Agent Based Modeling for Business Gaming Simulation and Application for Supply Chain Management Education

Motonari Tanabu[1]

[1] International Graduate School of Social Sciences
Yokohama National University, Yokohama 240-8501, Japan

Abstract

Yokohama Business Game (YBG) is a business game platform supporting a development, conduct and management of games. While YBG provides various types of business game and they are used in over 70 Japanese universities, there is a difficulty in keeping game balance or finding optimal strategy when developing and/or conducting a new game. For the problems, the author proposes a gaming-and-agent based modeling approach. An agent-based simulator is introduced to illustrate an effectiveness of the approach, and data generated by human player is used to design a new game. An application of the approach to SCM education is shown.

Keywords: business education, business simulation, business game, agent based simulation, supply chain management

1. Introduction

Business games are widely used for development of individual's decision making or organization ability in universities and companies. Yokohama National University (YNU) offers business game courses in both graduate and undergraduate (Shirai, 2008). In undergraduate classes, student manages each company in a simulated market or industry in order to get a deeper understanding of management issues or business structure. In graduate, each student is required to analyze a business system and realize it as a business game. Through the semester, each student develops own business game and sometimes receives evaluation and criticism of the game being developed. This interactive modeling and testing process enables the student to improve his/her game in terms of reality or validity.

The web based business gaming support system YBG (Yokohama Business Game) is used in these classes. YBG had developed in order to support business gaming activities such as development, conduct and management of business games. Since YBG provides a game model description language and its processing system online, developer can model a game without special development environment or language processing system.

In YBG model description language, a game model is expressed as a sequence of command sentence each of which is a composition of command, function, operator, variable, numerical data and string (Fig. 1). The language is so simple that developer who does not have an experience of computer programming can easily describe his/her model. Consequently, a lot of business game model has been developed and used in practice. In spite of simplicity of the language, there still exist some problems such as model validation, model verification, and parameter tuning. In order to resolve these problems, agent based modeling (ABM) approach had been proposed (Tanabu, 2008). In this approach, the developer builds a specific game simulator which has simple machine agents using YBG, and validates the original model and verifies the specific game simulator using YBG functionality. After getting partially valid model, the developer applies the model to a gaming simulation with human players. This simulation allows the developer to improve or modify the model based on a game result and evaluation received from the players (Fig. 2).

```
# General Definistion
def game-name SimpleGame
def max-team 5
def max-round 10
scon demand 0 1300 1334 1351 1369 1379
```

Input Definition

```
# Input
ipage price Decision
  <p>Input Selling Price</p>
ivar Selling_Price range 0 1200 700
```

Process Definition

```
# Process
tvar NofOrdRecieved
pinv NumOfOrdReceived = demand by SellingPrice;
```

Output Definition

```
# Output
opage sales sales_status public
  <H1>Sales Status</h1>
  <P>${ROUND}: Total Demand: ${demand}</p>
begintable
out teams
out teams-vars SellingPrice  NumOrdReceived
endtable
```

Fig. 1: Model description language in YBG.

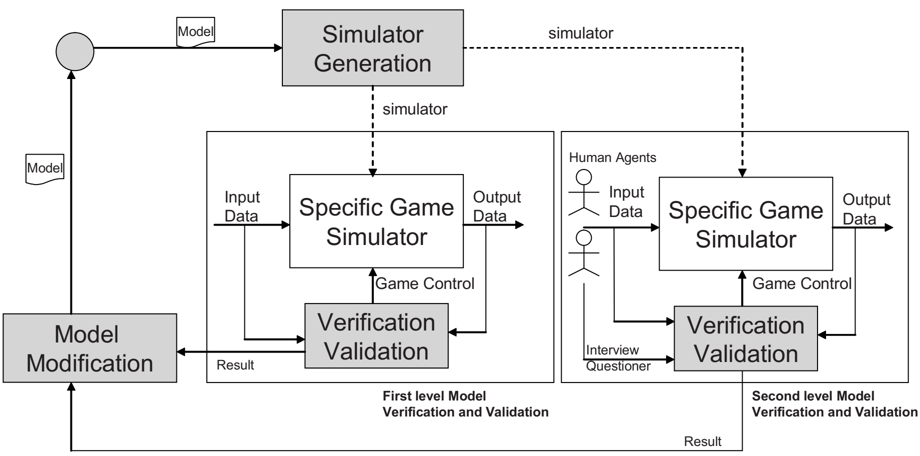


Fig. 2: Modeling cycle in YBG.

