli utenti potranno navigare, rispondere, intervenire e creare nuovi temi o contesti. Ai testi è possibile allegare immagini o documenti.



Figura 5. Knowledge Building

Jamming

L'area improvvisazioni: permette ai singoli utenti di inserire dei documenti concatenati l'uno all'altro creando una complessa rete di conoscenza. Si può sviluppare come connettore di diversi contesti, e può collegare documenti di natura diversa: un commento, un collegamento ipertestuale, un'immagine.

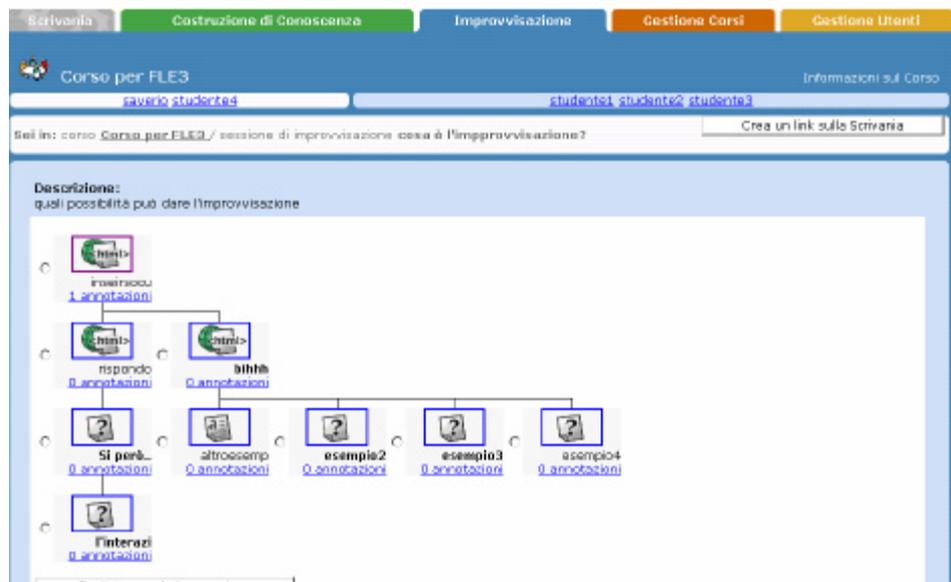


Figura 6. Jamming

8.3 Real Presenter (basic)

Real Presenter basic è un programma gratuito prodotto dalla Real (<http://www.real.com>) che permette di realizzare presentazioni commentate partendo da file generati dal programma Power Point. La facilità nella costruzione della presentazione, la semplicità di utilizzo da parte dell'uditore lo rendono un programma interessante per la didattica.

Il software non è sostitutivo della didattica frontale, ma può divenire un valido supporto ad esempio in sostituzione delle dispense distribuite dal docente durante le lezioni.



Figura 7. Real Presenter Basic

Le presentazioni realizzate con Real Presenter potranno essere archiviate o su un cd-rom, o direttamente su di un server web, nel qual caso occorre un software-server aggiuntivo, RealServer, per la visione in remoto.

Real Presenter è un programma semplice ed intuitivo, tanto nella fase di realizzazione del lavoro (tramite un plug-in che si attiva direttamente da Power Point), che nella fase di consultazione per la completa integrazione con RealPlayer.

La realizzazione della presentazione si articola in semplici passaggi:

1. da Power Point si avvia Real Presenter,
2. è richiesto il controllo video/audio. Real Presenter permette oltre che la riproduzione delle slide con il contestuale commento, anche la registrazione video del docente, attraverso una webcam,
3. terminato il controllo audio le slide vengono convertite nel formato jpg e si procede alla registrazione,
4. con il tasto invio si avanza di slide, con start si attiva la registrazione con pausa e con stop per terminare,
5. si procede quindi con la lettura dei contenuti da associare alle singole slide; qualora si commettano errori si potrà correggere la registrazione effettuata in ogni singola slide in un secondo momento.
6. ottenuta la qualità desiderata si può salvare la registrazione.



Figura 8. Menù di Real Presenter

8.4.1 Modifiche alle presentazioni

Qualora sia necessario modificare presentazioni realizzate in precedenza Real Presenter dispone di un tool per la gestione delle presentazioni.

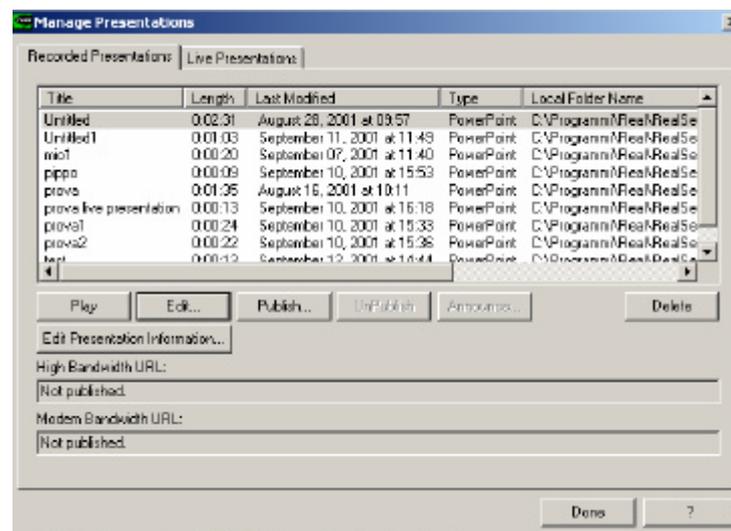


Figura 9. Manage your presentations

Dal manage your presentations è possibile accedere a tutte le presentazioni create con Real Presenter, di ognuna si ha una serie di dati informativi: durata, titolo, ultima modifica, cartella in cui è salvata. Da qui è poi possibile cancellare, riascoltare o modificare le singole presentazioni.

Il programma non gestisce la versione di Power Point 2002 presente in Office XP.

8.5 Microsoft Producer

Software free scaricabile dal sito Microsoft (<http://microsoft.com>). Come Real Presenter è utilizzato per la realizzazione di presentazioni commentate a partire dal file Power Point. Per chi intende realizzare la presentazione i requisiti minimi di sistema sono:

- Windows 2000 (con service Pack 1) o Window XP;
- Office XP o Microsoft Power Point 2002.

Per chi vuole visualizzare le presentazioni :

- Explorer 5;
- Windows Media Player.

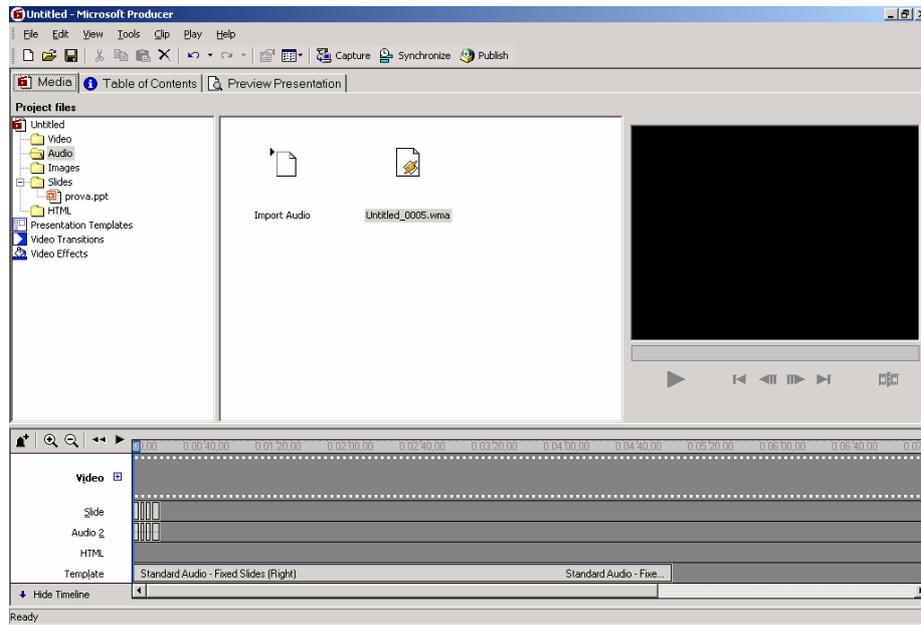


Figura 10. Microsoft Producer

La creazione delle presentazioni è semplice come con Real Presenter. Si può registrare direttamente man mano che scorrono le slide o montare audio e video in un secondo momento.

Nella fase di visione si utilizza Microsoft Explorer che gestisce la sequenza delle slide, mentre Windows Media Player gestisce il sonoro.

I video commentati potranno essere archiviati o su cd-rom o direttamente su di un server web senza alcun software di gestione.

Conclusioni

In questo lavoro abbiamo trattato di come l'acquisizione di sapere sia stata coinvolta da profondi cambiamenti sociali che hanno portato alla diffusione di un nuovo orientamento metodologico nella didattica.

La trasformazione in atto è dovuta essenzialmente all'introduzione di processi informatici nel metodo di insegnamento.

Tali processi permettono la realizzazione di strumenti attraverso cui è possibile creare rinnovati procedimenti di apprendimento.

L'analisi dei nuovi modelli didattici porta alla conclusione che lo strumento della simulazione per lo studio della realtà economica, così come la formazione di una "classe virtuale" devono essere considerate un valore aggiunto ai metodi tradizionali di insegnamento.

Da un lato ai modelli letterario-descrittivi e matematico-statistici si affiancano i modelli simulativi. La simulazione attraverso software specialistici, come Swarm, permette una visione della realtà economica nuova e diversa rispetto a quella offerta dagli altri due modelli didattici. Questa nuova impostazione dovrebbe garantire allo studente una migliore comprensione delle scienze complesse ed astratte.

Occorrerà però quantificare quale sia il contributo del modello didattico simulativo, un interessante sviluppo al presente lavoro sarà quello di sperimentare in aula i modelli come Consumer e verificare attraverso questionari specifici il grado di comprensione dei concetti trattati.

