INFORMATICA, 2002, Vol. 13, No. 1, 73–88

© 2002 *Institute of Mathematics and Informatics, Vilnius*

Search for Dynamic Equilibrium in Duel Problems by Global Optimization

Raimundas MATULEVIČIUS
Vilties 6–85, 4580 Alytus, Lithuania
e-mail: raimunda@idi.ntnu.no

Received: May 2000

Abstract. Two examples of open-loop differential games are considered in the paper. Starting with simplified dynamic Duel, further it was developed to differential economic Duel modelling problem.

The first example regards a "military" duel of two objects, the second one is about economic duel and presents the economic competition situation. In both cases Monte Carlo models are used. The search for equilibrium is performed by global optimization.

The military model is a convenient illustration of differential game theory. It is interesting for its dynamics, it can be used for teaching purposes. The economic model shows some important features of dynamic competition. In this case objects try to maximize their final profits at the end of the period. The destruction of competitor is a feasible option to achieve this purpose.

New numerical methods and software system for the Internet environment are developed to implement this theory.

Key words: game theory, differential games, duel problem, Nash equilibrium, contract vector, fraud vector.

The Review of Dynamic Equilibrium Research

Game theory is the mathematical theory of bargaining, the essentials of which were developed by John Von Neumann and Oskar Morgenstern in their book *The theory of Games and Economic Behaviour* (1947). Von Neumann and Morgenstern restricted their attention to zero-sum games, that is, to the game in which no player can gain any additional points of winning except at another's expense. However, this restriction was overcome by the work of John Nash during the 1950s. Contemporary game theorists search for so-called Nash equilibria, that is sets of strategies used by the 1 to n players in the game such that, for each agent i, given the strategies of the other player, i has no incentive to change her strategy.

Game theory is based on analysis of conflicts and competitions. The purpose of studying games is to determine rational decisions by competitors intending to maximize some payoff or minimize some penalty. There are two players in most games: buyer – seller, pursuer – evader, client – server and etc.

The main purpose is to obtain the equilibrium situation between game objects.

Each player of the game tries to maximize his profits by controlling the initial vector. It's not sufficient to describe the problem by static variables and parameters while solving the problem. It is necessary to add dynamics. So we have differential models, where transactions and equations are differential.

Game theory problems are in the view of different scientists and researchers. Most scientists create competitive situations. For example: a differential pursuit game with one pursuer and several evaders. The aim of the pursuer is to catch all the evaders. The payoff of an evader is the time when the pursuer intercepts it. The payoff of the pursuer is defined as the negative payoff value of the last caught evader. There are several conditions under which the proposed strategies of the pursuer and evaders form Nash equilibria (Tarashina, 1998).

Usually it is necessary to construct the optimal strategies for game objects. Let's say we have a differential pursuit-evasion differential game with two evaders and one pursuer. One of the evaders (it is known in advance which one) must disappear at some predetermined time. The payoff function is defined as the minimal distance between the pursuer and the remaining evader. Optimal positional strategies for the pursuer and for both evaders under some conditions an initial position and objects' parameters are constructed (Zemskov and Pashkov, 1997).

Obviously lots of problems consider with competitive problems where one of the main parameters is time. A simple-motion differential game of k pursuers and one evader is considered. Integral constraints are imposed on the controls of the pursuers and geometrical constraint is imposed on the control of the evader. The duration of the game is fixed. A formula giving the value of the game for every initial position of players is being determined. Optimal strategies are being constructed too (Ibragimov, 1997).

A two player-differential game in which one player wants the state to reach an open target and the other player wants the state of the system to avoid the target is defined. Victory domains as the largest set satisfying some geometric conditions are constructed (Cardaliaguet and Pierre, 1995).

Open-loop differential games depend to the class of game theory too. Differential n-person games with open-loop strategies and fixed initial and final points in mixed open-loop strategies is proved. The existence of a thread-counterthread type equilibrium is shown as well (Smlo'yakov, 1995).

Let's say we have a cooperative programmed differential games, which consider n-person open-loop ("programmed") differential games. Games can be solved as cooperative. The axioms on which a game solution is based differ from Neuman-Morgenstern one in that the classical dominance is replaced by axioms on coalition formation. A game collection of mixed strategies that depend on the target. A dynamic analogue of Sharplay value is used to establish the solution (Smlo'yakov, 1996).

A noncooperative differential game in classes of programming mixed strategies is considered for differential equations with two point boundary conditions. For such type of games maximin formulas can be used for constructing algorithms of equilibria search (Smlo'yakov, 1997).

