Management flight simulators:

Development of a multiplayer, asymmetric, network, ILE

Andrea M. Bassi * Tobias M. Lorenz
System Dynamics Master Students
University of Bergen, Norway

The importance of management flight simulators for learning has been already tested and documented. Single player simulation games are usually utilised, but a multiplayer simulation game adds direct competition to the existing problems (delays, nonlinearities and feedbacks). An asymmetric game also introduces bounded rationality and the dynamics of the information flow within the team. A network team game finally allows for the analysis of cooperation dynamics (by letting the users play against each other: against an unpredictable competitor, with no pre-defined strategy). This paper presents an asymmetric multiplayer network game that is considered to be easy to play and understand. The main advantage of the ILE here introduced is the facilitation of the analysis of: learning and decision making processes, cooperation and competition dynamics.

Introduction

Flight Simulators represent one effective way to provoke learning. Simulation and practice can help the player to understand how the dynamics of a company, due to feedbacks and delays, work. Facing the reality of a market, managers have to deal with people, with uncertainty and with some seemingly external factors affecting the environment. Most of time the misperception of feedback lead them to the creation of wrong mental models, affecting their decision making process. Conventional and traditional analysis (i.e. Resource-Based View, Balanced Score Card, Industry Framework Analysis), are static and don't take into consideration the feedback characterising the analysed system, i.e. the company. A dynamic management simulator can help the users understand which are the market

drivers and competitive leverages, learning how the system works while playing (Just-In-Time Open Learning, JITOL[1]).

"Boom and bust" has been analysed for companies in fast growing markets[2]. In that specific case, individual decisions led to amplification, phase lag and over capacity building, on the individual level. These effects were made even worse on the global level, i.e. the company.

An ILE (Interactive Learning Environment) allows people non familiar with the System Dynamics methodology to profit from the insights gained through modelling. A single user game can stimulate learning, provide hints for developing strategies, and give experience to the player. A symmetric multiplayer game has the advantage to let two, or more, players confront themselves. Once they learn how the structure of the system affects their decisions, they have to define a strategy. A very large number of possibilities arise, according to the number of variables they can manage, and the strategy will be the output of learning from the structure and from the action of the opponent player. Since a strategy consists of a well-coordinated set of policies[3], we expect the players to become policy designers, instead of decision makers. As stated by Jay W. Forrester: "People can make decisions without understanding why. Decisions then tend to be capricious and are dominated by short-term pressures. A decision-maker runs an organisation, but a policy-maker designs an organisation. The distinction is like that between an airplane pilot and an airplane designer. It is the challenge of the designer to create a system that can function as intended in the hands of the kinds of operators who will be available"[4].

An additional feature of games is the possibility to create different roles, which defines the structure as asymmetric. In this case there will be two or more teams competing, but the single player will have different tasks and instruments. The easiest example is represented by departments in a company: finance, marketing, production, human resource, etc. Every manager has different goals, information to rely on and constraints. In reality, the optimal result for a department can produce unexpected difficulties in operating for some other departments. A company need to be run harmonically, the departments should work together for the good of the company.

The aim of this work is to present a multiplayer asymmetric ILE. The system dynamics techniques presented are: dynamic modelling in the form of stock and flow as well as causal diagrams, simulation, and single user as well as multiple user (asymmetric) simulation-based learning

---

[1] Davidsen P.I. (1993) "System Dynamics as a Platform for Educational Software Production", in Berta et al.: "Proceeding of the IFIP WG3.4".
[2] Paich M. and Sterman J.D., "Boom, Bust, and Failures to Learn in Experimental Markets", Management Science, 39, 12 (1993), 1439-1458.
[3] Davidsen P.I. (1995) "Educational Features of the System Dynamics Approach to Modelling and Simulation", Journal of Structural Learning.
[4] Forrester J.W., "System Dynamics–Future Opportunities", in Legasto A.A.Jr. Et al.: System Dynamics, North-Holland, 1980.

environments. Tools to analyse the learning process and how local decisions affect the total performance of the company will be presented.

The game here presented starts from the "boom and bust" simulation game created by Paich and Sterman, but include some different sectors and features. The capital stock is not considered, while it has been introduced the labour chain; active competition has been included (through the addition of two player playing against the first team); some well known SD model have been used, in order to keep the basic SD structure for learning purposes. It is also possible to analyse the expected labour force value, in order to test the learning process of the players (a comparison between learning mode and arcade mode could provide useful hints).

Model Structure

The model has been created with the intent to represent a simplified real market. The structure of the model accounts for the creation of a market demand and a specific demand according to the attractiveness of each product. Production and labour chain are introduced as well as pricing and financial allocation process.

The total structure consists of seven modules, all of them are well known and have been tested and validated: Bass Diffusion Model, Polya Process, Manufacturing Supply Chain, and Labour Supply Chain are the most famous structures used to create the flight simulator presented here. Each of them has been modified to allow interactions between the sectors and to let the user play with decision variables.

Most variables of the model are characterised by the presence of two subscripts: Company A and B. While in the Asymmetric Multiplayer Game the equations are the same for both companies, in the Single Player Learning Game there are various differences between simulated and directly managed company.
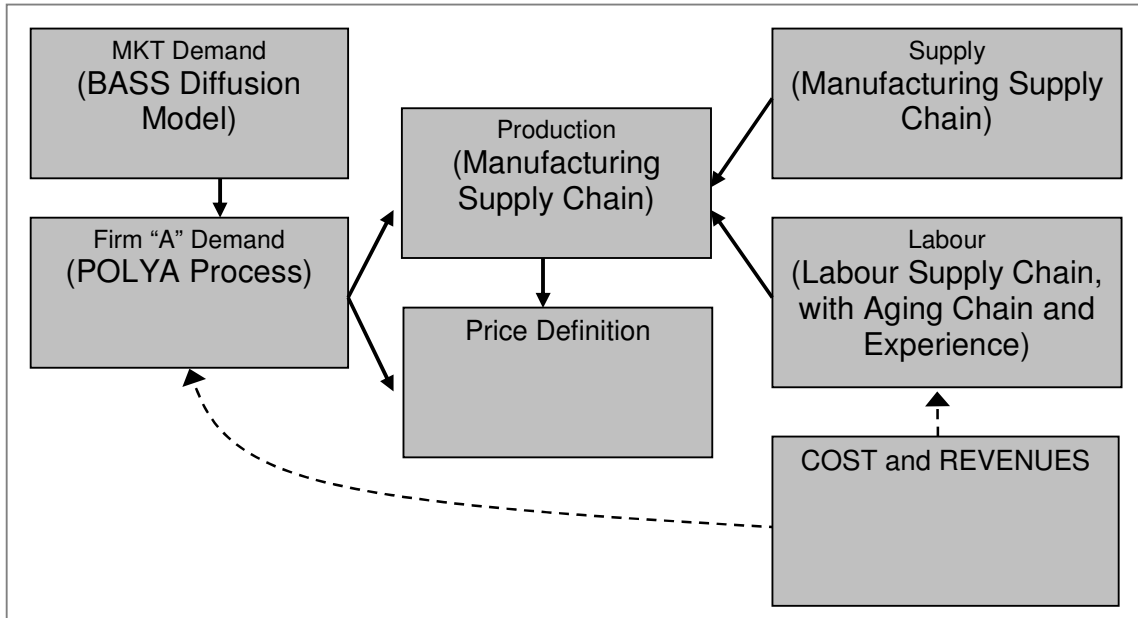
Figure 1: The complete DBM Flight Simulator Model, modules and interactions.

In the next section a brief analysis of the seven sectors of the model is presented. Since the basic structures present in various sectors are well known and have been already largely documented, only the utilisation of the models with respect to the simulator, and modifications, will be explained. A sketch of each sector and the relative Causal Loop Diagram will be also shown.
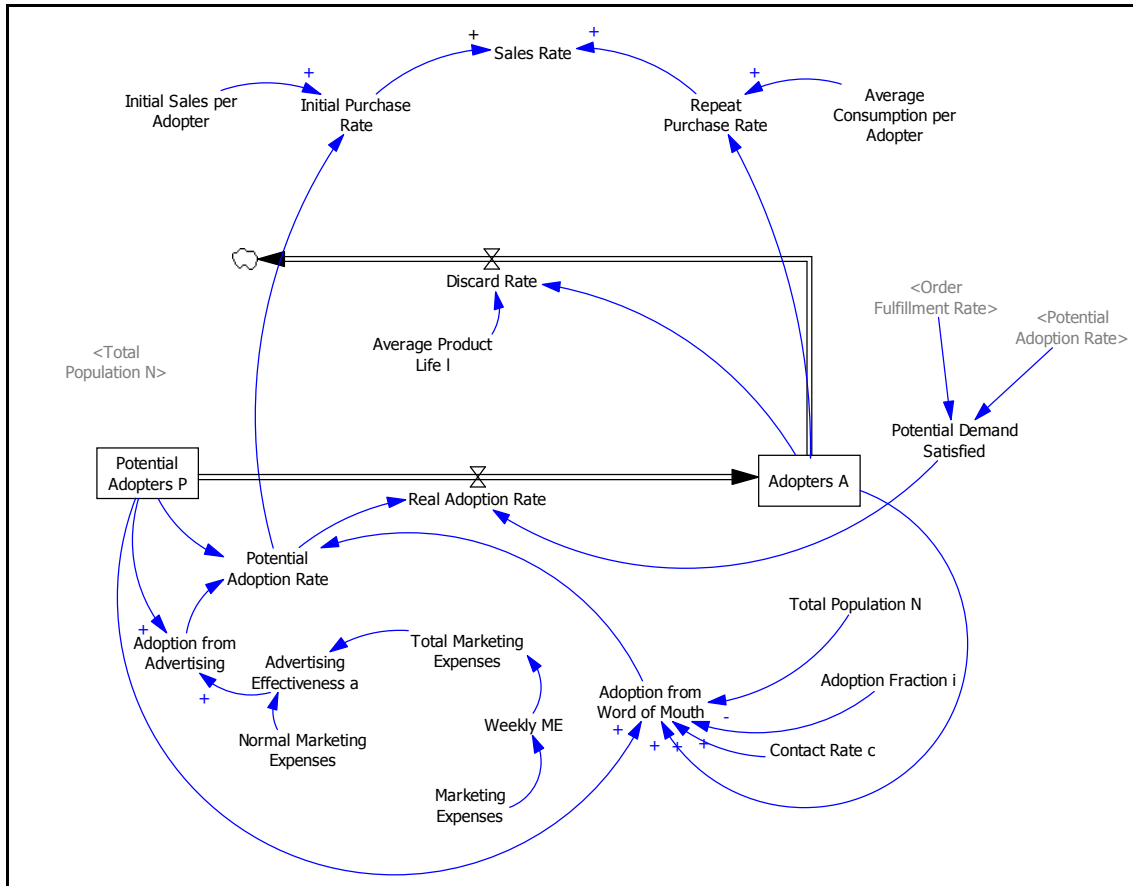
Market Demand (Bass Diffusion Model)

Figure 2: Stock and Flow Diagram of the Market Demand Sector; basic structure: Bass Diffusion Model.