Even this modeling cycle helps the developer to improve the model being developed, YBG does not support large scale simulation such as 100,000 time periods simulation with 10,000 computer agents. When such a large simulation is required, the developer has to use a multi-agent simulator or newly create a computer program. To avoid using of multiple implementation models, the author has developed a simulator program which has affinity with YBG. The simulator consists of Perl packages and a main control program, and supports domain specific language (DSL) which is similar to YBG model description language. The game result data generated in YBG games can be used in this simulator to design a simulation for education. In this paper, through a case study of a business game modeling using YBG-like agent base simulator, we examine how gaming-and-agent approach in game modeling is effective. Application shown in this paper is a case of supply chain management education.

2. The Bakery Game

The Bakery Game is one of the most popular business game used in YBG. This was originally developed by Hiroaki Shirai who is an originator of YBG. Teaching materials and game source code of the game are provided as a sample for YBG user who conducts a business game in his/her class or develops a business game. In this section, we see an overview of The Bakery Game.

In the game, about ten players manage each bakery shop in a market. As a manager, each player has to decide three items every day (period). The first item is a selling price of the bread. If the price level is lower, more customers will visit to the player' s shop. The second item is a production order which means how many loaves of bread is manufactured to sell tomorrow (next period). The last one is material procurement which is how many pieces of frozen dough is needed for tomorrow. The business objective is to gain profit and maximize retained earnings (cumulative profit).

Each player can see the game result including message from game controller, sales report (Fig. 3), management information (Fig. 4) and accounting information (Fig. 5). All the data until the day before can be seen except the first day.

Let T be a number of game players. The i-th player's decision in t-th period is a triple

$$M_i(t)=\left(s_i(t),\ p_i(t),\ q_i(t)\right)\ (i=1\ldots T) \quad (1)$$

where each element is selling price, order quantity for production and order quantity for material, respectively. T-tuple $\boldsymbol{M}(t)=(M_1(t),\ldots,M_T(t))$ of triple in t-th period is an input for the game model. T-tuple $(s_1(t),\ldots,s_T(t))$ of selling price, a projection of $\boldsymbol{M}(t)$, is used to distribute a total demand to each players. The distributed demand, a number of customers, is considers as an uncertainty for each team process (Fig. 6).

Most of the YBG games have a structure similar to The Bakery Game. Fig. 7 shows a part of agent based simulation model written in the DSL which is developed by the author.

Sales Report

Total demand for the product for the day 01 is 1283.

Team:	01	02	03	04	05	06	07	08	09	10
Selling_Price	700	700	700	700	700	700	700	700	700	700
Number_of_Customers_Visited	128	128	128	128	128	128	128	128	128	128

Fig. 3: Sales report.

Session : Business_Game_Class Team : 10 Round : 02

Management Information

Day: 01, Team: 10, Total demand for product: 1283

Item	Unit_Price	Quantity	Amount
Selling_Price	700	-	-
Production_Order	-	100	-
Materials_Procurement	-	100	-
	-	-	-
	-	-	-
	-	-	-
	-	-	-
Breads_Ready_for_Sale	-	100	-
Number_of_Customers_Visited	-	128	-
Number_of_Breads_Sold	-	100	-
Sales	-	-	70000
	-	-	-
	-	-	-
	-	-	-
	-	-	-
Shortage_of_Breads	-	28	-
Disposed_Breads	-	0	-
	-	-	-
	-	-	-
	-	-	-
	-	-	-
Material_Inventory_on_Tomorrow_Morning	-	0	-
Materials_to_be_delivered_on_Tomorrow_Morning	-	100	-
Number_of_Breads_Ready_for_Sale_on_Tomorrow_Morning	-	100	-

Fig. 4: Management information.