In seconda istanza la rete, per le potenzialità che offre, può essere considerata un ambiente di apprendimento in cui mettere in pratica varie modalità di acquisizione di nozioni, lo studente può:

1. ricercare, selezionare ed elaborare le varie informazioni desiderate,
2. interagire con altre persone,
3. apprendere insieme ad altri,
4. cooperare ad un progetto on-line.

Nel processo di apprendimento on-line c'è una maggiore interazione tra gli studenti. Essa si traduce in un arricchimento del processo formativo.

Tuttavia non si deve pensare che il virtuale sia qualcosa che surroghe il reale, si tratta, invece, di un'amplificazione delle possibilità di interpretazione e di uso del reale. Per virtuale dobbiamo intendere un ampliamento degli spazi, delle categorie, delle forme entro le quali ridefinire, contrattare una nuova idea di realtà formativa.

Verificare le teorie attraverso delle simulazioni, ovvero studiare via Web, significa entrare in una logica di apprendimento nuova che crea inevitabilmente un arricchimento culturale. Per mettere in atto questi metodi di studio è indispensabile imparare ad interagire con la macchina mettendo in gioco abilità, conoscenze, ed esperienze. Tutto ciò costituisce una sfida epistemologica che riguarda l'utenza, ma anche chi trasmette e produce cultura.

Il docente deve essere comunque insostituibile come regista multimediale e multi-culturale del processo educativo a cui sono e saranno sottoposti gli studenti delle future generazioni.

Riferimenti bibliografici

Testi:

ABBOLITO A., PAPA F. E SPEDALETTI S. (1995), *Human factors in tele-education: an application in the medical field*, Journal of medical education technologies, Vol 5, n. 4.

ARTHUR W.B., DURLAUF S.N., LANE D.A (1997), *The Economy as an Evolving Complex System II*, Reading: Addison-Wesley.

ASKENAZI M., BURKHART R., LANGTON C., MINAR N. (1996), *The Swarm Simulation System: A Toolkit For Building Multi-Agent Simulations*, Santa Fe Working Paper.

(<http://nelson.www.media.mit.edu/people/nelson/research/swarm>)

BUXTON B. (1991), *Telepresence: integrating shared tasks and personal spaces*, Proc. of the Conference groupware 1991, Amsterdam.

CILLERS P. (1998), *Complexity and Postmodernism: understanding complex systems*, Routledge.

CONTE R. (1997), *Il metodo simulativo*, Istituto di Psicologia, Working Paper, Cnr Roma.

DEDE C. (1990), *The evolution of distance learning: technology - mediated interactive learning*, Journal of research on computing in education, Vol 22, n. 3.

GILBERT N., TERNA P. (1999), *How to build and use agent-based models in social science*, Mind & Society.

HILTZ S.R. (1994), *The virtual classroom. Learning without limits via computer networks*, , Ablex publishing, Norwood.

HOLLAND H.J. (1995), *Hidden Order: how adaptation build complexity*, Addison-Wesley.

KAYE A. (1994), *Apprendimento collaborativo basato sul computer*, Td. Tecnologie didattiche, n. 4.

KAUFFMAN (1995), *At Home in the Universe: The Search for the Laws of self Organization and Complexity*, Oxford University Press.

LORENZ E. (1995), *The Essence of Chaos*, UCL Press.

MICHAEL L. KATZ E HARVEY S. ROSEN (1996), *Microeconomia*, McGraw-Hill.

OLSON M. (1987), *Computer supported cooperative work*, Office: technology and people, Vol. 3, n 2.

ORLANDO A. (1992), a cura di *Human factors in information services*, Final report of the Cost 212 project published by the Commission of the European communities, Bruxelles-Luxembourg.

PAPA F., PERUGINI M. E SPEDALETTI S. (in corso di pubblicazione), *Psychological factors in virtual classroom situations: a pilot study for a new model of learning through technological devices*, Behaviour & information technology.

PARISI D. (2001), *Simulazioni - La realtà rifatta nel computer*. Mulino, Bologna.

PERLO JEFFREY M. (2001), *Microeconomics*, Addison Wesley Longman.

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.

RIEL M. (1994), *Educational change in technology rich environment*, Journal of research on computing in education, n. 4.

SHACKEL B.(1986), *Ergonomics in designing for usability*, in Harrison M. D. e Mark A.F. (a cura di), *People and computers designing for usability*, Proc. of Hci 86, York.

TERNA P. (1999), *Economic Experiments with Swarm: a Neural Network Approach to the Self-Development of Consistency in Agents' Behaviour*, in LUNA F., STEFANSSON B., *Economic*

Simulations in Swarm: Agent Based Modelling and Object Oriented Programming, Kluwer Academic.

TERNA P. (2002), *La simulazione come strumento di indagine per l'economia*. Presentato a “Scienze Cognitive e Economia”, Workshop organizzato dalla Associazione Italiana di Scienze Cognitive, Rovereto.

VARIAN HAL R. (1998), *Microeconomia*, Cafoscarina.

WALDROP M. MITCHELL (1992), *Complexity: the emergence of science at the edge of order and chaos*, Simon and Schuster.

WOOD R. (2000), *Managing Complexity*, The Economist Books.

Link:

[1] CNR Genova , Istituto per le tecnologie didattiche:

<http://www.itd.ge.cnr.it>

[2] Commissione Europea sito web :

<http://europa.eu.int/comm/education/elearning>

[3] Europea Journal of open and distance learnig:

<http://www1.nks.no/eurodl/eurodlen/index.html>

[4] FESR - Fondo europeo di sviluppo regionale:

<http://www.inforegio.cec.eu.int>

[5] Ministero della pubblica istruzione:

<http://www.istruzione.it>

[6] Progetto Erasmus, UE:

<http://europa.eu.int/comm/education/socrates/erasmus/home.html>

[7] Progetto Socrates, UE:

<http://europa.eu.int/comm/education/socrates-it.html>

[8] Progetto SWARM:

<http://www.swarm.org>

[9] SCIENTER, centro di ricerche e servizi avanzati per la formazione: <http://www.scienter.org>