In order to define the market demand, Marketing and Word of mouth effect are taken into account. The Bass Diffusion Model[5] is used here to determine the aggregate demand and the adoption dynamics. A potential adoption rate is defined by advertising and word of mouth, while the real adoption rate consists of potential demand satisfied, multiplied by the potential adoption rate. While the initial purchase rate is based on intended adopters (potential adoption rate), the re-purchase rate is defined by the actual adopters.

---

[5] The Bass model is a well known model for the diffusion of innovations. It was developed as a tool for forecasting the sales of new products and has two explanations for the spread of a new product: the word-of-mouth effect and the effect of marketing. In the game it is used to generate the potential demand.
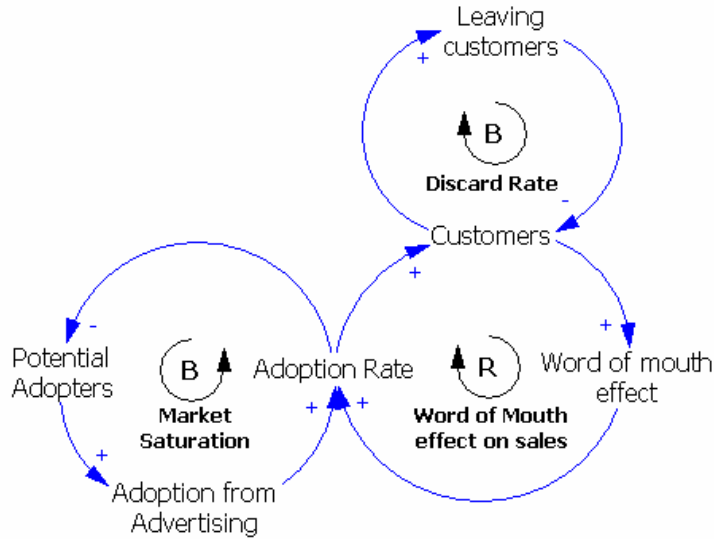
Figure 3: Simplified CLD of the Market Demand Sector.
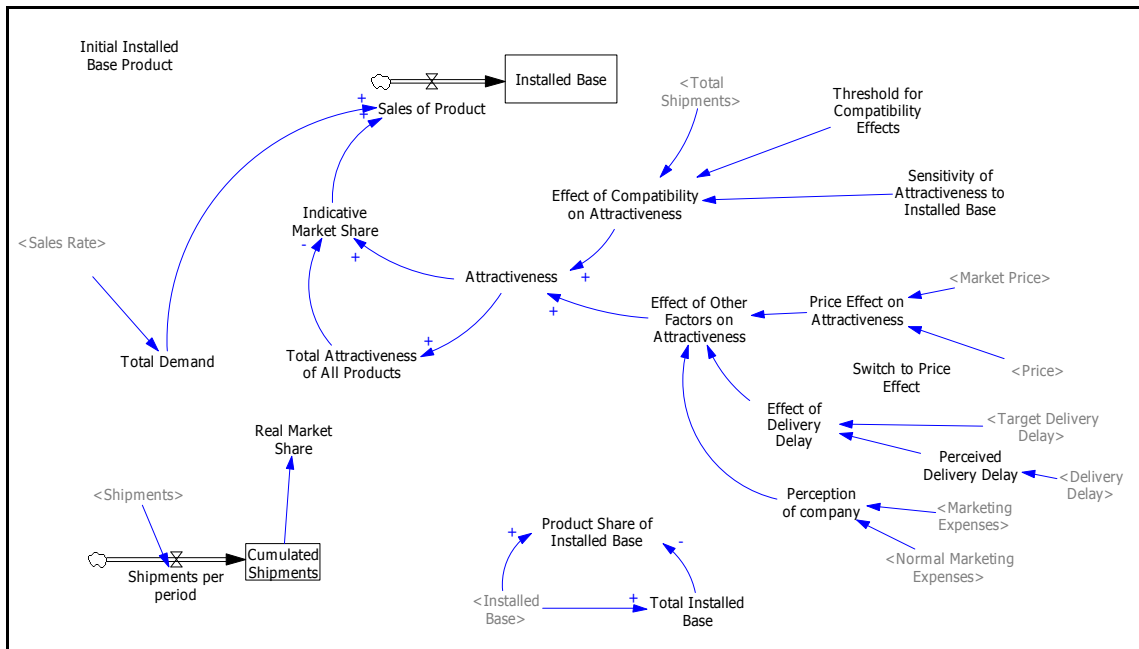
Market Share Definition (Polya Process)

Figure 4: Stock and Flow Diagram of the Market Share Definition Sector; basic structure: Polya Process.

The Polya Process is the second well known SD model used as a part of the structure presented here. The function of the Polya Process is the determination of the indicated market share of each company. The attractiveness of each product can be defined respectively and then an indicative market share (and consequently the demand) for both companies can be obtained. The factors affecting the attractiveness are: installed base, price, delivery delay, marketing expenses. They can easily be seen

as: compatibility, quality/price ratio, service, image of the product (perception of the company). The strength of the various factors can be varied easily according to the characteristics of the market.
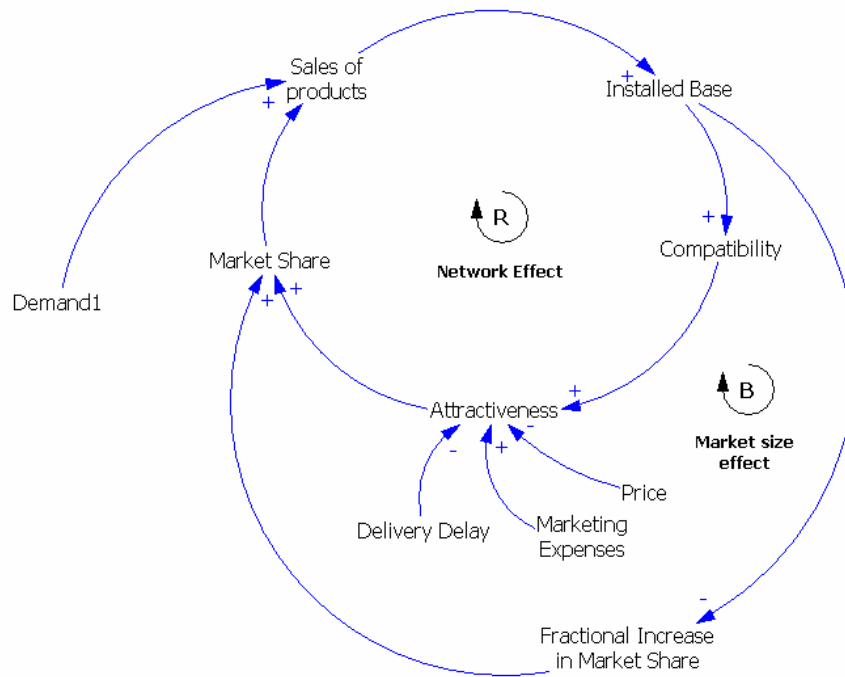

Figure 5: Simplified CLD of the Market Share Definition Sector.

The Polya Process is an easily accessible example of path dependence and is used to model the network effects in the game and thereby compute the market share. It is composed by two feedback loops, a positive and a negative one. The bigger the initial installed base of one of the two companies, the higher is its attractiveness for new customers. On the other hand, the bigger the total number of adopters in the market, the less is the influence of an additional customer. This effect stresses the importance of gaining a competitive advantage early in the game, in order to make the respective product the most compatible (network effect).

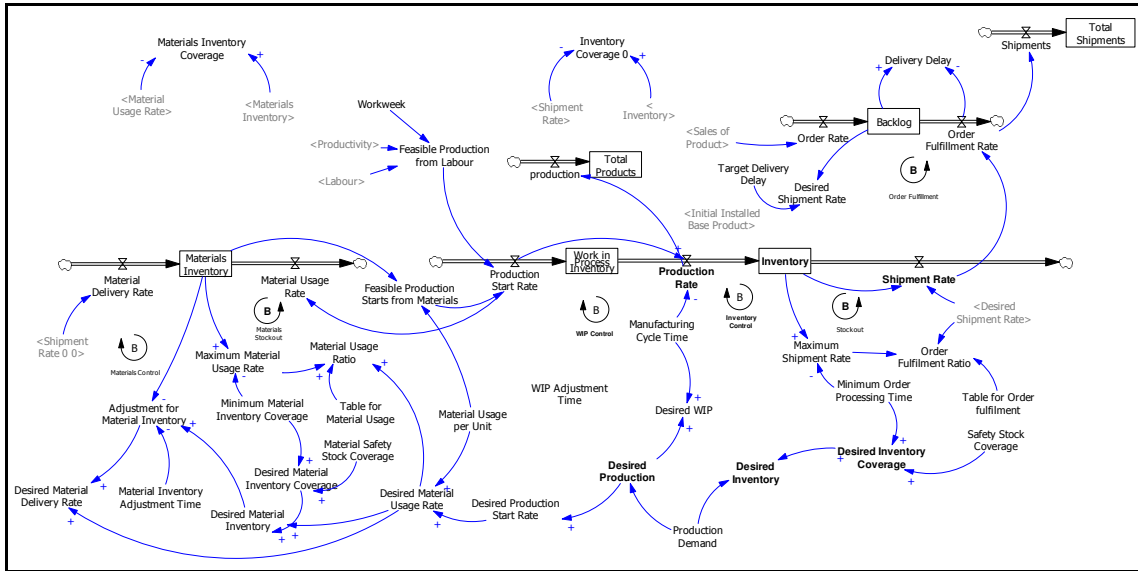## Production and Inventory (Production chain)

Figure 6: Stock and Flow Diagram of the Production and Inventory Sector; basic structure: Manufacturing Supply Chain.

The production chain accounts for material and finite inventory as well as production and shipments. The structure is represented by the Manufacturing Supply Chain developed by Sterman (….2000). Some modifications have been made in order to let the player define the production demand. The production start rate has been also modified to include the feasible production from labour, defined by employees, their productivity and the workweek.