Cash Flow

Day: 01, Team: 10

Item	Cash	Income	Expense
Balance_brought_forward	200000	-	-
Sales	-	70000	-
Materials_Procurement	-	-	40000
Production_Cost	-	-	10000
General_and_Administrative_Expenses	-	-	20000
Interest_Expense	-	-	0
Debt	-	0	-
Cash_Flow	0	-	-
Balance_to_be_Carried_Forward	200000	-	-

Income Statement

Day: 01, Team: 01

Sales	70000
Cost_of_Product_Manufactured	50000
Gross_Profit	20000
General_and_Administrative_Expenses	20000
Operating_Profit	0
Interest_Expense	0
Ordinary_Profit	0

Balance Sheet

Day: 01, Team: 10

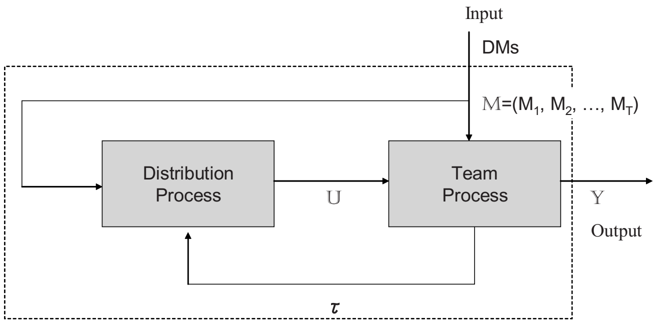
Assets

Cash	200000
Inventory_Assets	50000
Fiexd_Assets	250000
Total	500000

Assets Liabilities_and_Owners_Equity

Current_Liabilities	0
Capital_Stock	500000
Retained_Earnings	0
Total	500000

Fig. 5: Accounting information.



U: uncertainty for team process
M: manipulate variables or decision variables
τ :team state

Fig. 6: Structure of The Bakery Game as IO system.

```
# Series Constant
scon 基準需要 => 0, 128, 130, 133, 135, 137, 138, 139, 140, 141, 143;
# Global Constant
gcon 材料価格 => 400;
gcon 一般管理費 => 20000;
gcon 製造単価 => 100;
# Team variable and initial value
tvar 累積販売価格;
tvar 材料発注数 => 100;
….
# Simulation Model
for (1..MAXR) {
  …
  tlet {
    ${材料発注数()} = ${材料調達()};
    ${材料納入数()} = ${材料発注数(1)};
    ${累積材料納入数()} = ${累積材料納入数(1)} + ${材料納入数()};
    ${生産指示数()} = ${製造指示()};
    ${材料在庫数()} = ${材料在庫数(1)} + ${材料納入数()};
    ${生産可能数()} = ${材料在庫数()};
    ${生産待ち数()} = min2( ${生産可能数()}, ${生産指示数()} );
    ${材料在庫数()} = ${材料在庫数()} - ${生産待ち数()};
    ${生産数()} = ${生産待ち数(1)};
    ${累積生産数()} = ${累積生産数(1)} + ${生産数()};
    ${販売可能数()} = ${製品在庫数(1)} + ${生産数()};
    ${累積販売価格()} = ${累積販売価格(1)} + ${販売価格()};
  ${製品不人気度()} = pow(${販売価格()}-${最低有効価格()}, 3);
  };
  …
  tplot { ${来店者数()} };
  next_round();
}
exit_sim();
```

Fig. 7: Simulation model written in the DSL.

In the model, Japanese character is used for representing variables. Since this simulator program supports UTF-8 encoding, other character sets such as Chinese and Korean can be used. The model written in the DSL is similar to the model in YBG, therefore agent based simulation model can be easily converted to YBG system. The model for YBG can be also visually displayed. Fig. 8 shows a visual model of The Bakery Game. We can see three clusters in the model, that is, production process, sales and distribution process and accounting calculation process. The visualization helps developer to grasp an entire structure of the model being developed.

A simulation model written in the DSL works on any system which Perl and gnuplot are installed. An example of an agent simulation for The Bakery Game is shown in Fig. 9. Each line shows changes in number of customers for the player (team) during 50 periods.