[10] Servizio documentazione software didattico:
<http://sd2.itd.ge.cnr.it>

Appendice I

Il codice Java-Swarm del modello Consumer

modelswarm.java

```
// ModelSwarm.java
import swarm.Globals;
import swarm.Selector;
import swarm.defobj.Zone;

import swarm.objectbase.Swarm;
import swarm.objectbase.SwarmImpl;
import swarm.objectbase.EmptyProbeMap;
import swarm.objectbase.EmptyProbeMapImpl;

import swarm.activity.ActionGroupImpl;
import swarm.activity.ScheduleImpl;
import swarm.activity.Activity;

import swarm.space.Grid2dImpl;
import swarm.collections.ListImpl;

public class ModelSwarm extends SwarmImpl
{

    // Dichiaro i paramentri del modello e i loro valori standard

    public int worldXSize = 80, worldYSize = 80;
    public double consumerDensity = 0.10;
    public double enterpriceDensity = 0.10;
    public int endTime = 125;
    public int middleConsumerPrice = 50;
    public int middleEnterprisePrice = 50;
```

```
public int start = 12345;

public int optimism =1;
public int consumerWalk = 3;
public int rangeConsumerPrice = 5;
public int rangeEnterprisePrice = 5;
public int researchCost = 1;

// dichiaro le variabili non standard

public MarketSpace marketSpace;
public Grid2dImpl consumerSpace;
public ListImpl consumerList;
static public Grid2dImpl enterpriceSpace;

//occorre renderla static perchè vado a passarla al consumatore
//per la prima verifica di posizione, altrimenti va in errore le
//imprese una volta create non si muovo quindi l'attributo static non
//disturba

public ListImpl enterpriceList;
public ScheduleImpl modelSchedule;

// Questo è il costruttore per il nuovo model swarm

public ModelSwarm(Zone azone)
{

    super(azone);

// creiamo i probe I quali sono I menu del modello

    EmptyProbeMapImpl probeMap =
        new EmptyProbeMapImpl(azone, getClass());

//inseriamo le variabili che si vuole avere all'interno del
//menù generato dall'observer.swarm

    probeMap.addProbe(getProbeForVariable("worldXSize"));
    probeMap.addProbe(getProbeForVariable("worldYSize"));
    probeMap.addProbe(getProbeForVariable
```

```
("consumerDensity"));
    probeMap.addProbe(getProbeForVariable
("enterpriceDensity"));

    probeMap.addProbe(getProbeForVariable
("endTime"));

    probeMap.addProbe(getProbeForVariable
("middleConsumerPrice"));
    probeMap.addProbe(getProbeForVariable
("rangeConsumerPrice"));
    probeMap.addProbe(getProbeForVariable
("middleEnterprisePrice"));
    probeMap.addProbe(getProbeForVariable
("rangeEnterprisePrice"));

    probeMap.addProbe(getProbeForVariable
("optimism"));
    probeMap.addProbe(getProbeForVariable
("researchCost"));
    probeMap.addProbe(getProbeForVariable
("consumerWalk"));

    probeMap.addProbe(getProbeForVariable
("start"));

    // per ultimo installiamo i probe nel probeLibrary.

    // Note that this library was created by initSwarm().

    Globals.env.probeLibrary.setProbeMap$For(probeMap,
getClass());

}

private boolean checkEnterprice (int x, int y){
    return (enterpriceSpace.getObjectAtX$Y(x,y)!=null);

}
```

```
//questo è il metodo che permette il controllo della posizione delle  
imprese
```

```
// che fa ritornare
```

```
//le coppie di x e y in cui è presente una impresa
```

```
public int getWalk (){
```

```
    return consumerWalk;
```

```
}
```

```
//questo metodo restituisce il passo del consumatore,
```

```
//giro poi il valore allo StandardConsumer
```

```
static public StandardEnterprice getEnterprice (int x, int y){
```

```
    return
```

```
((StandardEnterprice)enterpriceSpace.getObjectAtX$Y(x,y));
```

```
    //il metodo restituisce l'impresa nella pos X Y,
```

```
    //conversione da obj a standardEnterprice, occorre forzare  
per evitare
```

```
    //messaggio di errore, questo serve per il randomWalk
```

```
    //rendendo la classe static si supera un precedente errore in  
cui non
```

```
    //passava i valori
```

```
}
```

```
public Object buildObjects()
```

```
{
```

```
    int x, y, num, conta;
```

```
    //la variabile conta verifica quanti consumatori nascono su di  
un'impresa ed incrementa il valore
```

```
    StandardConsumer aconsumer;
```

```
    double cP;
```

```
StandardEnterprice aenterprice;

double ccP;

int cO;
int cW;
int cR;

//creo tutte le variabili necessarie al consumatore

super.buildObjects();

Globals.env.randomGenerator.setStateFromSeed(start);

//questo metodo aggiorna il punto di partenza facendo
partire la simulazione

//ogni volta da un punto diverso definito dalla variabile start
che è settabile

// a video dal menù del Model Swarm

// Ora creiamo gli oggetti del model.

// First create the marketSpace and seed it with food.

marketSpace = new MarketSpace(Globals.env.globalZone,
                               worldXSize, worldYSize);

//Creiamo una griglia bidimensionale, la quale conterrà le imprese
//e i consumatori.
// lo spazio è un toroide

consumerSpace = new Grid2dImpl(Globals.env.globalZone,
                               worldXSize, worldYSize);

consumerSpace.fastFillWithObject(null);

//griglia in cui spargo i consumatori

enterpriceSpace = new Grid2dImpl(Globals.env.globalZone,
```

```
worldXSize, worldYSize);

//griglia in cui spargo le imprese

enterpriceSpace.fastFillWithObject(null);

//la griglia parte vuota (null)
// Now create a List object to manage all the bugs we are
// about to create.

consumerList = new ListImpl(Globals.env.globalZone);

enterpriceList = new ListImpl(Globals.env.globalZone);

//creo due liste una per i consumatori ed una per le imprese
//per gestire in memoria le caratteristiche dei consumatori

//Inizio a mettere le imprese sul mercato
num = 0;
ccP = 0;
for (y = 0; y < worldYSize; y++)
    for (x = 0; x < worldXSize; x++)
        if (
Globals.env.uniformDblRand.getDoubleWithMin$withMax(
                                0.0, 1.0) <=
enterpriceDensity)
            {
                ccP = middleEnterprisePrice +
Globals.env.uniformIntRand.getIntegerWithMin$withMax(-
rangeEnterprisePrice, +rangeEnterprisePrice);

aenterprice = new StandardEnterprise(Globals.env.globalZone,
enterpriceSpace, x, y, ++num, ccP);
                enterpriceSpace.putObject$atX$Y
(aenterprice, x, y);
                enterpriceList.addLast(aenterprice);
                System.out.println("L'impresa numero " +
enterpriceList.getCount() + " è nella posizione " + x + ", " + y + "
con un prezzo di " + ccP);

                //inserisco il printline per verificare l'effettiva
```

```
        // collocazione delle imprese nello spazio
        //le imprese sono sparse in modo casuale con una
probabilità
        //definita da enterprise density
    }

    // Riporta il numero di imprese create sul mercato
    System.out.println(enterpriseList.getCount() + " Imprese sul
mercato");

    //inserisco i consumatori sul mercato

    cW = consumerWalk;
    cO = optimism;
    cR = researchCost;
    cP = 0;
    conta=0;
    num = 0;
    for (y = 0; y < worldYSize; y++)
        for (x = 0; x < worldXSize; x++)
            if (
Globals.env.uniformDblRand.getDoubleWithMin$withMax(
                                0.0, 1.0) <=
consumerDensity)

                {

                    cP = middleConsumerPrice +
Globals.env.uniformIntRand.getIntegerWithMin$withMax(-
rangeConsumerPrice, +rangeConsumerPrice);

//inserisco il prezzo variabile dei consumatori
//I consumatori partono con un prezzo medio definito da
middleConsumerPrice a cui si aggiunge
//+ o - 10 devo verificare se posso inserire due variabili per meglio
gestirlo.
```

```
aconsumer = new StandardConsumer(Globals.env.globalZone,
consumerSpace, x, y, ++num, cP, cW, cO, cR );
consumerSpace.putObject$atX$Y(aconsumer, x, y);
consumerList.addLast(aconsumer);
System.out.println("Il consumatore numero " +
consumerList.getCount() + " parte dalla posizione " + x + ", " + y
+ " con prezzo " + cP);

//inserisco il printline per verificare l'effettiva collocazione di
partenza dei consumatori nello spazio

if (checkEnterprice(x,y)) {System.out.println( "Il consumatore n "
+ consumerList.getCount()+ " in " + x + ", " + y + " è su una
impresa");
        conta++; }
    }
//Qui verifico se il consumatore parte dalla medesima posizione in
cui

//vi è una impresa. Il controllo avviene grazie al metodo definito
sopra

    // Riporta il numero di consumatori creati sul mercato
    System.out.println(consumerList.getCount() + " consumatori
sul mercato, di cui su posizione di imprese " + conta);

    return this;

}

// The next three methods return some useful information about
the
// created ModelSwarm to the caller. They will be used by the
// Observer Swarm.
```

```
public ListImpl getConsumerList()
{
    return consumerList;
}

public ListImpl getEnterpriceList()
{
    return enterpriceList;
}

public Grid2dImpl getWorld()
{
    return consumerSpace;
}

public MarketSpace getMarket()
{
    return marketSpace;
}

// This is the method a) for building the list of actions for
// these objects to accomplish and b) for scheduling these actions
// in simulated time.
public Object buildActions()
{
    Selector sel;
    ActionGroupImpl modelActions;

    // First, use the parent class to initialize the process.
    super.buildActions();

    // Then create an ActionGroup object and insert both a
    // randomWalk message to every bug in the bug list and a

    // checkTime message to modelSwarm.
    modelActions = new ActionGroupImpl(getZone());
}
```

```
        sel = SwarmUtils.getSelector("StandardConsumer",
"randomWalk");
        modelActions.createActionForEach$message(consumerList,
sel);

        sel = SwarmUtils.getSelector(this, "checkTime");
        modelActions.createActionTo$message(this, sel);

        // Now create the schedule and set the repeat interval to
unity.
        modelSchedule = new ScheduleImpl(getZone(), 1);

        // Finally, insert the action list into the schedule at period
zero
        modelSchedule.at$createAction(0, modelActions);

        return this;
    }

    // This method specifies the context in which the model is to be
run.
    public Activity activateIn(Swarm swarmContext)
    {
        // Use the parent class to activate ourselves in the context
// passed to us.
        super.activateIn(swarmContext);

        // Then activate the schedule in ourselves.
        modelSchedule.activateIn(this);

        // Finally, return the activity we have built.
        return getActivity();
    }

    // This is a pretty crude method to end the simulation after an
// arbitrary number of periods given by the endTime parameter.
//If
// the simulation time returned by getCurrentTime() is greater
// than endTime, we terminate the modelSwarm activity. The
// ObserverSwarm will pick this up and handle it. Note that this
// is no longer strictly necessary since the user can now stop the
```

```
// simulation at any time using the control panel. Still, this is
// a placeholder for a more sophisticated end-of-simulation
// routine to be introduced later.
public void checkTime()
{
    if (Globals.env.getCurrentTime() >= endTime)
        getActivity().terminate();

    return;
}
}
```