Supplier Production and Inventory (Supply chain)
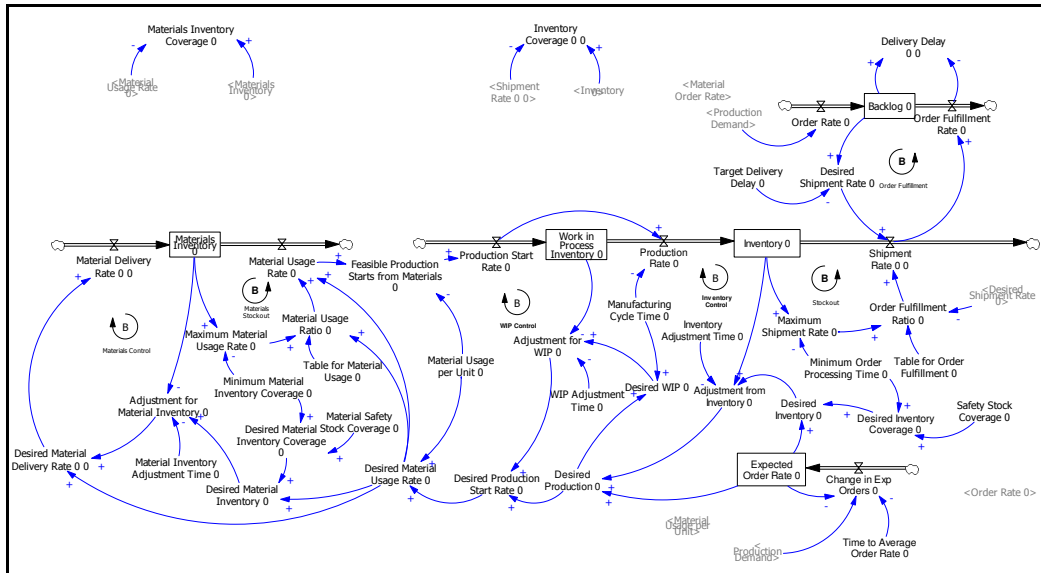

Figure 7: Stock and Flow Diagram of the Supplier Production and Inventory Sector; basic structure: Manufacturing Supply Chain.

The Supply chain is exactly the one created by John Sterman[6], and utilises fixed delays and adjustment mechanisms for material, work in process and finite products inventory. In this sector there aren't decision variables and the supply chain is used to explain how the flow of material works and how the decision variable "production demand" affects the delivery delay of the products.
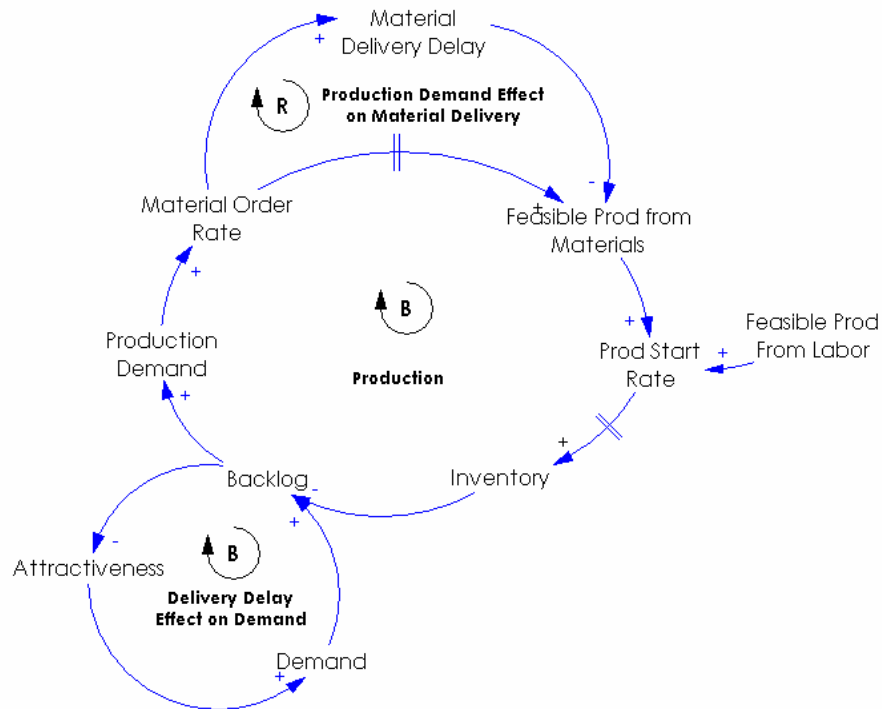

Figure 8: Simplified CLD of the Production and Inventory Sector.

Here is presented the CLD of the production as it would be played by the computer. The production demand increases the material necessary for production with a delay. On the other hand a high production demand increases the material delivery delay, so that the raw material is received later than expected, fact that could increase the production demand even more. After the production has started and the backlog decreased, the production demand also decreases. In addition a high backlog decreases the attractiveness of the product.

---

[6] Sterman J. D. (2000), Business Dynamics, Systems Thinking and Modeling for a Complex World, New York, Irwin/ McGraw-Hill.
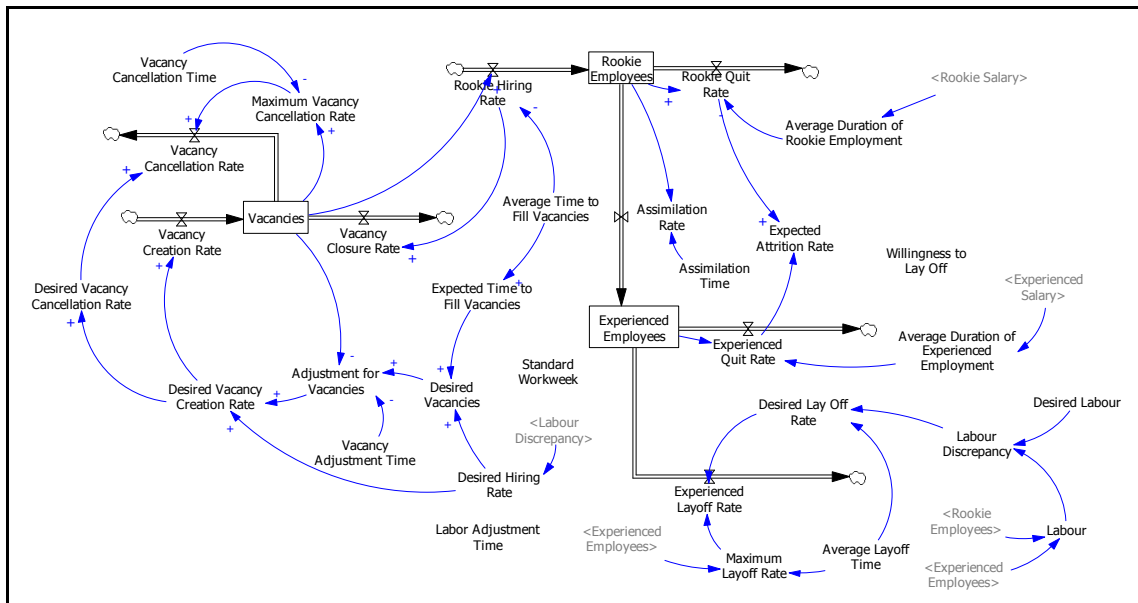
Labour (Labour chain and experience)

Figure 9: Stock and Flow Diagram of the Labour Sector; basic structure: Labour Supply Chain.

The labour sector includes the vacancy and hiring process, as well as the learning process from rookie to experienced employee. Desired labour is the gaming variable affecting the sector, modifying directly hiring and layoff rate. Part of the structure is represented by the labour chain[7] but here as before, an additional part of the structure has been introduced: the small aging chain that links rookies and experienced employees.
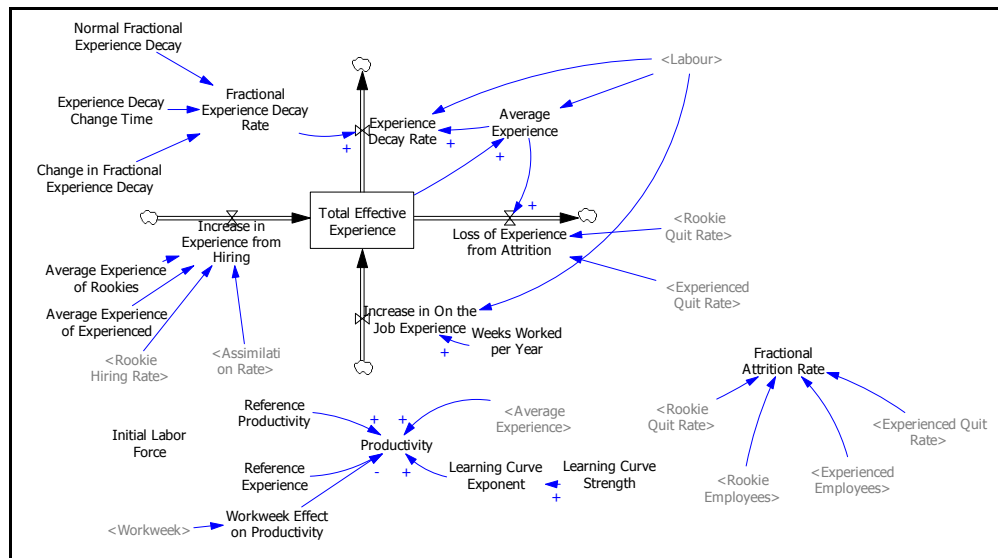

Figure 10: Stock and Flow Diagram of the Labour Sector; experience and productivity.

---

[7] Sterman J. D. (2000), Business Dynamics, Systems Thinking and Modeling for a Complex World, New York, Irwin/ McGraw-Hill.

Experience and Productivity are here utilised to analyse the effect of hiring and layoff policies and to determine the feasible production from labour. The rookies have less experience than the experienced employees, factor that affect the total productivity of the workforce. The total Effective Experience is used to calculate the average experience, factor of the Productivity variable. The workweek, decision variable for the Production Manager, is also considered as a negatively correlated input for the determination of the workforce Productivity.
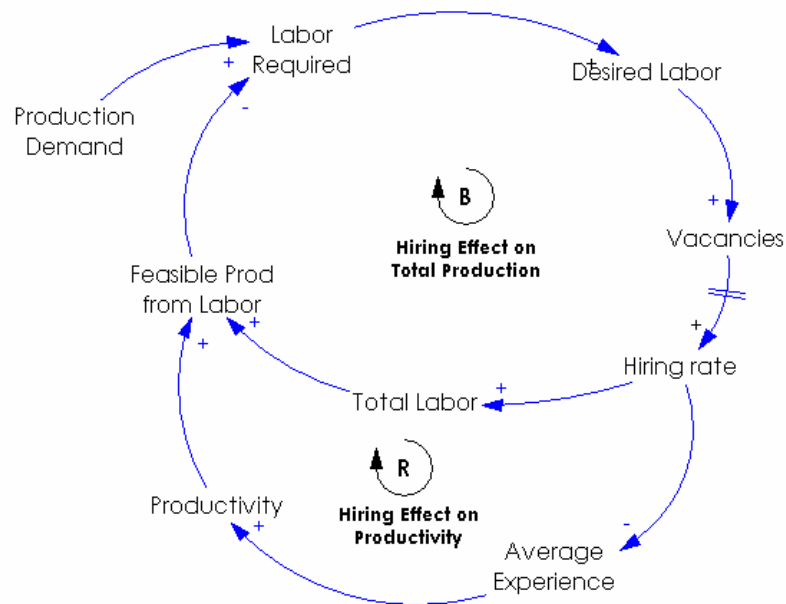

Figure 11: Simplified CLD of the Labour Sector.