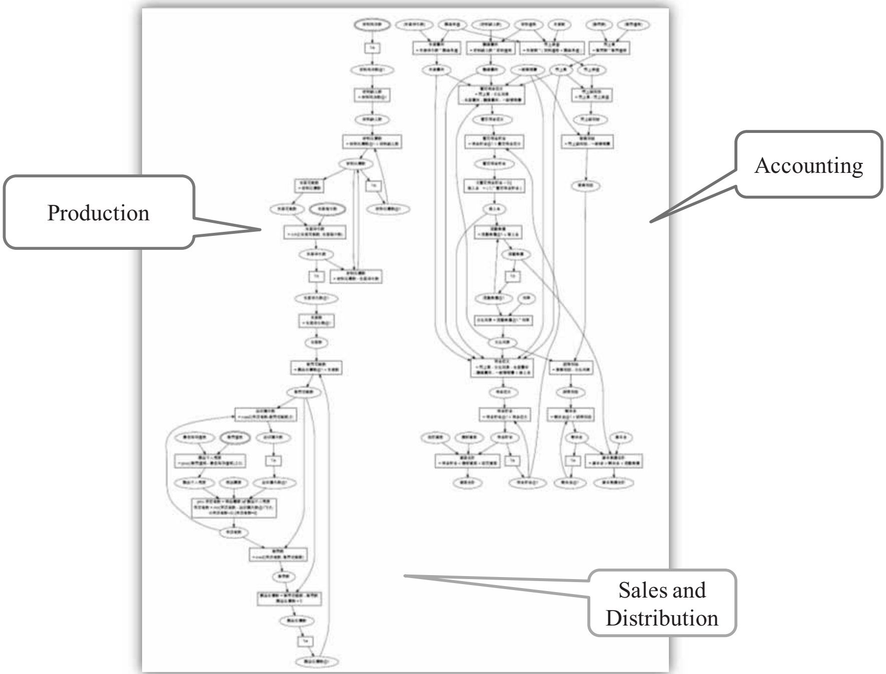


Fig. 8: Visual model of The Bakery Game.

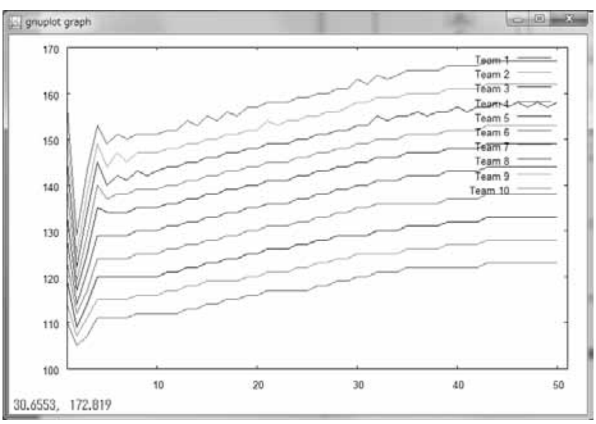


Fig. 9: Simulation on number of customers.

3. SCM Game in YBG

As a business game, The Bakery Game has elements of production and marketing. This game is usually used for introductory business education, but it has a great potential for extension. One possibility is to extend the game for a supply chain management game. In the original game, the capacity of material supplier is assumed to be infinite, that is, the player can always receive all materials that he/her ordered previous period. By changing the condition about the capacity, The Bakery Game can be used for SCM education. Even in the original setting, the bullwhip effect is identified by comparing between the coefficient of variation (CV) for order quantity of material and demand from actual game play data. When we introduce finite capacity characteristic of supplier, inventory cost, and/or backlog into the original game, it can be used in higher level of logistics education.

The Beer Distribution Game (or The Beer Game) is a famous simulation game for better and deeper understanding supply chain management. The game is played by teams of at least four players (Fig. 10).

The Beer Game is also implemented in both YBG and YBG agent simulator (Fig. 11 and Fig. 12). Using YBG-implemented Beer Game, we can easily collect play data from YBG server. These data sets can be used for analysis of human player's behavior and design of agent simulation.