observer.java

```
// ObserverSwarm.java The observer swarm is collection of objects  
that  
// are used to run and observe the ModelSwarm that actually  
comprises  
// the simulation.
```

```
import swarm.Globals;  
import swarm.Selector;
```

```
import swarm.defobj.Zone;  
import swarm.defobj.ZoneImpl;  
import swarm.defobj.Symbol;
```

```
import swarm.gui.Colormap;  
import swarm.gui.ColormapImpl;  
import swarm.gui.ZoomRaster;  
import swarm.gui.ZoomRasterImpl;
```

```
import swarm.space.Value2dDisplay;  
import swarm.space.Value2dDisplayImpl;  
import swarm.space.Object2dDisplay;  
import swarm.space.Object2dDisplayImpl;
```

```
import swarm.simtoolsgui.GUISwarm;  
import swarm.simtoolsgui.GUISwarmImpl;
```

```
import swarm.activity.ActionGroup;  
import swarm.activity.ActionGroupImpl;  
import swarm.activity.Schedule;  
import swarm.activity.ScheduleImpl;  
import swarm.activity.Activity;
```

```
import swarm.objectbase.Swarm;  
import swarm.objectbase.SwarmImpl;  
import swarm.objectbase.EmptyProbeMap;  
import swarm.objectbase.EmptyProbeMapImpl;
```

```
import swarm.collections.ListImpl;

import swarm.analysis.EZGraph;
import swarm.analysis.EZGraphImpl;

//importo le classi per la creazione del grafico
// ObserverSwarm is a subclass of GUISwarm implementation
class.
import java.util.*;

public class ObserverSwarm extends GUISwarmImpl
{
    // Declare the display parameters and their default values.
    public int displayFrequency = 1;
    public int zoomFactor = 8;
    public int simulatedTime = 0;
    public EZGraph priceGraph;

    //Aggiungo la variabile per il grafico
    // A flag to signal the end of the simulation.

    public boolean simulationFinished = false;

    // Declare other variables local to ObserverSwarm.
    ModelSwarm modelSwarm;
    ZoomRaster worldRaster;
    Value2dDisplay marketDisplay;
    Object2dDisplay consumerDisplay;
    ScheduleImpl displaySchedule;

    Object2dDisplay enterpriceDisplay;

    // This is the constructor for a new ObserverSwarm.
    public ObserverSwarm(Zone azone)
    {
        // Use the parent class to create an observer swarm.
        super(azone);

        // Build a custom probe map. Without a probe map, the
        default
```

```
// is to show all variables and messages. Here we choose to
// customize the appearance of the probe, giving a nicer
// interface.

// Create the probe map and give it the ObserverSwarm class.
EmptyProbeMapImpl probeMap =
    new EmptyProbeMapImpl(azone, getClass());

// Now add probes for the variables we wish to probe, using
// the method in our SwarmUtils class.
probeMap.addProbe(getProbeForVariable("displayFrequency"));
probeMap.addProbe(getProbeForVariable("zoomFactor"));
probeMap.addProbe(getProbeForVariable("simulatedTime"));
);

// And finally install our probe map into the probeLibrary.
// Note that this library object was automatically created by
// initSwarm.
Globals.env.probeLibrary.setProbeMap$For(probeMap,
getClass());
}

public Object _priceGraphDeath_ (Object caller) {
priceGraph.drop ();
priceGraph = null;
return this;
}

// aggiungo il grafico

// The buildObjects method.
public Object buildObjects()
{
    Zone modelZone;
    Colormap colormap;
    Selector sel;

// Use the parent class to initialize the process.
super.buildObjects();
```

```
    // First we create the model that we're actually observing, by
    // creating an instance of the ModelSwarm class,
modelSwarm.
    // modelSwarm will now be a subSwarm of this top-level
    // ObserverSwarm rather than the top-level Swarm in its own
    // right. We create modelSwarm in its own newly-created
Zone
of
    // so that modelSwarm's storage is segregated from the rest
    // of
    // the application. Note that as in SimpleSwarmBug3, we are
    // reading the modelSwarm parameters from a file and so use
    // the List Archiver to create modelSwarm.

modelZone = new ZoneImpl(getZone());
modelSwarm =

(ModelSwarm)Globals.env.lispAppArchiver.getWithZone$key(
    modelZone, "modelSwarm");

    // Now create probe objects on the model and on ourselves.
    // This provides a simple interface to allow the user to
    // change the model and simulation parameters. Note that the
    // createArchivedProbeDisplay() method is provided by the
Swarm
    // environment, set up by the call to initSwarm().
    Globals.env.createArchivedProbeDisplay(modelSwarm,
"modelSwarm");
    Globals.env.createArchivedProbeDisplay(this,
"observerSwarm");

    // Instruct the control panel to wait for a button event: we
    // halt here until someone hits a control panel button.
    // Eventually this will allow the user a chance to fill in
    // parameters before the simulation runs.
getControlPanel().setStateStopped();

    // OK - the user has pressed a button. Now we're ready to
    // start.
```

```
// Allow the model swarm to build its objects.
modelSwarm.buildObjects();

// Now build the GUI display objects.

// First, create a colormap, the correspondence between a
// color and a byte integer code. This is a global resource
// which is used by lots of different objects. Then set the
// three colors we will be using. Since the FoodSpace grid
// uses one to indicate a cell with food and zero to indicate
// a cell without food, FoodSpace cells will be displayed in
// red or black depending on whether they contain food or
not.
// We'll use green to indicate the location of our bugs.
// (Yellow is reserved for future use.)
colormap = new ColormapImpl(getZone());
colormap.setColor$ToName((byte)0, "black");
colormap.setColor$ToName((byte)1, "red");
colormap.setColor$ToName((byte)2, "green");
colormap.setColor$ToName((byte)3, "yellow");
colormap.setColor$ToName((byte)4, "blue");
//modifico il set color aggiungendo il blu ed il giallo.

// Now tell each of the bugs in the model to set its default
// display color to green (2). We do this by getting the list
// of bugs created in modelSwarm and iterating through it.
ListImpl consumerList = modelSwarm.getConsumerList();
for (int i = 0; i < consumerList.getCount(); i++)
{
    StandardConsumer consumer =
(StandardConsumer)consumerList.atOffset(i);
    consumer.setConsumerColor((byte)4);
}

//setto il colore standard per i consumatori, si può poi
modificare all'interno della classe
ListImpl enterpriceList = modelSwarm.getEnterpriceList();
for (int i = 0; i < enterpriceList.getCount(); i++)
{
    StandardEnterprice enterprice =
(StandardEnterprice)enterpriceList.atOffset(i);
```

```
        enterprice.setEnterpriceColor((byte)3);
    }
    //setto il colore standard per le imprese, che potrò modificare
    all'interno della classe

    // Next, create a "raster widget", a 2-dimensional display
    // window. We tell the raster what to do if it dies, give it
    // its colormap, set its zoom factor (its actual size on the
    // display screen), set its virtual dimensions to the size of
    // our world, and give it its title.
    worldRaster = new ZoomRasterImpl (getZone(),
"worldRaster");
    sel = SwarmUtils.getSelector(this, "_worldRasterDeath_");
    worldRaster.enableDestroyNotification$notificationMethod
        (this, sel);
    worldRaster.setColormap (colormap);
    worldRaster.setZoomFactor (zoomFactor);
    worldRaster.setWidth$Height((modelSwarm.getWorld()).get
SizeX(),
(modelSwarm.getWorld()).getSizeY());
    worldRaster.setWindowTitle ("Market World");

    // This instructs the raster to digest all the information we
    // have just given it and to initialize itself for display.
    worldRaster.pack();

    // Now create a Value2dDisplay, an object that will display
    an
    // arbitrary 2-dimensional value array, in this case our
    // foodspace, on the raster widget. Think of the foodspace
    // lattice overlaying the raster. Remember that we have set
    // the colormap such that cells with no food (0's) will be
    // displayed in black and cells with food (1's) will be
    // displayed in red. We use the Value2dDisplay
    implementation
    // class and give it a zone, the raster, the colormap and the
    // foodspace.
    marketDisplay = new Value2dDisplayImpl(getZone(),
worldRaster,
```

```
        colormap,
        modelSwarm.getMarket());

    // Also create an Object2dDisplay that will display the bugs
    // on the raster, giving it the raster, the grid on which the
    // objects (bugs) are located, and the draw message to the
    // bugs. (The Object2dDisplay relies on the objects to send
    // their own draw messages to the raster.) Once the
    // Object2dDisplay is created, we give it the list of bugs to
    // which the draw message needs to be sent. Again, think of
    // the bugspace as overlaying the raster.)
    sel = SwarmUtils.getSelector("StandardConsumer",
    "drawSelfOn");
    consumerDisplay = new Object2dDisplayImpl(getZone(),
    worldRaster,
        modelSwarm.getWorld(), sel);
    consumerDisplay.setObjectCollection(modelSwarm.getConsumerList());

    sel = SwarmUtils.getSelector("StandardEnterprice",
    "DrawSelfOn");
    enterpriceDisplay = new Object2dDisplayImpl(getZone(),
    worldRaster,
        modelSwarm.getWorld(), sel);
    enterpriceDisplay.setObjectCollection(modelSwarm.getEnterpriceList());
    //inizia qui il grafico che ho aggiunto
    priceGraph = new EZGraphImpl
    (getZone (),
    "Middle ConsumerPrice vs. time",
    "time", "consumerPrice",
    "priceGraph");

    //Globals.env.setWindowGeometryRecordName (unhappyGraph,
    "unhappyGraph");

    // instruct this _unhappyGraphDeath_ method to be called when
    // the widget is destroyed
    try {
        priceGraph.enableDestroyNotification$notificationMethod
        (this, new Selector (getClass (),
```

```
        "_priceGraphDeath_",
        false));
    } catch (Exception e) {
        System.err.println ("Exception _priceGraphDeath_: "
            + e.getMessage ());
    }

    // create the data for the average heatbug unhappiness
    try {

priceGraph.createAverageSequence$withFeedFrom$andSelector
    ("consumerPrice", modelSwarm.getConsumerList (),
        new Selector (Class.forName ("StandardConsumer"),
            "getConsumerPrice",
                false));
    } catch (Exception e) {
        System.err.println ("Exception getConsumerPrice: "
            + e.getMessage ());
    }

        return this;
        //devo verificar che non andasse prima il grafico de return
    }

    public Object _update_ () {
if (priceGraph != null)
    priceGraph.step ();
    return this;
    }
    //lo aggingo per il grafico

    public Object buildActions()
    {
        Selector sel;
        ActionGroupImpl displayActions;

        // Use the parent class to begin the process.
```

```
super.buildActions();

// Call on the model swarm to build and schedule its actions.
modelSwarm.buildActions();

// fino a qui aggiungo dal Hb!!!!!!!

// Create an ActionGroup for display. This is a list of
// display actions that we want to occur at each step in
// simulation time. First we tell the foodDisplay and the
// bugDisplay to display themselves on the raster widget, and
// then tell the raster widget to display itself on the
// screen. "doTkEvents", is required at the end to make
// everything happen. We then check to see if modelSwarm
has
// told us to stop the simulation.
displayActions = new ActionGroupImpl(getZone());

try {
displayActions.createActionTo$message
(this, new Selector (getClass (), "_update_", false));

// Schedule the update of the probe displays
displayActions.createActionTo$message
(Globals.env.probeDisplayManager,
new Selector (Globals.env.probeDisplayManager.getClass (),
"update", true));

// Finally, schedule an update for the whole user
// interface code. This is crucial: without this, no
// graphics update and the control panel will be
// dead. It's best to put it at the end of the display
// schedule
displayActions.createActionTo$message
(getActionCache (), new Selector
(getActionCache ().getClass (), "doTkEvents", true));
} catch (Exception e) {
System.err.println ("Exception in setting up displayActions : "
```

```
        + e.getMessage ());
    }

    sel = SwarmUtils.getSelector(marketDisplay, "display");
    displayActions.createActionTo$message(marketDisplay,
sel);

    sel = SwarmUtils.getSelector(consumerDisplay, "display");
    displayActions.createActionTo$message(consumerDisplay,
sel);

    sel = SwarmUtils.getSelector(enterpriceDisplay, "display");
    displayActions.createActionTo$message(enterpriceDisplay,
sel);

    sel = SwarmUtils.getSelector(worldRaster, "drawSelf");
    displayActions.createActionTo$message(worldRaster, sel);

    sel = SwarmUtils.getSelector(this, "updateSimulatedTime");
    displayActions.createActionTo$message(this, sel);

    sel =
SwarmUtils.getSelector(Globals.env.probeDisplayManager,
                        "update");
    displayActions.createActionTo$message(Globals.env.probe
DisplayManager,
                                        sel);

    sel = SwarmUtils.getSelector(getActionCache(),
"doTkEvents");
    displayActions.createActionTo$message(getActionCache(),
sel);

    sel = SwarmUtils.getSelector(this, "checkForDone");
    displayActions.createActionTo$message(this, sel);

    // Finally, put the ActionGroup into a display schedule.
    displaySchedule = new ScheduleImpl(getZone(),
displayFrequency);
    displaySchedule.at$createAction(0, displayActions);
```

```
        return this;
    }

    // Activate the schedules so that they are ready to run. The
    // swarmContext argument is the zone in which the
ObserverSwarm is
    // activated. Typically the ObserverSwarm is the top-level
swarm,
    // so it is activated in "null". The other (sub)swarms and
    // schedules will be activated inside of the ObserverSwarm
    // context.
    public Activity activateIn(Swarm swarmContext)
    {
        // Use the parent class to activate ourselves in the context
        // passed to us.
        super.activateIn(swarmContext);

        // Now activate the model swarm in the ObserverSwarm
context.
        modelSwarm.activateIn(this);

        // Then activate the ObserverSwarm schedule in the
        // ObserverSwarm context.
        displaySchedule.activateIn(this);

        // Finally, return the activity we have built - the thing that
        // is ready to run.
        return getActivity();
    }
}
```

```
// This method checks each period to see if the simulation is
// done. It is called at the end of each period.
```

```
//inserisco la parte finale per il grafico
```

```
public Object graphConsumer (StandardConsumer aConsumer) {
    if (priceGraph != null)
        try {
            priceGraph.createSequence$withFeedFrom$andSelector
                ("Consumer", aConsumer, new Selector
(aConsumer.getClass (),
                    "getConsumerPrice", false));
        } catch (Exception e) {
            System.err.println ("Exception graphConsumer: " +
e.getMessage());
        }
        return this;
    }
    public void drop () {
        if (priceGraph != null)
            priceGraph.disableDestroyNotification ();
        super.drop ();
    }

    public void checkForDone()
    {
        if (simulationFinished)
            {
                // The simulation is over. Presumably we got here because
                // the user did not QUIT when told to do so after the
                // modelSwarm activity was terminated. We therefore
chide
                // her and press QUIT ourselves.
                System.out.println("I said to QUIT!");
                modelSwarm.getActivity().terminate();
                modelSwarm.drop();
                getControlPanel().setStateQuit();
            }

        else if (modelSwarm.getActivity().getStatus() ==
            Globals.env.Completed)
            {
                // modelSwarm has signaled us that the simulation is
                // finished by terminating itself. (ObserverSwarm sees
```

```
        // this as Completed".) Press the STOP button on the
        // control panel. Pressing STOP rather than QUIT leaves
        // the raster window and the control panel on the display
        // so that the user can look at the results of the
        // simulation. (Those windows will disappear when the
        // user presses QUIT.) We also set a flag to indicate
        // that the simulation is over, in case the user presses
        // START or NEXT instead of QUIT.
        simulationFinished = true;
        System.out.println("The simulation ended after "
            + Globals.env.getCurrentTime()
            + " periods.");
        System.out.println("Press QUIT when ready.");
        getControlPanel().setStateStopped();
    }
}

// We want simulatedTime to keep track of the current time so
that
// we can display it in our custom probe map.
public void updateSimulatedTime()
{
    simulatedTime = Globals.env.getCurrentTime();
}

// This is a method given to the raster object to tell it what to
// do in the event of an untimely death.
public Object _worldRasterDeath_ (Object caller)
{
    worldRaster.drop ();
    worldRaster = null;
    return this;
}
}
```