This is the CLD of the Labour Sector as it would be played by the computer. The gap between the feasible production from labour and the production demand determines the labour required. This value should be identified by the user, which decides the desired labour. Vacancies opening and subsequent hiring or layoff follow the decision of the user. The higher the hiring rate, the higher will be the total experience, which through time increases the productivity. When hiring, total number of employees and productivity increase, the feasible production from labour increases, generating the reduction of labour required to satisfy the production demand.

Price definition
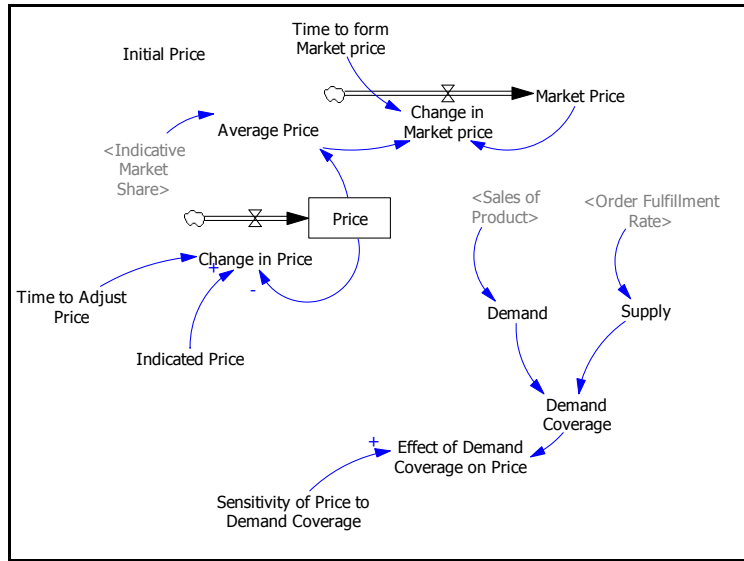
Figure 12: Stock and Flow Diagram of the Price Definition Sector.

The price sector includes the Market Price formulation defined as weighted average between the priced indicated by the CEO of the two companies. The indicated price, as a gaming variable, in the arcade mode is defined by the users, while in the learning game, as for the simulated company, is defined by the "effect of demand coverage on price" multiplied by the "market price". A time delay for the implementation of the price decision is assumed.

Cost and Revenues
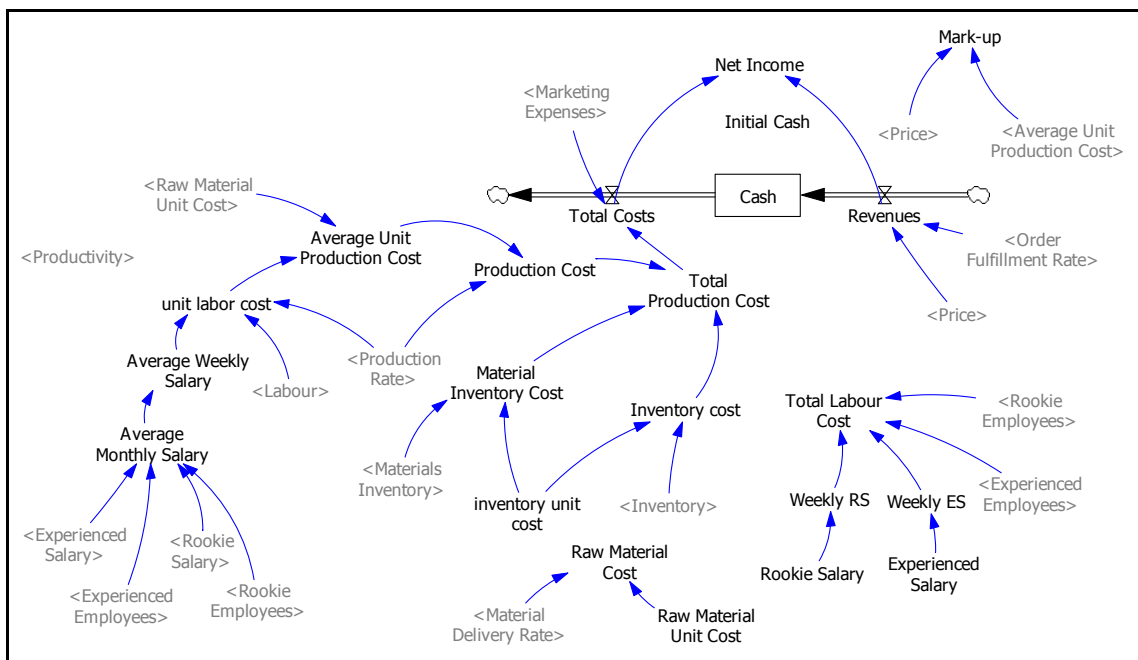
Figure 13: Stock and Flow Diagram of the Finance Sector.

The final sector is used to obtain all the necessary financial information about the company. All costs, including labour, production, raw material, material and finite product inventory storage, are considered. Cash and net income are obtained subtracting the costs from the revenues. Unit and total costs are calculated and displayed to the users while playing.

Sectors and interactions

Figure 14: The complete DBM Flight Simulator Model, modules and interactions with explicit subscripts.

In the next section the interactions between the sectors will be presented. Once the structure of the model is understood and the interactions between the modules have been analysed, the feedback

structure of the game will result explicit and more intuitive. As stated earlier, the model is supposed to reproduce simplified market dynamics as well as the firm environment, which includes consequences of the decisions taken.
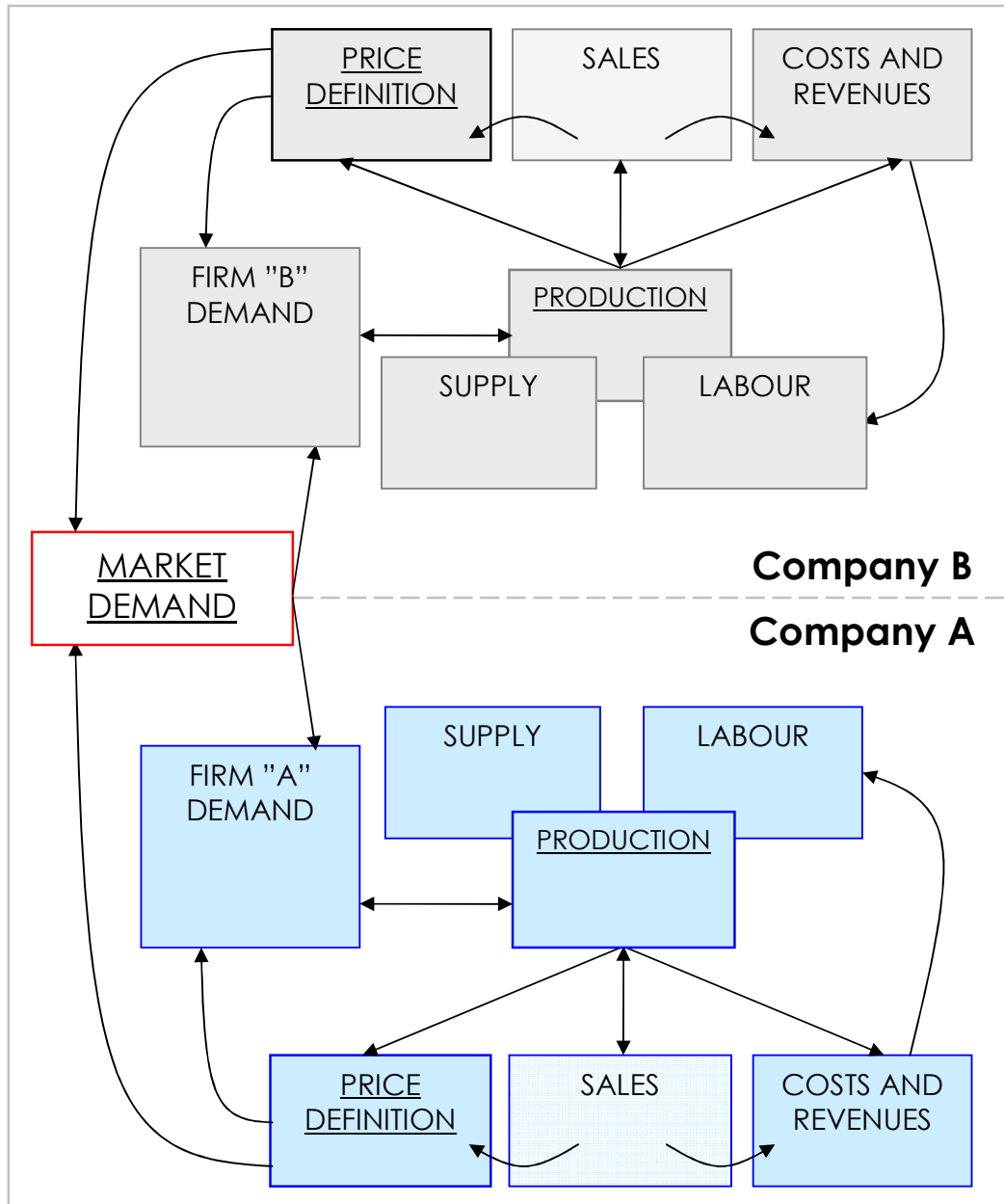
Market Demand (Bass Diffusion Model): this sector receives as an input the Marketing Expenses (gaming variable), which determine how fast the demand is going to grow according to the Advertising Effectiveness. High expenses in marketing will speed up the attention of the customers to the product making them willing to buy it sooner. The Word of Mouth effect will also be stronger, due to the fastest acquisition of the product by the population. The Order Fulfilment Rate (Production Sector) is also used as an input for the sector. Its value is necessary to calculate the Potential Demand Satisfied. This value is calculated as the minimum between the Order Fulfilment Rate and the Potential Adoption Rate, and affects the real adoption rate.