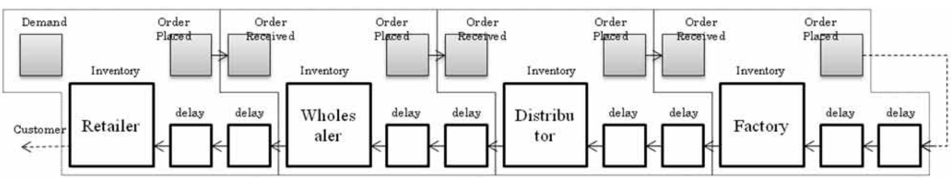


Fig. 10: Basic structure of the beer game.

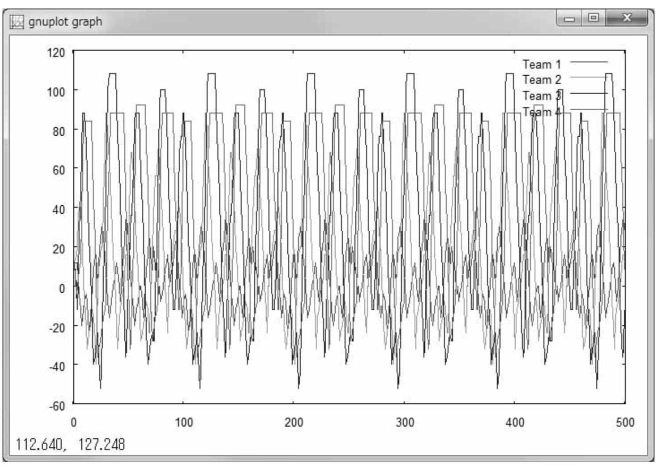


Fig. 11: Beer Game simulation output.

```
for (1..MAXR) {
    print "Period: ", ROUND, "\n";
    # 期首初期状態
    carry_forward qw(在庫数 受注残 注文数 配送遅れ); # 前期から引継ぎ
    tlet {
      @{ ${配送遅れ()} } = @{ ${配送遅れ(1)} }; # 配列のコピーは注意が必要
      ${出庫数()} = '-' ;
      ${顧客需要()} = ROUND>4 ? 8 : 4;
      ${受注数()} = TEAM==1 ? ${顧客需要()} : ${受注数(1)};
    };
    show_state("[初期状態]\n");
    # 入庫
    tlet {
      ${在庫数()} += ${配送遅れ()}->[0];
      shift @{${配送遅れ()}};
      push   @{${配送遅れ()}}, '-';
    };
    show_state("[入庫後]\n");
    # 受注
    tlet {
      ${受注残()} += ${受注数()};
      ${受注数()} = '-';
    };
    show_state("[受注後]\n");
    # 出庫
    tlet {
      ${出庫数()} = min2( ${在庫数()}, ${受注残()} );
      ${在庫数()} -= ${出庫数()};
      ${受注残()} -= ${出庫数()};
      ${配送遅れ()}->[-1] = TEAM==MAXT ? '-' : next_team->{tvar}->{'出庫数'}->[0];
    } reverse with {1};            # ループ処理を逆順に行う
    show_state("[出庫後]\n");
    # 注文票移動
    tlet {
      ${受注数()} = TEAM==1 ? '-' : prev_team->{tvar}->{'注文数'}->[0];
      i f (TEAM==MAXT) { ${配送遅れ()}->[-1] = ${注文数()} };
      ${注文数()} = '-';
    } reverse with {1};
    show_state("[注文票移動後]\n");
    # 発注
    tlet {
      ${注文数()} = ${在庫数()} < ${標準在庫数()} ? ${標準発注数()} : 0;
    };
    show_state("[発注後]\n");
    # 期末処理
    tplot { ${在庫数()}-${受注残()} }; # gnuplot で表示
    # 更新処理
    next_round();
}
```

Fig. 12: Beer Game written in the agent simulator.

Fig. 13 shows an example of a usage of actual game dataset gathered from YBG server. In this case, data of order quantity (order placed) is used for estimation of parameters of an inventory management model. For example, target inventory level Y_t at period t is sometimes assumed as $Y_t = L\hat{d}_t + z\sqrt{L}\sigma$ where L is lead time, z is a parameter for service level, $\hat{d}_t$ is demand forecast at t, and σ is the standard deviation of the error of the forecast at period t. In case of Fig. 13, parameter L and z are estimated by comparing average, standard deviation and coefficient of variation of order quantity of actual data to the values generated by the equation of the model.