StandardConsumer.java

```
// StandardConsumer.java

import swarm.Globals;
import swarm.defobj.Zone;
import swarm.objectbase.SwarmObjectImpl;

import swarm.space.Discrete2dImpl;
import swarm.space.Grid2dImpl;

import swarm.gui.Raster;
import swarm.gui.ZoomRasterImpl;
import java.io.*;
public class StandardConsumer extends SwarmObjectImpl
{
    // These instance variables keep track of a given consumer's
marketspace,
    // position and identity. We also save the dimensions of the
    // marketspace so that we can make fewer calls to the getSizeX()
and
    // getSizeY() methods in the consumer's randomWalk().

    public Grid2dImpl myConsumerSpace;
    public int xPos;
    public int yPos;
    public int consumerNumber;
    int worldXSize;
    int worldYSize;
    public double consumerPrice;

    public int consumerWalk;
    public int consumerOptimism;
    //inserisco la variabile per rendere selezionabile la lunghezza del
passo

    public byte consumerColor;
    public int consumerResearch;
```

```
// Constructor to create a StandardConsumer object in Zone
aZone and to
// place it in the consumerspace, fSpace and cSpace, at
// the specified X,Y location. The consumer is also given a
numeric id,
// bNum.

public StandardConsumer(Zone aZone, Grid2dImpl cSpace,
                        int X, int Y, int cNum, double cPrice, int cWalk,
int cOptimism, int cResearch)
{
    // Call the constructor for the bug's parent class.
    super(aZone);

    // Record the bug's foodspace, bugspace, initial position and
    // id number.
    //Inizializzo le variabili
    myConsumerSpace = cSpace;
    worldXSize = 80;
    worldYSize = 80;
    xPos = X;
    yPos = Y;
    consumerNumber = cNum;
    consumerPrice = cPrice;
    consumerWalk = cWalk;
    consumerOptimism = cOptimism;
    consumerResearch = cResearch;
}

// consumerWalk= modelSwarm.getWalk();

// This is the method to have the bug take a random walk
backward
// (-1), forward (+1), or not at all (0) in first the X and then
// the Y direction. The randomWalk method uses
// getIntegerWithMin$withMax() to return an integer between a
// minimum and maximum value, here between -1 and +1.
// Globals.env.uniformRand is an instance of the class
// UniformIntegerDistImpl, instantiated by the call to
```

```
// Globals.env.initSwarm in StartSimpleBug. Note that the bug's
// world is a torus. If the bug walks off the edge of its
// rectangular world, it is magically transported (via the modulus
// operator) to the opposite edge. If on its walk the bug finds
// food, it eats it and turns on the haveEaten flag so it can
// report its feast if asked. Note that before the bug actually
// moves, we must check to see that there is no other bug at the
// destination cell. If there is, the this bug just stays put.

public double getConsumerPrice () {
    return consumerPrice;
}

//metodo che restituisce all'observer il consumer price per
ilgrafico

public void randomWalk()
{

    int newX, newY;
    double saving;
    //saving è il risparmio compiuto dal consumatore
nell'acquistare ad un
    //prezzo inferiore a quello predefinito.

    //aConsumerWalk = ModelSwarm.getWalk;
    //    consumerWalk= modelSwarm.getWalk();
    //con il metodo Random Walk decido dove deve muoversi in
modo
    //casuale il consumatore, il passo è compreso tra -3 et + 3
caselle
    //all'interno del mondo
    newX = xPos +

Globals.env.uniformIntRand.getIntegerWithMin$withMax(-
consumerWalk, +consumerWalk);
    newY = yPos +

Globals.env.uniformIntRand.getIntegerWithMin$withMax(-
consumerWalk, +consumerWalk);
    newX = (newX + worldXSize) % worldXSize;
```

```
newY = (newY + worldYSize) % worldYSize;

//      System.out.println( "Il consumatore " +
consumerNumber + " si sposta nella posizione " + newX + " " +
newY +" con un prezzo di " + consumerPrice);
//passo a video la posizione dei consumatori
//disabilito la funzione verificato il corretto svolgimento
della procedura

if (myConsumerSpace.getObjectAtX$Y(newX, newY) ==
null)
{
myConsumerSpace.putObject$atX$Y(null, xPos,
yPos);
xPos = newX;
yPos = newY;
myConsumerSpace.putObject$atX$Y(this, xPos,
yPos);
StandardEnterprice enterprice =
ModelSwarm.getEnterprice(xPos, yPos);
//carico le caratteristiche delle imprese, se le imprese
sono
//sulle posizioni x y implica != da null allora se anche
//il prezzo importato da get.Price è minore del prezzo
// di riserva del consumatore allora il consumatore
//acquista e aggiorna il suo prezzo a quello a cui ha
acquistato
//ovvero abbassa , se non era già uguale il suo prezzo
di acquisto a quello
// a cui l'impresa vende. Per vedere a video il tutto il
consumatore contento di avere acquistato
//cambia colore e diventa di colore verde.
//A conferma dell' avvenuta scambio a video si stampa
if (enterprice != null)
if (enterprice.getPrice()<=consumerPrice) {
consumerColor= 2;

//cambio il colore quando acquisto
saving = consumerPrice - enterprice.getPrice();
```

```
        System.out.println( "il consumatore " +
consumerNumber + " acquista dall'impresa " +
enterprice.getNum()+ " al prezzo di " + enterprice.getPrice());
        System.out.println( " risparmiando rispetto a
quanto era disposto a spendere "+ saving);

        consumerPrice= enterprice.getPrice();
        consumerPrice =consumerPrice- consumerOptimism;
        //metodo per il consumatore "egoista" butta giù il prezzo.
        System.out.println( "La sua disponibilità a pagare, scende
al prezzo di acquisto ridotto di "+ consumerOptimism + "(new p="
+ consumerPrice+ ")." );
        System.out.println(" ");
    }
    else{ consumerColor= 1;
        //il conusmatore arrabbiato diventa rosso.
        //aggiorna il suo prezzo
        System.out.println("Il consumatore "+
consumerNumber + " non compra dall' impresa
"+enterprice.getNum()+" perchè il prezzo ("
+enterprice.getPrice()+ ") è troppo caro!");
        System.out.println("il consumatore non voleva
spendere più di " + consumerPrice);
        consumerPrice=enterprice.getPrice()-
consumerResearch;
        System.out.println("Sconsolato, aumenta la sua
disponibilità a pagare a "+consumerPrice+ ".");
        System.out.println(" ");
        //aggiorno il prezzo del consumatore;
    }
}

}

// Metodo che definisce il colore: 3 = giallo , 1= rosso, 4= blu,
2=verde

public Object setConsumerColor(byte c)
```

```
{
    consumerColor = 4;
    return this;
}

public Object drawSelfOn (Raster r)
{
    r.drawPointX$Y$Color (xPos, yPos, consumerColor);
    return this;
}

}
```