Firm Demand (Polya process): the market sales rate (Market Demand sector) is used as total demand, starting value to define the market share and the demand for the two companies. The Real Market Share is defined by the ratio between company's and total cumulated shipments. Price (Price Definition sector), Delivery Delay (Production sector) and Marketing Expenses (gaming variable), together with the cumulative installed base, define the attractiveness of the product. All of them are normalised over the value of the competitor or normal constant values.

Production (Manufacturing Supply Chain): the inputs of the sector are mainly related to shipments and production. The Sales of Product (Firm Demand sector) are the input for defining the backlog of orders, a value on which production demand and inventory management are usually based. As for the production, Labour and Productivity (Labour sector) determine the feasible production from labour that, compared with the feasible production from materials, defines the production start rate. The decision variable in the sector is the production demand, directly affecting the material orders.

Supply (Supply chain): this sector is naturally linked to the production sector, no other interactions to other modules have been introduced. The Production Demand, input from the Production sector, is equal to the material orders received by the supplier. The only difference existing between Production and Supply sectors is the presence of adjustments for the inventories, due to the absence of gaming variables directly defined by the players.

Labour (Labour Supply Chain): three decision variables are the only inputs of the Labour sector: Desired Labour, experienced and rookie salary. As for the Desired Labour, the value submitted will produce a discrepancy between actual and desired workforce, which will define the desired hiring and layoff rate. The hiring process involves the creation of vacancies, while the layoff is characterised by a long delay time. The salaries define the attrition rate, affecting the average duration time of the employment of both categories of employees. They also directly determine the total labour cost of the company and indirectly affect the productivity of the employees.

Price definition: there are four inputs of the sector, but only two of them are utilised in the multiplayer game. Indicative Market Share and Indicated Price are used to calculate the Market Price and to define the product price. Sales and Order Fulfilment Rate are used to define the "Demand Coverage", important variable for the definition of the simulated price only in the Learning mode.

Costs and Revenues: the total costs can be divided in labour cost, production cost, storage cost and marketing expenses. The methodology utilised here consists in the computation of the unit cost of production, including labour and raw material cost. The unit production cost is calculated as follow: the unit raw material cost is added to the average weekly labour cost divided by the production rate. The production cost is then defined by multiplying the unit production cost by the production rate. The total cost is calculated by adding marketing expenses, material inventory and finite inventory storage cost to the production cost. The numerous inputs of the sector come from Production, Labour and Price modules.

Game Structure

Figure 15: The DBM Flight Simulator Game, structure.

Architecture of the Game

In order to understand how the game structure works, a brief introduction to the file architecture of the game seems to be useful. In the next paragraph the structure of a multiplayer simulation game constructed with Venapp will be explained. The following one, on the other hand, will be focused on restrictions and facilitations that have been introduced to the structure in order to avoid common problems.

The file structure of the game has been created step by step, considering every action that the simulator should make in order to read and write data. Starting from the choice of the role, when one of the four players chooses a profile in the decision screen, a file ".mrk" (e.g. "productionmanager.mrk") is created in the main folder. If another player tries to choose the same role, and the ".mrk" file already exists, a message box will pop up indicating that this position has already been chosen. On the other hand, if the position is still available, the player will be forwarded to a screen, which waits until the supervisor "current.mrk" file is created. This file is created by the supervisor to signal that his simulation run has been copied to the main folder. At the same time the model and the graph sets are loaded. On the next screen the "current.vdf" file, the run file created by the supervisor and placed in the main folder, is copied into the players directory, after having cleared the old file. Then the run is simulated and it is also specified as a game. On clicking the "Submit" button during the game, several tasks are fulfilled:

1. the "current.mrk" file earlier created by the supervisor is being deleted;
2. a file ".gin", which contains the decisions taken in that period, is written;
3. the game is being ended;
4. the player is forwarded to a waiting screen;

The waiting screen waits until the creation of the file "current.mrk" is once again done by the supervisor and then fulfils the same tasks as in the beginning of the game: it clears its actual "current.vdf" file, copies the new one situated in the main folder, which is now simulated one game interval further, into his personal folder (e.g. "player1" folder), and then simulates and specifies the run as a game again.

The supervisor screen acts, as the simulator, in the background: it reads-in all the decision files from the main folder, copies its current run there and creates the file "current.mrk" to signal the players that they can copy the "current.vdf" to their personal folders again. This is being done by an IFTHENELSE construction. Once all the ".gin" files of the players, the decisions they have made, are available, the supervisor reads them in and simulates with the command GAMEON. The resulting ".vdf" (the file that has been manually created by the supervisor at the beginning of the game) file is copied as "current.vdf" to the main folder. All this process is being done automatically.

The architecture still contains one problem, which is, that the game is stuck if a player already submits a decision file before all the player roles have been chosen. In this case the supervisor has to manually forward the game.

Another approach thought about, was to implement the presented structure into one of the players or make even every player actually simulate instead of just copying the "current.vdf" file. Nevertheless those ideas were discarded as it was considered useful to leave the supervisor with some basic functionality, like moving one time step further without reading-in the decision files or to end the game if necessary.

Action	Supervisor	Player
Pick-up a role	Manually create ".vdf" simulation file, create "current.mrk" file, copy the ".vdf" file into the "current.vdf" file.	Create ".mrk" file.
Access to the game screen		Check for the supervisor's "current.mrk" file. If existing, clear old "current.vdf" file and copy the new "current.vdf" file from the main folder into the player folder.
Submit Decisions	Check existence of all the ".gin" files, read them, and simulate (GAMEON command). A ".vdf" file is generated by the simulation. Copy ".vdf" file into "current.vdf" in the main folder. Create a new "current.mrk" file.	Delete "current.mrk", create a ".gin" file and end game

Table 1: Architecture of the game, process of decision submission.

In order to make the game easy to setup, play and investigate, some restrictions and facilitations have been introduced:

— First of all a batch file "management.bat" has been created. In order to setup the game, this file should be called from the main folder on the server with the parameter "setup", e.g. "Z:\...\managementa\management.bat setup". This command will delete the current game folders and files and replace them with new copies of all the necessary files – if available – out of the folder Z:\...\managementa. The setup command will create the necessary folder architecture for the game, i.e. eight folders named "player1" to "player8". To start the game, it is necessary to run the "management.bat" file with the player as extension, e.g. "Z:\...\managementa\management.bat player1". This will start Venapp out of the folder "player1" which has been created and supplied with the necessary files by the setup.

— The total number of players has been set at eight, but it can be easily extended. In the actual configuration the active players can range from one to five, concerning one simulation and both Single Player Learning Game and Multi-user Asymmetric Game. The number of Observers can range from zero to infinitive. The number of active player can easily be increased according to

their roles: the actual configuration accounts for only two different roles, but additional ones can be added, increasing the number of the decision variables and defining responsibilities.

— Both game modes don't need a supervisor to be played. In the arcade game only the specific name of the simulation must be indicated and the supervisor screen must be loaded, but it is not necessary the active presence of a moderator.

— In order to let the moderator focus on the players needs, once the game is started, the supervisor screen is automatically updating every five seconds, as well as the observer screen. The supervisor, whenever needed, can also forward and end the game.

— Every run is stored in the folder of the player and can be loaded and analysed at the end of the game. The simulation can be displayed with the ILE in the Analysis screen or can be investigated with Vensim once the model is loaded.

— In the multiplayer game, once a role has been chosen, it is not possible to choose it a second time. This feature allows the user to freely choose the profile they like and to define easily teams and roles.

— Still in the arcade game, the players start the game at the same time, when the supervisor has created the "simulation file", this feature allow the game to be fluent, avoiding long delays for waiting people to play.

— When the game is over, the "results" screen will appear, in order to allow the players to check their personal performance, the team performance and the overall results.

Features of the Game

Once the application is opened, the user can read information about the market and the game, find instructions, investigate the structure of the game and decide what kind of game to play.

Information given: an introduction to the game is displayed once the player access to the Information Screen. Purpose, context and structure of the simulator are then presented. An additional screen is also used to display the different roles of the game: CEO and Production Manager. In the last two information screens market guidelines and a brief presentation of key variables affecting the behaviour of the model are introduced in order to let the players realise what kind of simulated reality they are going to play with.

Model structure: the analysis of the structure can be made following two different approaches, on the one hand there is the structural analysis, and, on the other hand, the behavioural analysis. The structure analysis comprehends the possibility to see all the sectors of the model (as a Vensim sketch), relative causal loops, causal trees for each variable (causes and uses) and respective equation. The behaviour analysis allows the user to change constant values, non-linear functions, and then to simulate the

model according to the modifications introduced. Once the model is simulated it is possible to check behaviour and value of every variable, but also to load and compare different runs listing the differences between them. Pre-defined graphs, shown during the game, can be displayed as well.

Game choice: there are two different modes that can be chosen to play with the simulator. On the one hand there is the Single User Symmetric Learning Game, on the other hand the Multi-user Asymmetric Arcade Game.

Learning mode: one player game, created to make the user understand the basic structures of System Dynamics. The total demand for the market is defined by a step function (two steps), in order to let the player realise how the delays affect his decisions and the subsequent behaviour of the system. In order to test the learning process, the player is asked to indicate the expected value of the workforce every period. He has to take into account his own decision, delays and dynamics of layoff and quit rate. The competitor in the Learning mode is simulated: the price definition is based on demand coverage and market price, and orders are usually equal to sales, unless the two adjustments for inventory reduce the actual order of material. With the learning mode it is also easier to understand how the sectors are linked to each other and how local decisions affect the performance of the company. An important role in this case is played by the factors that affect the attractiveness of the product (delivery delay, price, installed base, marketing expenses). In fact the player starts elaborating elementary strategies, starting from the sales results, in order to balance the effect of the previous mentioned variables. Four sectors are involved in the attractiveness computation: production, labour force, financial resource allocation and price setting. The co-presence of those sectors makes the player think of links and casual relations existing between them.

The decision variables in the Learning Mode are: Production Demand, Desired Labour, Indicated Price, and Marketing Expenses.

The information given is: Finance (Cash, Market Share, and Production Cost); Production (Inventory, Products to be Shipped, Feasible Production from Labour and Materials, Demand of the Product). The graphs of Finite Products inventory, Labour and Demand, are also provided. In addition the player has access to a financial analysis and to the model structure analysis (comprehending sectors, causal trees, CLD diagrams and equations).