These estimates are considered to be a characteristic of the human players. After identifying characteristics of each players in a supply chain, namely estimates of model parameters, we can execute new simulation using agents defined by these parameters. Replacing a rule of one or more agents with new inventory management method allows us further analysis for supply chain management, and a result of analysis can be apply to new business game.

4. CONCLUSION

The teacher or professor who uses YBG game in his/her class sometimes faces a difficulty in keeping game balance or in finding optimal strategy when developing new business game or conducting a game in large class. In this paper, gaming-and-agent based modeling approach was proposed. With combining YBG and YBG-like agent simulator which is implemented as a domain specific language processing system written in a general purpose programming language, a game developer can analyze his/her model more deeply in advance.

The Bakery Game was introduced as a basic and standard YBG game, and an actual example of use of the agent based simulator was shown. The model of the agent based simulator can be easily converted to YBG enable game model. The game result data generated in YBG games can be also used in the agent based simulator in order to design a simulation for education. A case of YBG implemented Beer Game illustrated an effective use of gaming-and-agent designing approach. The gaming-and-agent based modeling approach facilitates not only game developer who conduct business game as a teacher but also student who create his/her own business game in order to get a deeper understanding of a business structure, market characteristics.

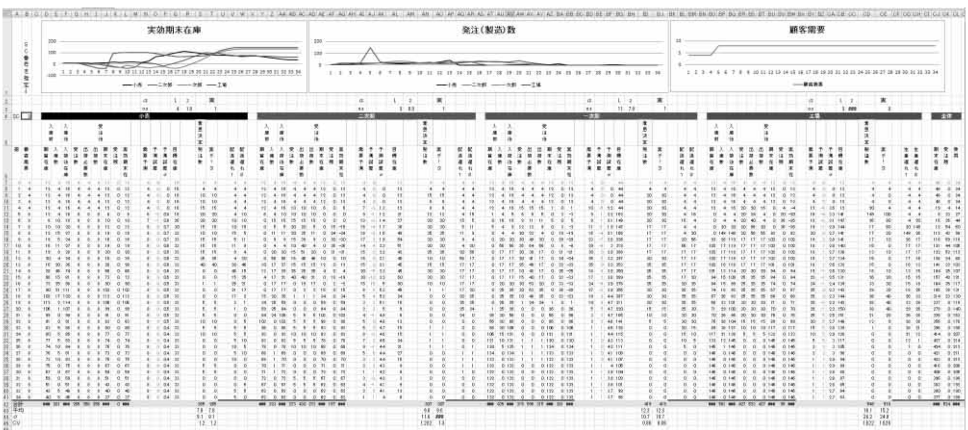


Fig. 13: Spreadsheet model for designing agent based simulation using actual game play data.

Acknowledgement

This work was supported by JSPS KAKENHI (20653020).

References

[1] Shirai, H., Tanabu, M. (2003): A do-it-yourself modeling toolkit using Web technology, *International Simulation and Gaming Yearbook*, Vol.11.

[2] Shirai, H., Tanabu, M., Terano, T., Kuno, Y., Suzuki, H., Tsuda, K. (2003): Game development toolkit for business people in Japan, *Simulation & Gaming*, Vol.34, No.3, pp.437-446.

[3] Shirai, H. (2008): A Consideration to University Programs Using Business Games (in Japanese), *Yokohama*

Business Review, Vol.29, No.3, pp.171-188.

[4] Tanabu, M. (2008): Facilitating Business Gaming Simulation Modeling, *Developments in Business Simulation and Experiential Learning*, Vol. 35, 360-367.

[5] Tanabu, M. (2004): Implementation of Business Game Activity Support System, *Journal of Electronic Science and Technology of China*, Vol.2, No.3, pp.27-32.

[6] Terano, T., Suzuki, H., Kuno, Y., Fujimori, H., Shirai, H., Nishio, C., Ogura, N., Takahashi, M. (1999): Understanding Your Business through Home-Made Simulator Development, *Developments in Business Simulation and Experiential Learning*, Vol. 26, pp. 65-71.