StandardEnterprise.java

```
// StandardEnterprise.java

import swarm.Globals;
import swarm.defobj.Zone;
import swarm.objectbase.SwarmObjectImpl;

import swarm.space.Discrete2dImpl;
import swarm.space.Grid2dImpl;

import swarm.gui.Raster;
import swarm.gui.ZoomRasterImpl;

public class StandardEnterprise extends SwarmObjectImpl
{

    public Grid2dImpl myEnterpriseSpace;
    public int xPos;
    public int yPos;
    public int enterpriseNumber;
    int worldXSize;
    int worldYSize;
    double enterprisePrice;

    public byte enterpriseColor;

    public StandardEnterprise(Zone aZone, Grid2dImpl eSpace,
                             int X, int Y, int eNum, double ePrice)
    {
        // Call the constructor for the bug's parent class.
        super(aZone);

        // Record the bug's foodspace, bugspace, initial position and
        // id number.

        myEnterpriseSpace = eSpace;
```

```
worldXSize = 80;
worldYSize = 80;
xPos = X;
yPos = Y;
enterpriceNumber = eNum;
enterpricePrice = ePrice;

}

public int getNum (){
    return enterpriceNumber;

    //rende il numero delle imprese
}

public double getPrice (){
    return enterpricePrice ;
    //rende il prezzo delle singole imprese
}

public Object setEnterpriceColor(byte c)
{
    enterpriceColor = 3;
    return this;
}

public Object DrawSelfOn (Raster r)
{
    r.drawPointX$Y$Color (xPos, yPos, enterpriceColor);
    return this;
}
}
```

StartConsumer.java

```
// StartConsumer.java
// Java application.

import swarm.Globals;
import swarm.defobj.Zone;

public class StartConsumer
{
    public static void main (String[] args)
    {
        ObserverSwarm displaySwarm;
        // Swarm initialization: all Swarm apps must call this first.
        // int start;
        //start =10;
        Globals.env.initSwarm ("StandardConsumer", "1.1",
                               "s.iacomussi@inwind.it", args);
        //
        Globals.env.randomGenerator.setStateFromSeed(start);
        //con questo metodo , presso dalla guida di swarm inserisco
un numero casuale per fare partire
        //la simulazione da punti diversi

        // Create a top-level Swarm object, now DisplaySwarm, and
        // build its internal objects and activities. Note that we use
        // the Lisp Archiver to create displaySwarm and to load the
        // model's display parameters from the a file. The default
        // filename is the name given by the first argument to
        // initSwarm, above, with the file extension .scm. In this
        // case that is SimpleBug.scm.

        // "displaySwarm", the second argument to
getWithZone$key(), is
        // the key in the .scm file which contains the values of the
        // display parameters in the ObserverSwarm class. For now
        // there is only one parameter, displayFrequency.
```

```
displaySwarm =  
(ObserverSwarm)Globals.env.lispAppArchiver.getWithZone$key(  
    Globals.env.globalZone,  
    "displaySwarm");  
  
    displaySwarm.buildObjects();  
    displaySwarm.buildActions();  
    displaySwarm.activateIn(null);  
  
    // Now start the displaySwarm and the control panel it  
    // provides.  
    displaySwarm.go();  
  
    // The user has pressed Quit. Drop everything and return.  
    displaySwarm.drop();  
    }  
}
```

swarmUtil.java

```
import swarm.Globals;
import swarm.Selector;

public class SwarmUtils
{
    // These two static methods create a selector. A selector is an
    // object of the Selector class used by Swarm to encapsulate a
    // "message" destined for an object, where the message is the
name
    // of a method defined for the class to which the object belongs.
    // Because the method must indeed be defined for the class of the
    // object and because this can be determined only at run time,
    // there is a possibility that the creation of the selector will
    // throw an exception if the class and the method do not match.
    // Java requires that events that might throw exceptions be
    // enclosed in try/catch blocks. If there is an error creating
    // the new selector in the try block, the catch block can handle
    // the resulting exception. Here we have taken a pretty crude
    // approach to handling the exception: we simply call
    // System.exit(1). (Note that the "return null" which ends the
    // catch block is there only to tell the compiler that "return
    // sel" will never be reached if an exception occurs and sel is
    // undefined. We'll exit on an exception before ever returning
    // sel to the calling method.)

    // Note that the getSelector method overloaded. It can be
    // called with either a string containing the class name as its
    // first argument, or with an object of the desired class. In the
    // first case, the string is converted to a class identifier using
    // the forName() method, while in the second case getClass() is
    // used to find the class identifier for the object. The second
    // argument to getSelector is always a string containing the
    // method name. (The boolean "false" at the end of the Selector
    // constructor is theobjCFlag. It allows one to use
    // ObjectiveC-type key/value method syntax. Since we always
use
    // Java-style method names, for us the flag is always false.)
```

```
public static Selector getSelector(String name, String method)
{
    Selector sel;

    try
    {
        sel = new Selector(Class.forName(name), method, false);
    } catch (Exception e)
    {
        System.err.println("There was an error in creating a
Selector for method "
+ method + "\nin Class " + name +
".");
        System.err.println (name + "." + method + " returns "
+ e.getMessage());
        System.err.println("The process will be terminated.");
        System.exit(1);
        return null;
    }

    return sel;
}

public static Selector getSelector(Object obj, String method)
{
    Selector sel;

    try
    {
        sel = new Selector(obj.getClass(), method, false);
    } catch (Exception e)
    {
        System.err.println("There was an error in creating a
Selector for method "
+ method + "\nin Class " +
(obj.getClass()).getName() + ".");
        System.err.println ((obj.getClass()).getName() + "." +
method
+ " returns " + e.getMessage());
        System.err.println("The process will be terminated.");
    }
}
```

```
        System.exit(1);  
        return null;  
    }  
    return sel;  
}  
}
```

MarketSpace.java

```
// MarketSpace.java
// Defines the MarketSpace class as a subclass of Discrete2dImpl.

import swarm.Globals;
import swarm.defobj.Zone;
import swarm.space.Discrete2dImpl;

public class MarketSpace extends Discrete2dImpl
{
    // The constructor for a foodspace. Create a foodspace in zone
    // aZone with dimensions xSize and ySize. Note that a
    // food space is based on the Swarm's Discrete2dImpl class.
    public MarketSpace(Zone aZone, int xSize, int ySize)
    {
        // Call the constructor for the parent class and then fill the
        // lattice with zeros.
        super(aZone, xSize, ySize);
        fastFillWithValue(0);
    }
}
```

Makefile

JAVA_SRC = StartConsumer.java ObserverSwarm.java

ModelSwarm.java StandardConsumer.java

StandardEnterprise.java MarketSpace.java SwarmUtils.java

all: \$(JAVA_SRC)

\$(SWARMHOME)/bin/javacswarm \$(JAVA_SRC)

clean:

rm -f *.class

cleanall:

rm -f *.class

rm -f *~

rm *.stackdump

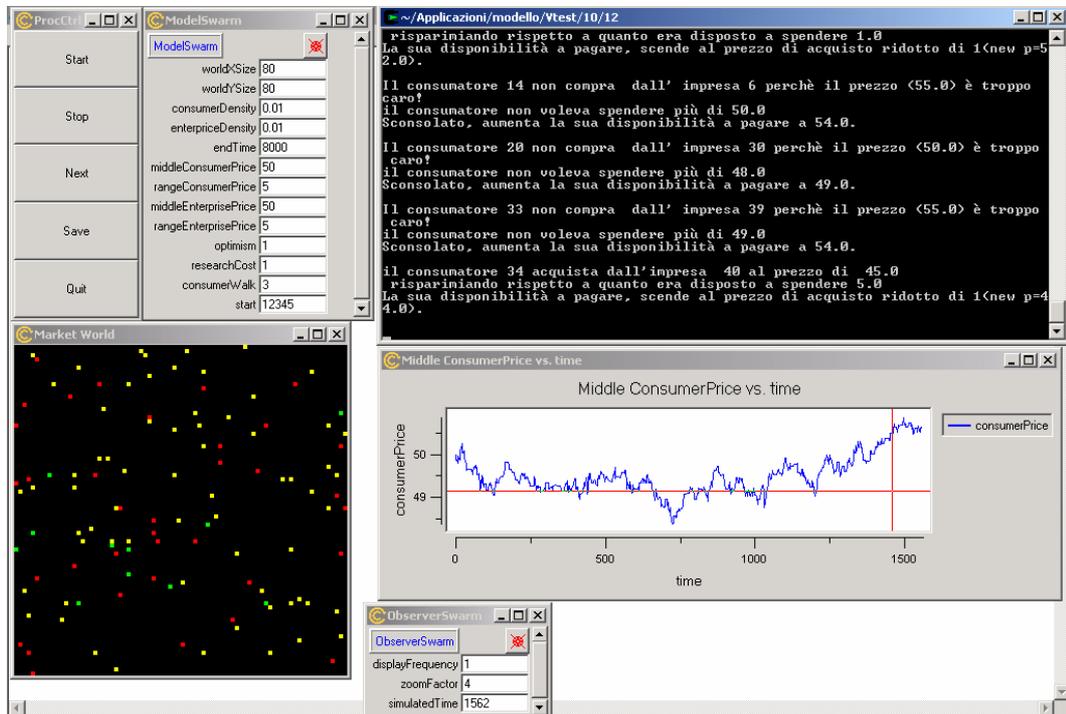
StandardConsumer.scn

```
(list  
  
(  
  
  cons 'displaySwarm  
  
    (  
      make-instance 'ObserverSwarm  
  
        #:displayFrequency 1  
        #:zoomFactor 4  
  
    )  
  
  )  
(  
  
  cons 'modelSwarm  
  
    (  
  
      make-instance 'ModelSwarm  
  
        #:worldXSize 80  
        #:worldYSize 80  
        #:consumerDensity 0.001  
        #:enterpriceDensity 0.01  
        #:endTime 8000  
  
    )  
  
)
```