Decision Variable	Information Given	Additional Information
	Finance	**Financial Analysis**
Production Demand	Cash	Revenues
Desired Labour	Market Share	Total Costs
Indicated Price	Average Production Cost	Production Cost
Marketing Expenses	**Production**	Labour Cost
	Inventory	Inventory Cost
Expected Labour	Products to be Shipped	Material Inventory Cost
	Feasible Prod. from Labour	Marketing Expenses
	Feasible Prod. from Material	Net Income
	Demand	Cumulative Profit

Table 2: Decision variables, information given and additional information - Single Player Learning Game.
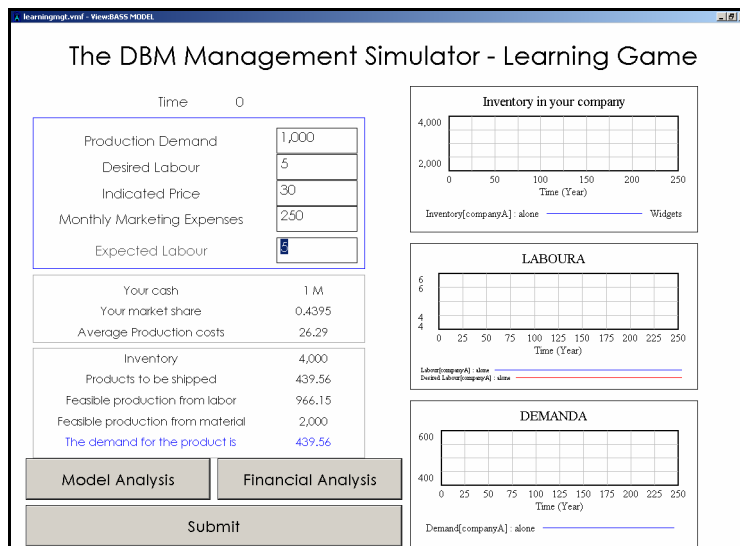

Figure 16: The DBM Flight Simulator Game, Single Player Learning Game interface.
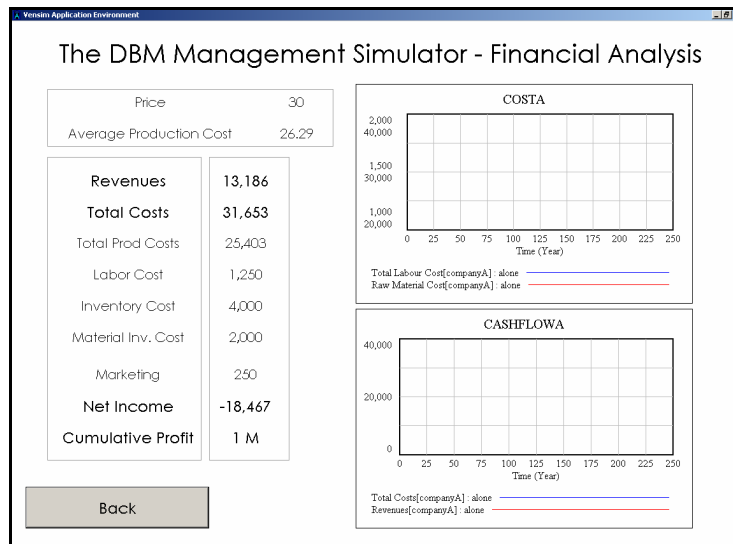
Figure 17: The DBM Flight Simulator Game, Single Player Learning Game, Financial Analysis interface.

Arcade Mode: in the Arcade Mode there are four active users, divided into two teams of two people, competing for the same market. The roles are Production Manager and CEO, for each company. Once the player decides his role, an introduction to his goals and leverages is presented and the initial conditions of the game are explicated. In this game mode the dynamic complexity is much higher than in the Learning Game. As a consequence, the creation of strategies is more difficult, and should account for the decisions of the opponent team as well as for the dynamics of the model structure. Delays, feedback and opponent's mental models must be analysed and understood in order to succeed and gain the highest market share. Over expansion of the production capacity and a subsequent price war are the most common emerging results of the strategic choices generated by a competition between two companies operating in the same market[8].

Production Manager: the decision variables for this role are "Workweek", "Production Demand" and "Desired Labour". He has to plan production and define labour desired value. His decisions will affect the attractiveness of the product, which defines sales and consequently revenues. The actions taken by the production manager will then have an effect on the CEO decision making process, and vice versa.

Decision Variable	Information Given
Workweek	Demand
Production Demand	Products to be Shipped
Desired Labour	Feasible Production from Labour
	Feasible Production from Material
	Inventory

Table 3: Decision variables, information given - Multi-player Asymmetric Game, Production Manager.

---

[8] Paich, M. and Sterman, J.D., "Boom, Bust, and Failures to Learn in Experimental Markets", Management Science, 39, 12 (1993), 1439-1458.
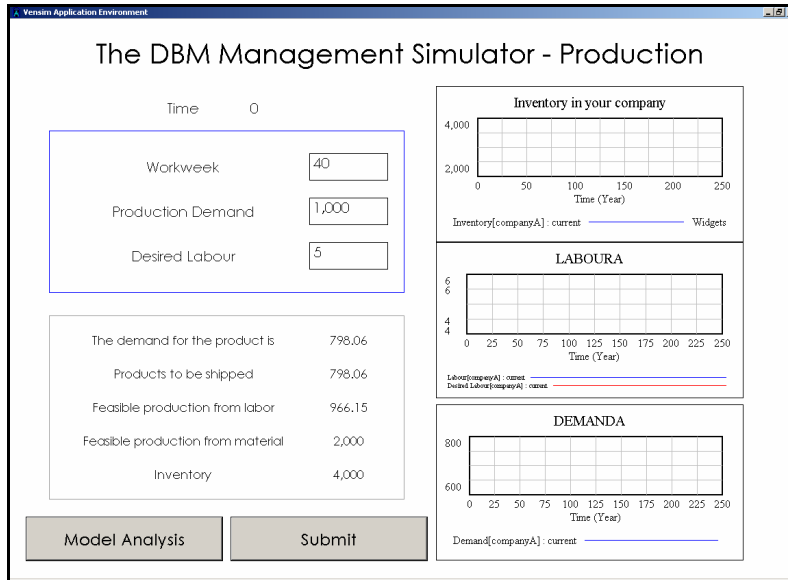
Figure 18: The DBM Flight Simulator Game, Multi-player Asymmetric Game, Production Manager interface.

CEO: the user playing this role has to manage the financial resources of the company and related issues. In fact the decision variables are "Indicated Price", "Marketing Expenses" and "Salaries". A detailed financial analysis is also available to help the player keeping under control costs and revenues. As for the Production Manager, the decision taken by the CEO will determine the performance of the team as well as the decisions of the Production Manager himself.

Decision Variable	*Information Given*	*Additional Information*
	Finance	**Financial Analysis**
Indicated Price	Average Production Cost	Revenues
Marketing Expenses	Mark-up	Total Costs
Rookie Salary	Market Share	Production Cost
Experienced Salary	Cash	Labour Cost
		Inventory Cost
		Material Inventory Cost
		Marketing Expenses
		Net Income
		Cumulative Profit

Table 4: Decision variables, information given and additional information - Multi-player Asymmetric Game, CEO.
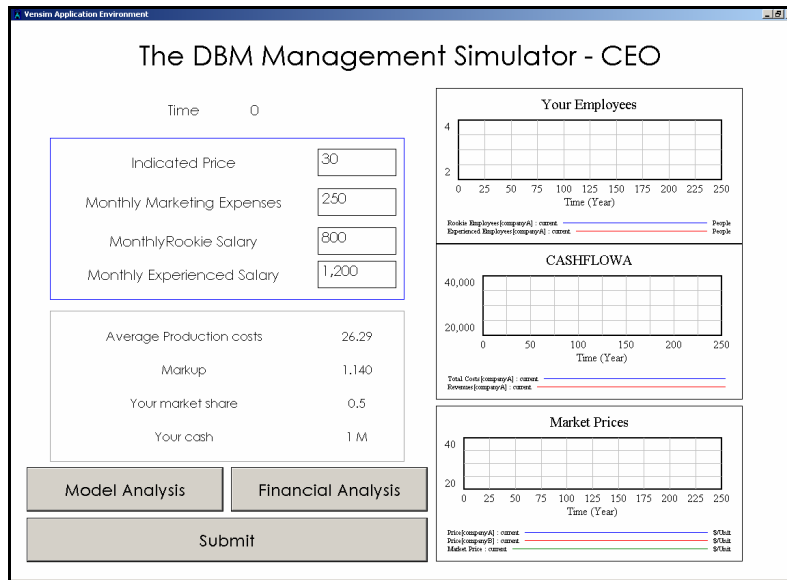
23

Figure 19: The DBM Flight Simulator Game, Multi-player Asymmetric Game, CEO interface.

Two additional roles are Supervisor and Observer. Those roles are not active, but can be useful for learning, analysing the decisions of the player and, in the case of supervisor, managing/customising the game.

Supervisor: the role of the supervisor can vary every time the game is played. He can be a passive user, helping the players understanding the task they have to face and how the system works. On the other hand, the supervisor can play an active role, speeding up the simulation, ending the game earlier to focus the attention of the players on one specific topic, or just watching the partial results of the simulation. The Supervisor task, to let the game proceed by submitting the decisions of the users, has been avoided, so that the uploading of the values happens once every five seconds and no intervention is needed. According to the structure created, the four users could also play without a supervisor.
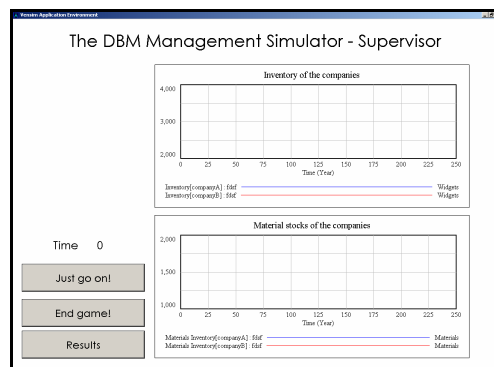

Figure 20: The DBM Flight Simulator Game, Multi-player Asymmetric Game, Supervisor interface.

The Observer has access to any information desired from the model. Every graph can be displayed and the overall results can be viewed as well as pre-loaded analysis. The graphs in the observer screen are updated every five seconds.
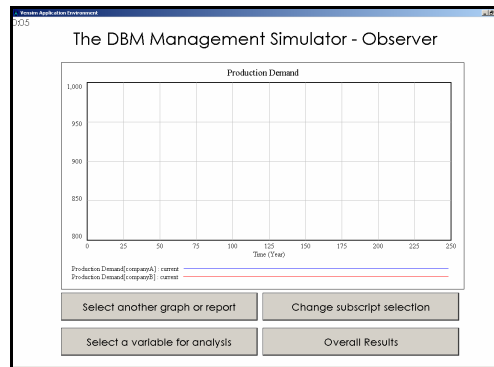

Figure 21: The DBM Flight Simulator Game, Multi-player Asymmetric Game, Observer interface.

