An agent-based serious game for entrepreneurship

Mario Allegra¹, Giovanni Fulantelli¹, Manuel Gentile¹, Dario La Guardia¹, Davide Taibi¹, Gianluca Zangara²

(1) Italian National Research Council
Institute for Educational Technologies
Via Ugo La Malfa 153, 90146 Palermo, ITALY
E-mail: {allegra, fulantelli, gentile, laguardia, taibi}@itd.cnr.it

(2) Italian National Research Council
Institute of Biomedicine and Molecular Immunology "A. Monroy"
Via Ugo La Malfa 153, 90146 Palermo, ITALY
E-mail: zangara@ibim.cnr.it

Abstract

Serious games can create models of the real world, allowing students to enter lifelike environments and conditions which they would otherwise be unable to access, to gain experience of complex situations, reacting to specific and dynamic input they receive as they are playing. In this kind of environment they can assume a role, test their abilities and acquire new competences.

When designing a serious game, one of the first questions to ask is: “How realistic does the model have to be?”. A high level of reality is not always the best choice for learning. To reproduce complex situations may not be appropriate for a beginner, so it may be better to start with a level at which the user interacts with only a few of the components, adding others later. Besides, reality can refer to a variety of aspects: people, activities, objects, skills and places. A simplified model can be more effective for learning, if it focuses on specific content and/or skills at any one time. After the mastering of specific contents/skills by the user, the model can introduce a higher level of reality and integrate several components. For these reasons the system presented in this paper is based on a step-by-step approach which increases in complexity, with the aim of simulating an enterprise environment that allows students to learn business dynamics.

Keywords: Entrepreneurial mindset, Multi Agent System, Serious game

Introduction

Problems related to the employment perspectives of young people have always been taken into consideration by the leading economies. The cause of these problems is often attributed to a lack of entrepreneurship skills among young people who, at the end of their studies, have to cope with their first work experiences.

From this point of view, it is important to provide young people with methodologies and tools to stimulate an entrepreneurial mindset. In particular, our project aims to help students in learning the dynamics at the basis of enterprise environments.

Only some specialized schools include subjects related to the enterprise environment in their curricula, but simulations or analysis of real-world contexts that can help students in practical situations, are not often considered.
In this paper we present an agent based serious game that we are going to develop; it models an entrepreneurship environment using intelligent software agents. The game will be integrated in a training path involving enterprises, schools and universities.

Our system is based on a step-by-step approach which increases in complexity, with the aim of simulating an enterprise environment that allows students to learn business dynamics.

Many studies (McDowell et al, 2006; Pannese et al, 2007) argue that this type of simulation environment, in which students learn in contexts similar to the real world is useful and effective for dealing with similar situations that may occur in reality.

The intelligent software agent approach is useful to model a complex environment such as that of an enterprise. Every agent can be modelled with its own behaviour, that may differ in its complexity and in adherence to reality. The interaction of the single agents produces the evolution of the whole system.

We have designed two types of agents: autonomous and semi-autonomous. Autonomous agent behaviour is established at design time and cannot be directly modified by the user, while semi-autonomous agent behaviour can be controlled at game-time according to player choices.

The game can be accessed via a GUI integrated in Facebook as a web-browser game; it is aimed at developing students' entrepreneurial skills and to increase awareness of their own abilities.

**State of the art**

The field of simulation gaming research has been studied in depth by numerous researchers in the last few years, and special consideration has been given to the economic sector. The use of business strategy games which aim to develop entrepreneurial skills is well documented in the literature (Schreier and Komives, 1976) and it has led to the development of very sophisticated systems (Faria, 2001).

The design of games and simulation environments requires careful attention to the definition of rules and functionalities that the “decision engine” has to follow in order to generate real scenarios according to specific learning objectives. A typical approach in the design of the “decision engine” is to use a centralized architecture based on numerical techniques or rules systems. Using this approach, designers have to deal with the problem of complexity management. The studies carried out in (Cannon, 1995) affirm that complexity is a fundamental characteristic in this kind of learning environments if it is to simulate reality and be effective.

On the other hand, high complexity involves other problems such as a lack of transparency in the relationships of cause and effect in the actions undertaken by the players during the game (Cannon et al, 2009). Moreover, the more complex the system is, the greater will be the delay in obtaining feedback regarding the actions taken by a player (Dobson et al, 2004).

The result is an intrinsic difficulty for players to assess the quality of their actions, and consequently poor support of the processes of reinforcement learning that these tools are aimed to activate.

From a technological point of view, the evolution of artificial intelligence techniques and the consequent development of new programming paradigms, such as agent programming, are opening up new scenarios in the development of business strategy games.

The studies carried out (Dobson et al, 2004) state that agent technologies are suitable for the design and development of this kind of learning environment, in order to overcome the problems of the current platforms. First of all, this paradigm allows a designer to model the system in terms of the entities that operate within it, providing a natural description of the system.

This bottom-up approach facilitates the design and the development of a more flexible system; users can add new agents or tune their complexity.
For example, it is possible to construct a simulation with increasing levels of difficulty, thus adapting the environment to a student’s abilities in accordance with the curiosity-gap theory (Loewenstein, 1994).

Moreover, it is possible to model the system even when the level of description of agent’s behaviours is not defined beforehand.

Finally, agent based systems capture emergent behaviours, even simple interactions between agents can generate complex behaviour patterns.