Appendice II

L'output del modello Consumer

Output delle finestre grafiche



Stampa dell'output del modello

L'impresa numero 1 è nella posizione 26, 0 con un prezzo di 49.0
L'impresa numero 2 è nella posizione 51, 1 con un prezzo di 48.0
L'impresa numero 3 è nella posizione 9, 3 con un prezzo di 49.0
L'impresa numero 4 è nella posizione 11, 3 con un prezzo di 51.0
L'impresa numero 5 è nella posizione 18, 6 con un prezzo di 53.0
L'impresa numero 6 è nella posizione 27, 7 con un prezzo di 53.0
L'impresa numero 7 è nella posizione 41, 7 con un prezzo di 48.0
L'impresa numero 8 è nella posizione 78, 7 con un prezzo di 46.0
L'impresa numero 9 è nella posizione 77, 9 con un prezzo di 51.0
L'impresa numero 10 è nella posizione 8, 11 con un prezzo di 55.0
L'impresa numero 11 è nella posizione 40, 12 con un prezzo di 52.0
L'impresa numero 12 è nella posizione 33, 13 con un prezzo di 55.0
L'impresa numero 13 è nella posizione 0, 14 con un prezzo di 53.0
L'impresa numero 14 è nella posizione 43, 14 con un prezzo di 49.0
L'impresa numero 15 è nella posizione 53, 14 con un prezzo di 45.0
L'impresa numero 16 è nella posizione 15, 15 con un prezzo di 53.0
L'impresa numero 17 è nella posizione 18, 16 con un prezzo di 48.0
L'impresa numero 18 è nella posizione 18, 17 con un prezzo di 50.0
L'impresa numero 19 è nella posizione 30, 17 con un prezzo di 51.0
L'impresa numero 20 è nella posizione 23, 19 con un prezzo di 52.0
L'impresa numero 21 è nella posizione 40, 21 con un prezzo di 50.0
L'impresa numero 22 è nella posizione 37, 22 con un prezzo di 51.0
L'impresa numero 23 è nella posizione 73, 22 con un prezzo di 51.0
L'impresa numero 24 è nella posizione 1, 24 con un prezzo di 50.0
L'impresa numero 25 è nella posizione 26, 24 con un prezzo di 51.0
L'impresa numero 26 è nella posizione 18, 25 con un prezzo di 50.0
L'impresa numero 27 è nella posizione 57, 25 con un prezzo di 48.0
L'impresa numero 28 è nella posizione 10, 27 con un prezzo di 49.0
L'impresa numero 29 è nella posizione 16, 28 con un prezzo di 46.0
L'impresa numero 30 è nella posizione 2, 29 con un prezzo di 51.0
L'impresa numero 31 è nella posizione 66, 30 con un prezzo di 47.0
L'impresa numero 32 è nella posizione 14, 31 con un prezzo di 52.0
L'impresa numero 33 è nella posizione 20, 31 con un prezzo di 47.0
L'impresa numero 34 è nella posizione 30, 31 con un prezzo di 51.0
L'impresa numero 35 è nella posizione 46, 32 con un prezzo di 47.0
L'impresa numero 36 è nella posizione 73, 35 con un prezzo di 48.0
L'impresa numero 37 è nella posizione 68, 37 con un prezzo di 53.0
L'impresa numero 38 è nella posizione 13, 38 con un prezzo di 47.0
L'impresa numero 39 è nella posizione 44, 39 con un prezzo di 46.0

L'impresa numero 40 è nella posizione 60, 39 con un prezzo di 46.0
L'impresa numero 41 è nella posizione 36, 40 con un prezzo di 49.0
L'impresa numero 42 è nella posizione 4, 41 con un prezzo di 47.0
L'impresa numero 43 è nella posizione 71, 44 con un prezzo di 47.0
L'impresa numero 44 è nella posizione 18, 45 con un prezzo di 51.0
L'impresa numero 45 è nella posizione 44, 46 con un prezzo di 54.0
L'impresa numero 46 è nella posizione 43, 47 con un prezzo di 51.0
L'impresa numero 47 è nella posizione 46, 48 con un prezzo di 52.0
L'impresa numero 48 è nella posizione 59, 48 con un prezzo di 51.0
L'impresa numero 49 è nella posizione 24, 49 con un prezzo di 50.0
L'impresa numero 50 è nella posizione 51, 50 con un prezzo di 49.0
L'impresa numero 51 è nella posizione 72, 50 con un prezzo di 53.0
L'impresa numero 52 è nella posizione 2, 51 con un prezzo di 53.0
L'impresa numero 53 è nella posizione 73, 54 con un prezzo di 55.0
L'impresa numero 54 è nella posizione 30, 59 con un prezzo di 45.0
L'impresa numero 55 è nella posizione 33, 59 con un prezzo di 55.0
L'impresa numero 56 è nella posizione 43, 59 con un prezzo di 55.0
L'impresa numero 57 è nella posizione 69, 61 con un prezzo di 53.0
L'impresa numero 58 è nella posizione 12, 63 con un prezzo di 50.0
L'impresa numero 59 è nella posizione 28, 64 con un prezzo di 52.0
L'impresa numero 60 è nella posizione 29, 64 con un prezzo di 50.0
L'impresa numero 61 è nella posizione 38, 67 con un prezzo di 54.0
L'impresa numero 62 è nella posizione 2, 69 con un prezzo di 51.0
L'impresa numero 63 è nella posizione 70, 71 con un prezzo di 51.0
L'impresa numero 64 è nella posizione 39, 73 con un prezzo di 52.0
L'impresa numero 65 è nella posizione 37, 74 con un prezzo di 53.0
L'impresa numero 66 è nella posizione 6, 76 con un prezzo di 55.0
L'impresa numero 67 è nella posizione 62, 76 con un prezzo di 49.0
L'impresa numero 68 è nella posizione 75, 76 con un prezzo di 50.0
L'impresa numero 69 è nella posizione 66, 77 con un prezzo di 52.0
L'impresa numero 70 è nella posizione 71, 77 con un prezzo di 47.0
L'impresa numero 71 è nella posizione 24, 78 con un prezzo di 49.0

71 Imprese sul mercato

Il consumatore numero 1 parte dalla posizione 74, 38 con prezzo 48.0

Il consumatore numero 2 parte dalla posizione 34, 39 con prezzo 45.0

Il consumatore numero 3 parte dalla posizione 44, 40 con prezzo 52.0

Il consumatore numero 4 parte dalla posizione 66, 48 con prezzo 48.0

Il consumatore numero 5 parte dalla posizione 29, 56 con prezzo 53.0

Il consumatore numero 6 parte dalla posizione 44, 63 con prezzo 49.0

6 consumatori sul mercato, di cui su posizione di imprese 0

Il consumatore 2 non compra dall'impresa 39 perchè il prezzo (46.0) è troppo caro!

il consumatore non voleva spendere più di 45.0

Sconsolato, aumenta la sua disponibilità a pagare a 45.0.

il consumatore 1 acquista dall'impresa 38 al prezzo di 47.0

risparmiando rispetto a quanto era disposto a spendere 1.0

La sua disponibilità a pagare, scende al prezzo di acquisto ridotto di 1 (new $p=46.0$).

Il consumatore 2 non compra dall'impresa 39 perchè il prezzo (46.0) è troppo caro!

il consumatore non voleva spendere più di 45.0

Sconsolato, aumenta la sua disponibilità a pagare a 45.0.

Il consumatore 1 non compra dall'impresa 42 perchè il prezzo (47.0) è troppo caro!

il consumatore non voleva spendere più di 46.0

Sconsolato, aumenta la sua disponibilità a pagare a 46.0.

Il consumatore 4 non compra dall'impresa 41 perchè il prezzo (49.0) è troppo caro!

il consumatore non voleva spendere più di 48.0

Sconsolato, aumenta la sua disponibilità a pagare a 48.0.

Il consumatore 4 non compra dall'impresa 45 perchè il prezzo (54.0) è troppo caro!

il consumatore non voleva spendere più di 48.0

Sconsolato, aumenta la sua disponibilità a pagare a 53.0.

Il consumatore 1 non compra dall'impresa 44 perchè il prezzo (51.0) è troppo caro!

il consumatore non voleva spendere più di 46.0

Sconsolato, aumenta la sua disponibilità a pagare a 50.0.

il consumatore 4 acquista dall'impresa 39 al prezzo di 46.0
risparmiando rispetto a quanto era disposto a spendere 7.0
La sua disponibilità a pagare, scende al prezzo di acquisto ridotto di
1(new $p=45.0$).

il consumatore 3 acquista dall'impresa 46 al prezzo di 51.0
risparmiando rispetto a quanto era disposto a spendere 1.0
La sua disponibilità a pagare, scende al prezzo di acquisto ridotto di
1(new $p=50.0$).

Il consumatore 2 non compra dall'impresa 44 perchè il prezzo
(51.0) è troppo caro!
il consumatore non voleva spendere più di 45.0
Sconsolato, aumenta la sua disponibilità a pagare a 50.0.

il consumatore 3 acquista dall'impresa 2 al prezzo di 48.0
risparmiando rispetto a quanto era disposto a spendere 2.0
La sua disponibilità a pagare, scende al prezzo di acquisto ridotto di
1(new $p=47.0$).

Il consumatore 3 non compra dall'impresa 64 perchè il prezzo
(52.0) è troppo caro!
il consumatore non voleva spendere più di 47.0
Sconsolato, aumenta la sua disponibilità a pagare a 51.0.