Simulation

In order to understand better the structure of the model and to make explicit the significance of the decisions made by the players, a brief behaviour analysis seems to be necessary. The behaviour of each sector will be compared with the one originating by the main structures taken as a starting point for the creation of the model. An overall investigation of the links between modules, and how they affect each other, will be part of the behaviour analysis too.

Market Demand sector

The total demand is generated considering both initial and repeated purchase rate. The behaviour of the sector replicates the Bass Diffusion Model (i.e. epidemic adoption of the product), and the shape of the demand matches a typical product life-cycle curve. The real adoption rate accounts for potential adoption rate and production capacity of the companies.
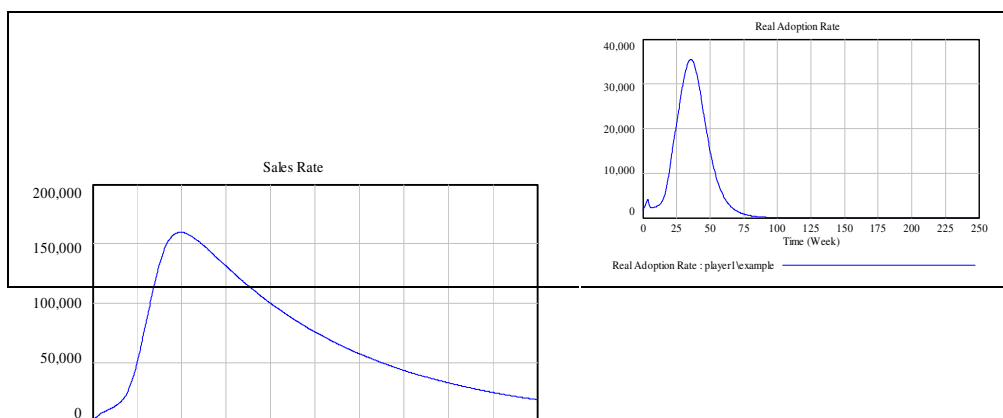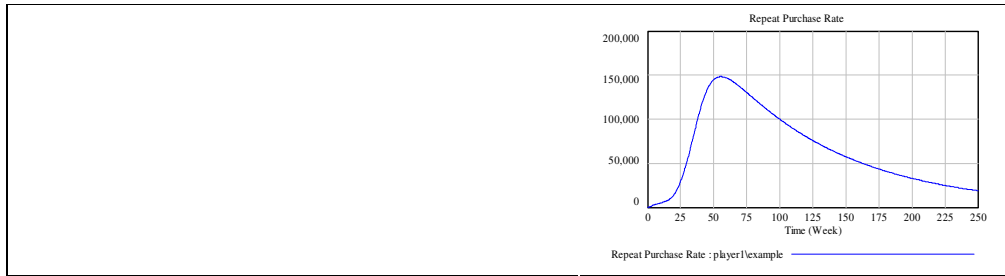
Figure 22: The DBM Flight Simulator Model behaviour of the Market Demand sector.

Firm Demand sector

The behaviour of this sector is strictly related to the structure: the Polya Process. In fact the market share tends to reach opposite extremes for the two companies taken into consideration. The factors affecting the market share definition are: installed base (cumulated shipments), price, delivery delay and marketing expenses. The installed base causes the lock-in effect, or network effect, that has a strong influence on the attractiveness of the products. The indicative market share represents the effect of all the factors, previously mentioned, on attractiveness. It defines the demand for each product, while the real market share is defined by the cumulated shipments of the company (determined by production and demand itself).
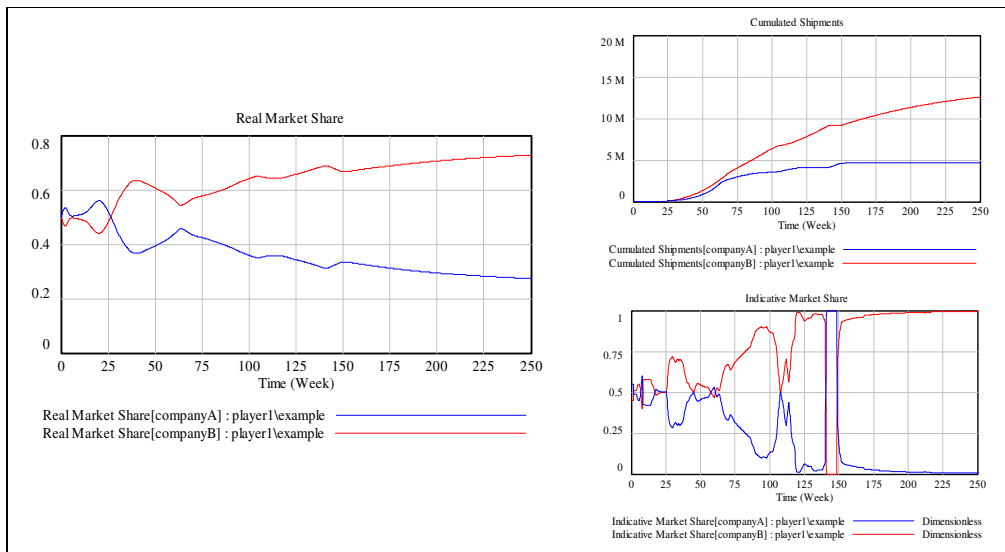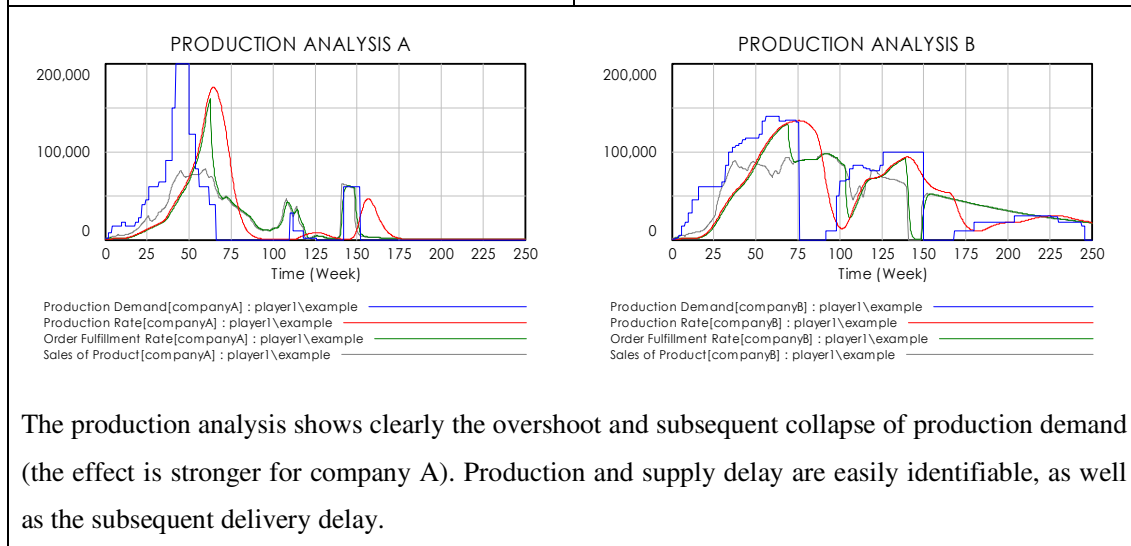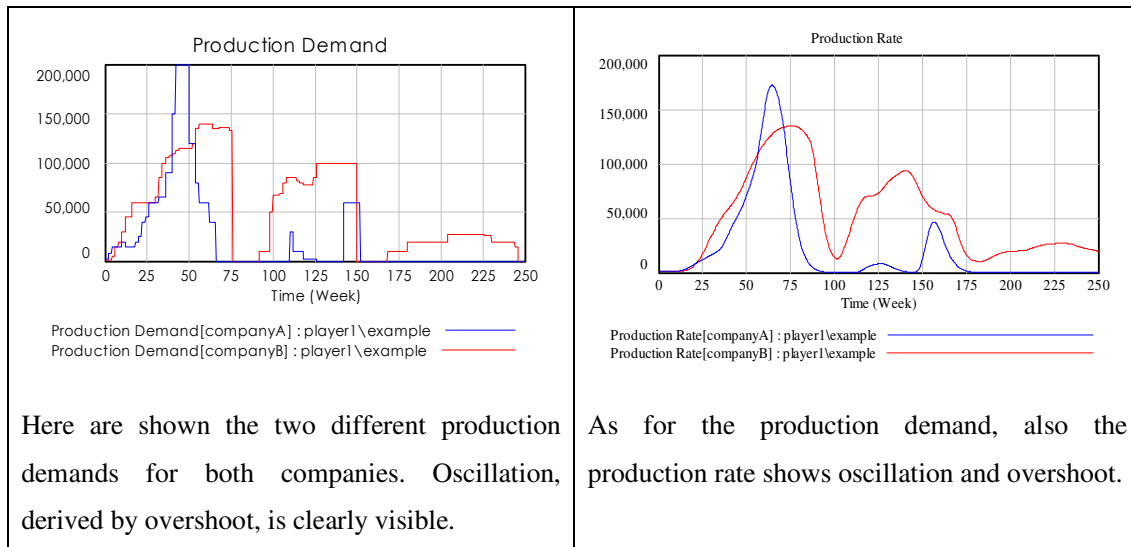

Figure 23: The DBM Flight Simulator Model behaviour of the Firm Demand sector.

Production sector

The production sector shows, as expected by the manufacturing supply chain structure, amplification, phase lag, and subsequent overshoot. The pressure imposed by the increasing demand and the presence of delays, pushes the player to order more then needed. The main consequence is the increase of delivery delay from the supplier and inventory (both material and finite, if the labour force is not adequate).

Production Demand[companyA] : player1\example
Production Demand[companyB] : player1\example

Here are shown the two different production demands for both companies. Oscillation, derived by overshoot, is clearly visible.

Production Rate[companyA] : player1\example
Production Rate[companyB] : player1\example

As for the production demand, also the production rate shows oscillation and overshoot.

Production Demand[companyA] : player1\example
Production Rate[companyA] : player1\example
Order Fulfillment Rate[companyA] : player1\example
Sales of Product[companyA] : player1\example

Production Demand[companyB] : player1\example
Production Rate[companyB] : player1\example
Order Fulfillment Rate[companyB] : player1\example
Sales of Product[companyB] : player1\example

The production analysis shows clearly the overshoot and subsequent collapse of production demand (the effect is stronger for company A). Production and supply delay are easily identifiable, as well as the subsequent delivery delay.