The existence of equilibrium is considered in absolute equilibrium in differential games. A differential N-person game with non-linear dynamic is being created. Each

player acts in the class of open-loop mixed strategies and tends to maximize a functional of Bolza type. The problem is to move the system from given initial state into given set. A definition of an absolute equilibrium is introduced. It is unprofitable from each coalition of players to deviate from the absolute equilibrium because the remaining players can change their strategies to punish the deviators. A necessary and sufficient condition and also sufficient one are established (Smlo'yakov, 1998).

Concepts of a random Nash equilibrium point for a random set valued operator system are introduced and the existence of random Nash equilibrium points are proved in games in which there are countably many players (Luo and Qun, 1997)

The time consistency (dynamic stability) in differential games with a discount factor is described in a nonzero sum differential games with n players, describly by $x^i = f^i(x_1, u_2, \ldots, u_n)$, $x^i \in U_i$, $x^i(0) = x_0^i$, $i = 1, \ldots, n$, where U_i are compact set in R^l and $t \in [0, \ldots, \infty)$. The payoff of the ith player is given by

$$H_i = \int_0^\infty \mathrm{e}^{-\lambda t} h_i(x(t)) \mathrm{d}\, t \tag{1}$$

the stability of Pareto-optimal solution of the game is being studied (Petrosjan, 1996).

The book "Mathematical Models in Environmental Policy Analysis" (authors Petrosjan's L.A., Zakharov's V.V., 1997) consider with a game theoretic approach. Problems of optimization of selecting population penalties and assessment of emissions are investigated. A dynamic model for air pollution control and dynamic model for environmental cooperation are studied (Petrosjan's and Zakharov's, 1997).

Lots of overviewed studies are based on pursuit-evasion differential games. It doesn't mean that only some problems can be modelled. This paper is concerned with differential Duel problems (Mockus, 2000). Dynamic equilibrium (Differential duel) problem defines the equilibrium of two flying objects, which are characterized by three parameters: the initial speed, the initial height and the shooting time. Expected winnings of both objects are defined using different algorithms of optimization, such as Mixed Strategies, Direct Search Algorithm or Strategy Elimination Algorithm.

The next problem – Economic duel – implements a new Monte Carlo model of dynamic economic competition when two servers maximize their long term profits including the option trying to obtain a monopolistic position, sometimes at the expense of current profit. New created model allows to analyze the Nash dynamic equilibrium and lets to see different characteristics of dynamic economic duel.

Both algorithms were developed and implemented in the Internet environment. That makes an easy application for distance studies and applications. Implementation of both models was made in the global optimization framework. That allows to use different global optimization methods to obtain the equilibrium solutions.

Dynamic Equilibrium Problem

Consider, as a simple illustration, two objects in space trying to destroy each other (Mockus, 2000). They may be rockets, planes etc. Both of them have their initial parameters and states, for example starting speed and height. Objects move according to differential law and their purpose is to destroy each other. Objects try to find the optimal strategies for choosing their firing times.

Movements of the objects are described in two-dimensional "high-time" space by the first-order differential equations

$$\frac{\mathrm{d}\, z(t)}{\mathrm{d}\, t} = az(t), \tag{2}$$

$$\frac{\mathrm{d}\, w(\tau)}{\mathrm{d}\, \tau} = bw(\tau), \quad \text{where } \tau = 2 - t. \tag{3}$$

The solution of these equations defines trajectories of the objects

$$z(t) = z_0 \mathrm{e}^{\,at}, \tag{4}$$

$$w(\tau) = w_0 \mathrm{e}^{\,b\tau}, \tag{5}$$

where z_0, w_0 – initial points, a, b – "climbing" rates, and t_1, t_2 – firing times.

The main purpose of the game is to find such a contract vector, what the difference between contract and so called fraud vector would be the smallest. The strategy is fraud irrelevant if equilibrium is reached – the difference is zero.

We define firing times t_1 and t_2 as *Contract vector* – it is such a vector when objects of the game agree about it before starting the duel. The transformation $z = T_t(z)$ is called *Fraud vector*.

$$\begin{aligned} t_1^1 &= \arg\max U(t_1, t_2^0), \\ t_2^1 &= \arg\max V(t_1^0, t_2). \end{aligned} \tag{6}$$

Here $U(t_1, t_2^0)$ is the expected winning of the first object. It is the solution of the corresponding bimatrix game at fixed firing times t_1 and t_2^0. $V(t_1^0, t_2)$ is the expected winning of the second object defined as the solution of the bimatrix game at given firing times t_1^0 and t_2. Winnings $U(t_1, t_2^0)$ and $V(t_1^0, t_2)$ can be defined using different algorithms: for example Mixed Strategies Algorithm (MSA). Using it we have to solve linear programming (LP) problem. If there is now solution of this problem, we search for pure strategies using Direct Search Algorithm (DSA). Otherwise Strategy Elimination Algorithm (SEA) is a good alternative to obtain equilibrium.