Il consumatore 2 non compra dall'impresa 61 perchè il prezzo
(54.0) è troppo caro!
il consumatore non voleva spendere più di 50.0
Sconsolato, aumenta la sua disponibilità a pagare a 53.0.

Il consumatore 5 non compra dall'impresa 12 perchè il prezzo
(55.0) è troppo caro!
il consumatore non voleva spendere più di 53.0
Sconsolato, aumenta la sua disponibilità a pagare a 54.0.

il consumatore 1 acquista dall'impresa 26 al prezzo di 50.0
risparmiando rispetto a quanto era disposto a spendere 0.0
La sua disponibilità a pagare, scende al prezzo di acquisto ridotto di
1(new $p=49.0$).

Il consumatore 6 non compra dall'impresa 58 perchè il prezzo (50.0) è troppo caro!
il consumatore non voleva spendere più di 49.0
Sconsolato, aumenta la sua disponibilità a pagare a 49.0.

il consumatore 5 acquista dall'impresa 16 al prezzo di 53.0 risparmiando rispetto a quanto era disposto a spendere 1.0
La sua disponibilità a pagare, scende al prezzo di acquisto ridotto di 1(new p=52.0).

il consumatore 5 acquista dall'impresa 18 al prezzo di 50.0 risparmiando rispetto a quanto era disposto a spendere 2.0
La sua disponibilità a pagare, scende al prezzo di acquisto ridotto di 1(new p=49.0).

Il consumatore 6 non compra dall'impresa 66 perchè il prezzo (55.0) è troppo caro!
il consumatore non voleva spendere più di 49.0
Sconsolato, aumenta la sua disponibilità a pagare a 54.0.

Il consumatore 4 non compra dall'impresa 2 perchè il prezzo (48.0) è troppo caro!
il consumatore non voleva spendere più di 45.0
Sconsolato, aumenta la sua disponibilità a pagare a 47.0.

Il consumatore 1 non compra dall'impresa 23 perchè il prezzo (51.0) è troppo caro!
il consumatore non voleva spendere più di 49.0
Sconsolato, aumenta la sua disponibilità a pagare a 50.0.

il consumatore 6 acquista dall'impresa 40 al prezzo di 46.0 risparmiando rispetto a quanto era disposto a spendere 8.0
La sua disponibilità a pagare, scende al prezzo di acquisto ridotto di 1(new p=45.0).

Il consumatore 4 non compra dall'impresa 51 perchè il prezzo (53.0) è troppo caro!
il consumatore non voleva spendere più di 47.0
Sconsolato, aumenta la sua disponibilità a pagare a 52.0.

Il consumatore 4 non compra dall'impresa 53 perchè il prezzo (55.0) è troppo caro!
il consumatore non voleva spendere più di 52.0
Sconsolato, aumenta la sua disponibilità a pagare a 54.0.

Il consumatore 5 non compra dall'impresa 34 perchè il prezzo (51.0) è troppo caro!
il consumatore non voleva spendere più di 49.0
Sconsolato, aumenta la sua disponibilità a pagare a 50.0.

Il consumatore 1 non compra dall'impresa 5 perchè il prezzo (53.0) è troppo caro!
il consumatore non voleva spendere più di 50.0
Sconsolato, aumenta la sua disponibilità a pagare a 52.0.

Il consumatore 5 non compra dall'impresa 34 perchè il prezzo (51.0) è troppo caro!
il consumatore non voleva spendere più di 50.0
Sconsolato, aumenta la sua disponibilità a pagare a 50.0.

Il consumatore 5 non compra dall'impresa 25 perchè il prezzo (51.0) è troppo caro!
il consumatore non voleva spendere più di 50.0
Sconsolato, aumenta la sua disponibilità a pagare a 50.0.

il consumatore 4 acquista dall'impresa 49 al prezzo di 50.0
risparmiando rispetto a quanto era disposto a spendere 4.0
La sua disponibilità a pagare, scende al prezzo di acquisto ridotto di 1(new $p=49.0$).

il consumatore 2 acquista dall'impresa 59 al prezzo di 52.0
risparmiando rispetto a quanto era disposto a spendere 1.0
La sua disponibilità a pagare, scende al prezzo di acquisto ridotto di 1(new $p=51.0$).

il consumatore 5 acquista dall'impresa 39 al prezzo di 46.0
risparmiando rispetto a quanto era disposto a spendere 4.0
La sua disponibilità a pagare, scende al prezzo di acquisto ridotto di 1(new $p=45.0$).

Il consumatore 6 non compra dall'impresa 43 perchè il prezzo (47.0) è troppo caro!
il consumatore non voleva spendere più di 45.0
Sconsolato, aumenta la sua disponibilità a pagare a 46.0.

il consumatore 1 acquista dall'impresa 44 al prezzo di 51.0
risparmiando rispetto a quanto era disposto a spendere 1.0
La sua disponibilità a pagare, scende al prezzo di acquisto ridotto di 1(new p=50.0).

Il consumatore 6 non compra dall'impresa 51 perchè il prezzo (53.0) è troppo caro!
il consumatore non voleva spendere più di 46.0
Sconsolato, aumenta la sua disponibilità a pagare a 52.0.

Il consumatore 2 non compra dall'impresa 61 perchè il prezzo (54.0) è troppo caro!
il consumatore non voleva spendere più di 51.0
Sconsolato, aumenta la sua disponibilità a pagare a 53.0.

il consumatore 6 acquista dall'impresa 62 al prezzo di 51.0
risparmiando rispetto a quanto era disposto a spendere 1.0
La sua disponibilità a pagare, scende al prezzo di acquisto ridotto di 1(new p=50.0).

Il consumatore 5 non compra dall'impresa 41 perchè il prezzo (49.0) è troppo caro!
il consumatore non voleva spendere più di 45.0
Sconsolato, aumenta la sua disponibilità a pagare a 48.0.

il consumatore 2 acquista dall'impresa 65 al prezzo di 53.0
risparmiando rispetto a quanto era disposto a spendere 0.0
La sua disponibilità a pagare, scende al prezzo di acquisto ridotto di 1(new p=52.0).

il consumatore 2 acquista dall'impresa 60 al prezzo di 50.0
risparmiando rispetto a quanto era disposto a spendere 2.0
La sua disponibilità a pagare, scende al prezzo di acquisto ridotto di 1(new p=49.0).

il consumatore 1 acquista dall'impresa 50 al prezzo di 49.0

risparmiando rispetto a quanto era disposto a spendere 1.0
La sua disponibilità a pagare, scende al prezzo di acquisto ridotto di 1 (new $p=48.0$).

il consumatore 3 acquista dall'impresa 17 al prezzo di 48.0
risparmiando rispetto a quanto era disposto a spendere 3.0
La sua disponibilità a pagare, scende al prezzo di acquisto ridotto di 1 (new $p=47.0$).

Il consumatore 6 non compra dall'impresa 37 perchè il prezzo (53.0) è troppo caro!
il consumatore non voleva spendere più di 50.0
Sconsolato, aumenta la sua disponibilità a pagare a 52.0.

Il consumatore 1 non compra dall'impresa 61 perchè il prezzo (54.0) è troppo caro!
il consumatore non voleva spendere più di 48.0
Sconsolato, aumenta la sua disponibilità a pagare a 53.0.

Il consumatore 5 non compra dall'impresa 22 perchè il prezzo (51.0) è troppo caro!
il consumatore non voleva spendere più di 48.0
Sconsolato, aumenta la sua disponibilità a pagare a 50.0.

Il consumatore 1 non compra dall'impresa 61 perchè il prezzo (54.0) è troppo caro!
il consumatore non voleva spendere più di 53.0
Sconsolato, aumenta la sua disponibilità a pagare a 53.0.

Il consumatore 3 non compra dall'impresa 17 perchè il prezzo (48.0) è troppo caro!
il consumatore non voleva spendere più di 47.0
Sconsolato, aumenta la sua disponibilità a pagare a 47.0.

Il consumatore 4 non compra dall'impresa 46 perchè il prezzo (51.0) è troppo caro!
il consumatore non voleva spendere più di 49.0
Sconsolato, aumenta la sua disponibilità a pagare a 50.0.

il consumatore 6 acquista dall'impresa 43 al prezzo di 47.0
risparmiando rispetto a quanto era disposto a spendere 5.0

La sua disponibilità a pagare, scende al prezzo di acquisto ridotto di 1(new p=46.0).

Il consumatore 4 non compra dall'impresa 45 perchè il prezzo (54.0) è troppo caro!
il consumatore non voleva spendere più di 50.0
Sconsolato, aumenta la sua disponibilità a pagare a 53.0.

il consumatore 5 acquista dall'impresa 33 al prezzo di 47.0 risparmiando rispetto a quanto era disposto a spendere 3.0
La sua disponibilità a pagare, scende al prezzo di acquisto ridotto di 1(new p=46.0).

Il consumatore 3 non compra dall'impresa 24 perchè il prezzo (50.0) è troppo caro!
il consumatore non voleva spendere più di 47.0
Sconsolato, aumenta la sua disponibilità a pagare a 49.0.

Il consumatore 3 non compra dall'impresa 20 perchè il prezzo (52.0) è troppo caro!
il consumatore non voleva spendere più di 49.0
Sconsolato, aumenta la sua disponibilità a pagare a 51.0.

il consumatore 4 acquista dall'impresa 39 al prezzo di 46.0 risparmiando rispetto a quanto era disposto a spendere 7.0
La sua disponibilità a pagare, scende al prezzo di acquisto ridotto di 1(new p=45.0).

Il consumatore 5 non compra dall'impresa 41 perchè il prezzo (49.0) è troppo caro!
il consumatore non voleva spendere più di 46.0
Sconsolato, aumenta la sua disponibilità a pagare a 48.0.

Il consumatore 3 non compra dall'impresa 32 perchè il prezzo (52.0) è troppo caro!
il consumatore non voleva spendere più di 51.0
Sconsolato, aumenta la sua disponibilità a pagare a 51.0.

The simulation ended after 800 periods.
Press QUIT when ready.