The inventory analysis helps us understand the overshoot in production demand and the creation of important costs of storage, both for material and finite inventory. The production delay is once again evident (phase shift between red and green line), and the backlog is shown in order to explain the reason of high production demand (when the backlog is high, the player order more than necessary, cause he doesn't take into account the orders already submitted).
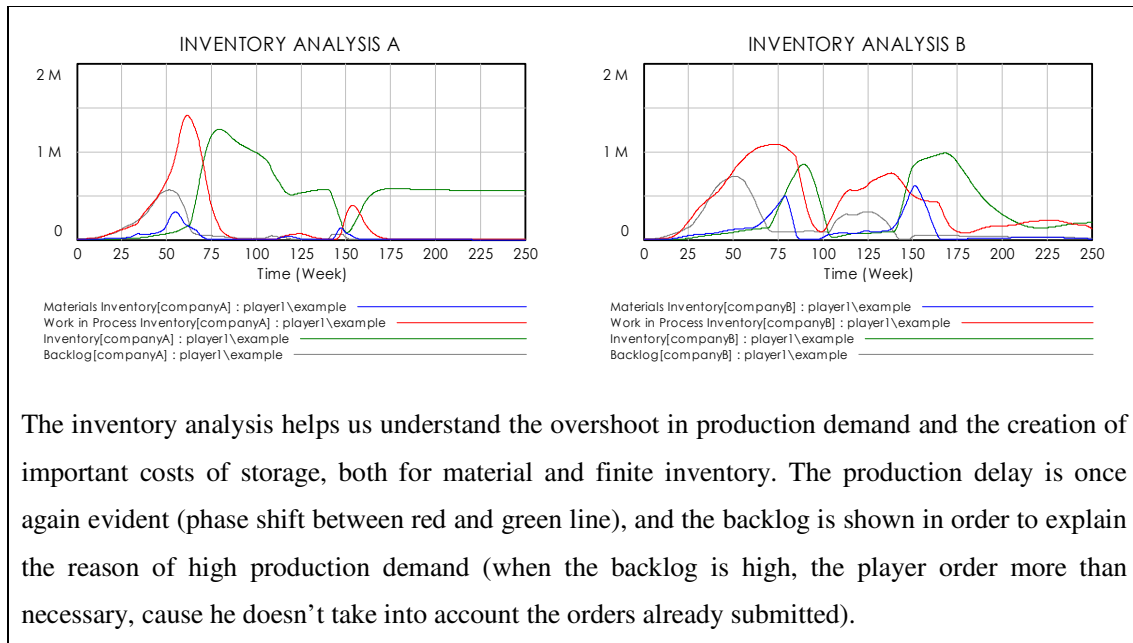
Figure 24: The DBM Flight Simulator Model behaviour of the Production sector.

Labour sector

Production and labour sectors produce basically the same behaviour. The labour module is characterised by less delays and a simplified structure, so that it is easier to understand the behaviour of the model and consequences of the decisions made. In the Learning Mode, the player is supposed to indicate the expected value of the labour force, functionality that makes easier the understanding of delays. As it is possible to capture from the following graphs, the hiring delay (vacancy creation time) is smaller than the layoff delay time.
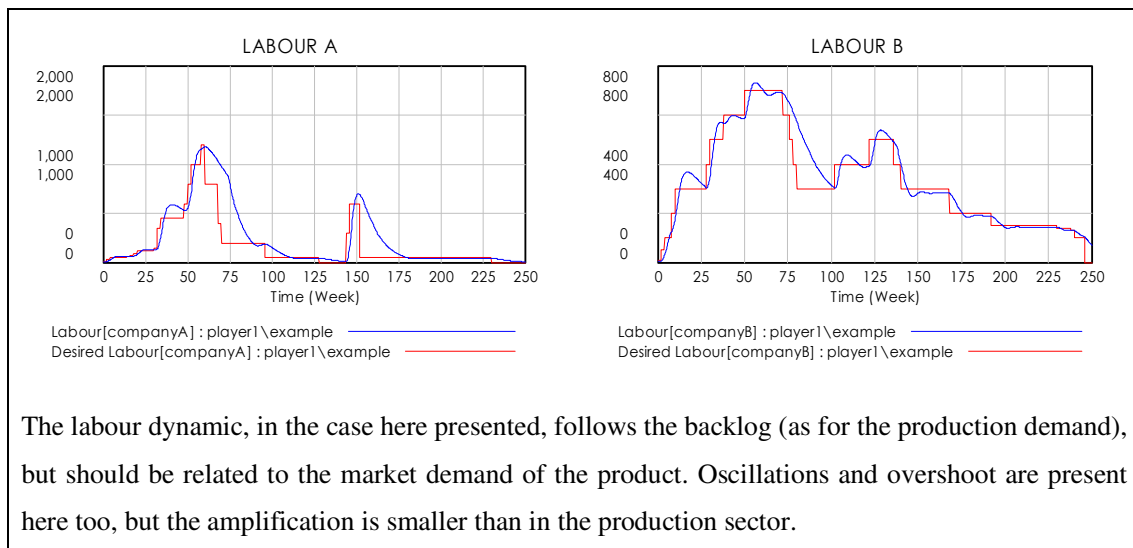

The labour dynamic, in the case here presented, follows the backlog (as for the production demand), but should be related to the market demand of the product. Oscillations and overshoot are present here too, but the amplification is smaller than in the production sector.

Figure 25: The DBM Flight Simulator Model behaviour of the Labour sector.

Costs and Revenues sector

The behaviour of the Cash, important indicator of the sector, is strongly related to the development of the real market share of the company. In detail, the strongest impact on costs is represented by the raw material cost or, eventually, by the storage costs. The price is the main determinant of revenues in two ways: first of all it directly defines the revenues, secondly it indirectly defines the demand for the product (being a factor affecting the attractiveness of the product).
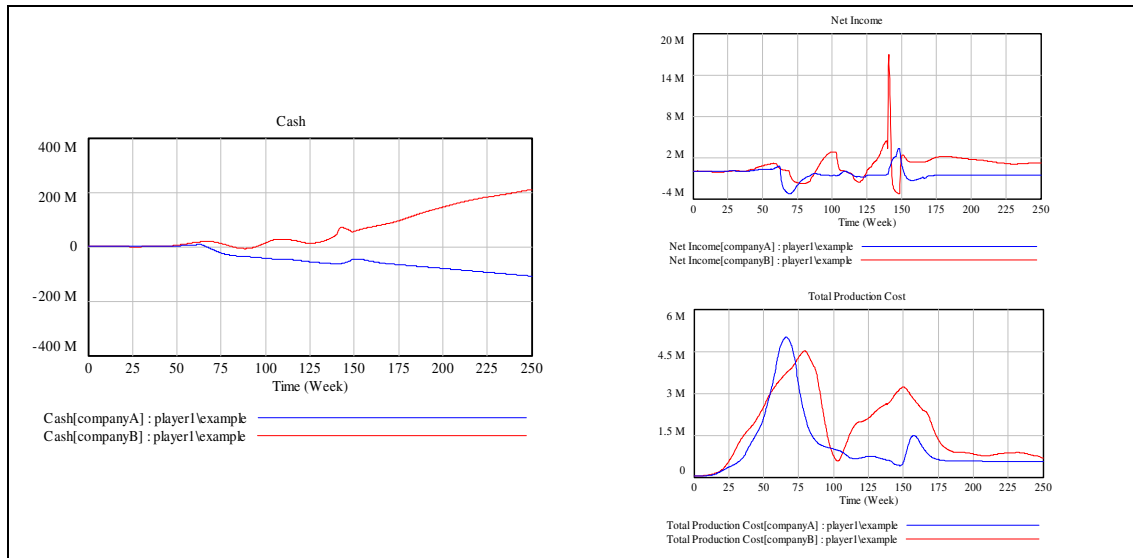

Figure 26: The DBM Flight Simulator Model behaviour of the Cost and Revenues sector.

Conclusion

The utilisation of multiplayer asymmetric simulation games can help the users understand the structure of a market and common behaviour rules. The possibility to play a learning game and an arcade simulation can give even more insights, e.g. once the structure is known emergent but assumed rational strategies can arise. Misperception of feedback, together with direct competition, can create high complexity, identifiable with the system dynamics methodology (source, causes and effects can be analysed and explained). In the present work, a statistical analysis of the decisions is missing, and the behaviour of the player cannot be characterised and grouped. The simulation game is actually being played, both leaning and arcade game, and the results will be shown as soon as possible. The patterns of behaviour previously shown are expected to be confirmed by the statistical analysis, while the learning analysis is expected to give better results in the learning mode than in the arcade mode. Further improvements are being studied (i.e. the introduction of a variable market size, improvement of the market share definition, enlargement of the labour sector including hiring and layoff rate for both categories of workers, introduction of investment in workers formation and in production capacity - capital stock -, utilisation of different market scenarios in order to allow the change of product type to better represent the environment, etc.).

We do believe that the simple simulation game here presented can make the players learn both from structure and from competition (i.e. cooperation). The delay structure is repeated in various sectors, so that every user can learn from it. The competition will show cooperation or price war, which will produce worse results for both companies. The flexibility of the software utilised makes improvements and eventual modification fast and easy to implement. System Dynamics skills are required for the modelling process, but are not strictly necessary for the supervisor/moderator role. As for the games run until now, the ILE here presented has resulted to be easily grasped and produced consistent results. In addition, learning has been achieved by playing with a stimulating reference market and the challenging competition.

References

Davidsen P.I. (1993), System Dynamics as a Platform for Educational Software Production, in Berta et al.: Proceeding of the IFIP WG3.4 Working Conference on Computer Mediated Education of Information Technology Professionals and Advanced End-Users.

Davidsen P.I. (1995) Educational Features of the System Dynamics Approach to Modelling and Simulation, Journal of Structural Learning.

Forrester J.W. (1980), System Dynamics–Future Opportunities, in Legasto, A.A.Jr. Et al.: System Dynamics, North-Holland.

Langley P.A., Paich M., Sterman J.D. (1998), Explaining Capacity Overshoot and Price War: Misperceptions of Feedback in Competitive Growth Markets, Proceeding of the International System Dynamics Conference, Quebec City, Quebec.

Legasto A. A., Forrester J. W. e Lyneis J. M., (1980), System dynamics, Amsterdam, North-Holland.

Paich M. and Sterman J.D. (1993), Boom, Bust, and Failures to Learn in Experimental Markets, Management Science, 39, 12 (1993), 1439-1458.

Senge P. M. (1990), The Fifth Discipline – The Art and Practice of the Learning Organization, London, Century Business.

Senge P. M., Sterman J. D. (1994), Systems Thinking and Organizational Learning: Acting Locally and Thinking Globally in the Organization of the Future, in: J.D.W. Morecroft and J.D. Sterman, Modeling for Learning Organizations, Portland (OR), Productivity Press.

Sterman J. D. (2000), Business Dynamics, Systems Thinking and Modeling for a Complex World, New York, Irwin/ McGraw-Hill.