The equilibrium is achieved, if the minimum is zero.

$$\min \left\| z - T_t(z) \right\|^2. \tag{7}$$

The software was created to illustrate main principles of the differential Duel problem. It was created using Java programming language, which provides all the possibilities of

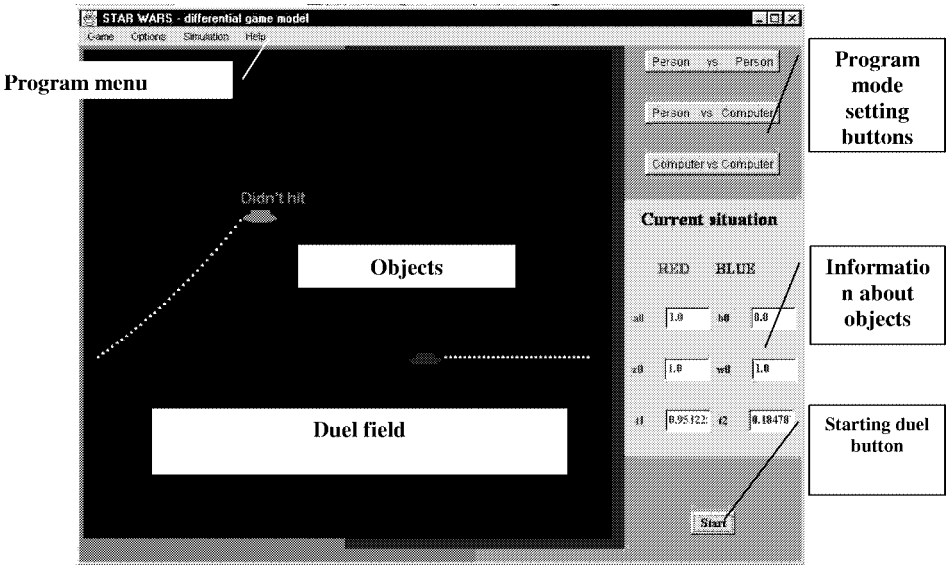


Fig. 1. *Duel* program main window and its components.

running the program by Internet. The main window of the program and its components are showed in Fig. 1.

The objects trajectories in dynamic Duel are defined by linear differential equations. The user of the program can choose three different modes of running the program:

Person vs Person – user can set all the parameters (z_0, w_0 – initial points, a, b – "climbing" rates and t_1, t_2 – firing times) of both objects himself.

Person vs Computer – user can set initial parameters of the first object. Computer generates parameters of the second object.

Computer vs Computer – both objects parameters are calculated by computer.

After starting the duel (button *Start*) the problem simulation starts. Objects start to move according their current parameters and fire to each other at chosen firing time. After the finishing duel simulation the window of results is displayed (Fig. 2). Window provides all the main duel results – the winner, minimized function (Eq. 7), fraud vector (as shooting time) (Eq. 6).



Fig. 2. Dynamic Duel *Result* window.

Modelling differential game model, looking for this problem equilibrium lot's of different algorithm (Mixed Strategies Algorithm, Direct Search Algorithm, Strategy Elimination Algorithm) are used. Realization results showed, that to find equilibrium is quite simple, therefore fraud vector is almost symmetric for contract vector. That is why objects "hear" each other shooting times and remained object (if it survives) can choose the best strategy to shoot. The classical conclusion can be done – there is simple decision for simple problem. The equilibrium in differential Duel problem exists and it is close to nil.

You can find some calculation examples in the Appendix 1.

Differential Duel problem is interesting for its dynamics, it can be used for teaching purposes, to show the meaning of contract and fraud vectors, dynamic states and etc. The results could be used creating new computer games and illustrating lectures in war academies.

Economic Duel Problem

Much more interesting is an Economical Duel problem – Nash equilibrium problem – where almost the real situation of market is being modelled, when several servers compete for clients.

The "market" is represented by a collection of independent servers (Mockus, 2000). Each server tries to maximize its profit by setting of optimal service prices and optimal server rates. The server rate is the average number of customers that would be served in non-stop operation. Thus, we can consider this rate as the server capacity, or running cost.

$$u_i = a_i y_i - x_i, \quad i = 1, \ldots, m, \tag{8}$$

where u_i is the profit, y_i – service price, a_i – rate of customers, x_i – running cost, and i is the server index.