**System Architecture**

The game is based on a business simulation environment, in which the main actors are enterprises, suppliers and customers. We use the agent-based simulation (ABS) to validate the design model and to test the automatic behaviours of the agents. In the game we identify two kinds of agents: autonomous and semi-autonomous agents. Customers and suppliers are examples of autonomous agents. Instead, semi-autonomous agents have some responsibilities in the management of the company, and include: production manager purchasing manager, inventory manager.

Semi-autonomous agents play a dual role within the game. At starting levels, these agents are autonomous; during the game, as the user gradually reaches more advanced levels, he should be able to cope with the events that may occur in each sector; and finally they can replace the expert agents in taking critical decisions.

At this point the semi-autonomous agents will monitor user activities, allowing assessment of user behaviour. In fact the user can compare his decisions with the optimal behaviour “suggested” by the agent.

The behaviour of an agent is encapsulated in the code that defines the agent, in this way relations between different software modules are reduced. The agents can change their behaviour without requiring radical changes in the whole system.

Agent based architecture is distributed; agents can be resident on different servers, thus balancing hardware resources. A well defined protocol is used to manage communication between agents “living” in different servers. In this way, agents can communicate with each other, independently of their localization.

In the following sections the main scenarios of the game are described.

**Production Management**

The production department is the area of the enterprise that transforms raw materials into products. In our model it is characterized by the number of employees and the machinery, these elements characterize the production capacity. On the one hand, increasing the number of employees and machinery increases the production capacity of the company; but at the same time this entails higher costs for maintenance and repairing machinery, and salary costs for employees.

Therefore the production manager agent has to balance these parameters, depending on the available resources and market requests. The availability of more machinery and employees makes it possible to diversify the number of products and increase the production level. It is necessary to have skills in scheduling techniques to manage all these parameters in order to maximize profits and meet the objective of obtaining the minimum delay in orders, reducing delivery delays that can cause/lead to loss of money. The manufacturing process should also take account of the resources available in the inventory management, thus influencing product supply, too.

Complex dynamics also affect the employees. The employee is an autonomous entity that evolves over time. Our environment intends to model the skills of employees, their qualifications, productivity and training level. Employee skills increase with experience over time or through
participation in training courses, while their productivity in general varies with the age of the employee.

The model we are designing, also takes into consideration an employee’s request for a higher salary as he gains qualifications or experience. If the user doesn’t accept the request, the employee can decide to accept a job offer from another company.

**Purchasing materials**

One of the key sectors in the management of a company is the purchasing area. This area controls the sourcing of materials necessary for production. Management of this sector is in the hands of the purchasing agent who is responsible for finding suppliers and managing the business relationships with them. The purchasing agent responds to needs identified by the inventory manager.

The search for suppliers may be carried out using the service directory facilitator (DF) provided by the JADE agent platform (Bellifemine, and Rimassa, 2001). The DF agent does not explore the entire market of suppliers, but rather the subset of providers who have decided to publicize their activities through the DF.

Moreover, using this mechanism, the purchasing agent does not have any parameters for evaluating and selecting the suppliers.

Instead, during the game the supplier selection is made using the model suggested by (Pi and Low, 2006) that allows an assessment of the supplier on the basis of different parameters (quality, price, on-time delivery). These parameters may be derived from personal experience or from the information obtained by companies belonging to the same social network.

The purchasing agent implements the FIPA-Iterated Contract Net Protocol (shown in Figure 1) to negotiate with the supplier agents.

The supplier agent may follow a policy of customer loyalty; for example, when a supplier agent does not receive orders from a loyal customer, it could adopt a policy of discounts for that company.

**Customer Agent**

Customer agents represent the entities that buy the products. In our model customers can simulate two types of behaviour related to b2c or b2b relationships.

In the b2c scenario the agent simulates a habitual/ordinary customer who buys the product in small quantities at the price fixed by the user, instead in the b2b scenario, agents make an order for a product which involves contracting the price, the delivery and the quantity. In the latter case the user can decide to accept the order or refuse it, considering the resources available to his enterprise. The production of a huge quantity of products requires the employment of many machines and workers in order to respect the pre-established deadline.

![Figure 1. Iterated Contract Net Protocol (Fipa Specification)](image-url)
Users should be able to balance the employment of resources with the need to obtain the maximum profit from the order. Users can also consider sharing the order with other users. The reasons that may lead to this solution are for example:

- The user considers the order to be favourable but he does not have the required resources to produce that quantity of product.
- During the production phase the user realizes that he cannot deliver on time, so he involves other users in order to meet the deadline.

As happens on the best-known sites for online transactions, at the end of the transaction the agent gives feedback on the user. The parameters of the feedback are related to time and quantity. The customers will take into account the feedback received by the user to place new orders. The B2C scenario also takes different factors into consideration. In this scenario the user has to consider investment in advertising for his products. The choice of a customer agent will be influenced by the advertising.

Conclusions

In this article we have described a serious game whose purpose is to foster an entrepreneurial mindset in young people.

A key aspect of the model presented in this paper is the creation of an environment where users acquire competence in business dynamics and management concepts.

The approach presented in this paper focuses on modeling such a complex environment through the definition of simple software agent behaviors. This approach has a dual aim: from a modeling point of view to make the designed environment close to reality and from a pedagogical point of view to provide users with the necessary feedback to improve their learning paths.

References