Suppose that a server gets broken, if the accumulated losses exceed some credit threshold. Here the bankrupt server is eliminated and the remaining one assumes a monopolistic position.

Denote by $U_i(t)$ a profit accumulated by the ith server at a time

$$U_i(t) = \int_{t_0}^{t} u_i(t) \mathrm{d}\, t, \tag{9}$$

where $u_i(t)$ defines a profit at a moment t

$$u_i(t) = a_i(t) y_i(t) - x_i(t). \tag{10}$$

Here $a_i(t)$ is a rate of customers of the ith server at a moment t. This rate is formally defined as a limit of the fraction

$$a_i(t) = \lim_{\delta \to 0} \frac{A_i(t+\delta) - A_i(t)}{\delta}, \tag{11}$$

where $A_i(t)$ is a total number of customers arrived during an interval $(0, t)$, $t < T$. The zero denotes the starting time of the system. T is the end of the operation period, the "horizon".

Assume that both servers are planning their dynamic competition in advance. Each server i defines trajectories of the service price $y_i(t)$ and the server capacity $x_i(t)$. We illustrate the idea by a simple example. Assume that servers control initial values $y_i(0)$, $x_i(0)$ and rates of change b_{yi}, b_{xi} of parameters $y_i(t)$, $x_i(t)$. Then the trajectories are defined by equations

$$\frac{\mathrm{d}\, y_i(t)}{\mathrm{d}\, t} = b_{yi} y_i(t) \tag{12}$$

and

$$\frac{\mathrm{d}\, x_i(t)}{\mathrm{d}\, t} = b_{xi} x_i(t). \tag{13}$$

The corresponding solutions are

$$y_i(t) = y_i(0) \exp\{b_{yi} t\}, \tag{14}$$

$$x_i(t) = x_i(0) \exp\{b_{xi} t\}. \tag{15}$$

The quality of service is defined by the average time lost by customers while waiting for services. A customer prefers the server with lesser total service cost.

$$c_i \leqslant c_j, \quad j = 1, \ldots, m, \quad j \neq i, \quad c_i \leqslant c_0. \tag{16}$$

The total cost includes the service price plus waiting losses.

$$c_i = y_i + \gamma_i. \tag{17}$$

A customer goes away, if the total cost exceeds a certain critical level.

$$\min_i c_i > c_0. \tag{18}$$

A flow of customers is stochastic. Service times are stochastic too. The results are obtained by Monte-Carlo simulation. The Monte Carlo simulation is used to consider more general cases.

The basic steps of the Monte Carlo algorithm:

1. Fix the zero event time $t = t^0 = 0$ when the first customer arrives.
2. Define the zero state vector n^0, by the condition: $n^0 = 0$, $i = 1, 2$ and the zero state vector h^0 by the condition: $h^0 = y_i = 0$, $i = 1, 2$, because there are no customers waiting for service yet.
3. Define the next arrival into the system by the expression

$$\tau_a(t) = -1/a(t) \ln(1 - \eta), \tag{19}$$

where η is a random number uniformly distributed in interval $[0 \dots 1]$.

4. Chose the best server i^0 for the first customer by the condition $i^0 = \arg \min_{i=0,1,2} h_i$, where $h_i = y_i$ because $\gamma_i = 0$ since there are no customers waiting yet, $i^0 = 0$ means that the customer abandons the service.
5. Define the time of event when the first customer will be served by the server i^0 using the expression

$$\tau_{i0} = -1/x_{i0}(t) \ln(1 - \eta). \tag{20}$$

6. Define the next event t^1 by comparing the arrival time $\tau_a(t)$ and the service time τ_{i0}.

$$\begin{aligned} &\text{if } \tau_a(t) < \tau_{i0}, \quad \text{then } t^1 = \tau_a, \\ &\text{if } \tau_a(t) > \tau_{i0}, \quad \text{then } t^1 = \tau_{i0}. \end{aligned}$$

7. Define the system state at the next event t^1:

$$\begin{aligned} &\text{if } t^1 = \tau_a(t), \quad \text{then } n_{i0} = 1, \text{ and } n_i = 0,\ i = 1, 2,\ i \neq i^0, \\ &\text{consequently } h_{i0} = y_{i0} + 1/w_{i0} \text{ and } h_i = y_i,\ \ i = 1, 2,\ \ i \neq i^0, \\ &\text{if } t^1 = \tau_{i0}, \text{ then } n_i = 0,\ \ i = 1, 2, \text{ and } h_i = y_i,\ \ i = 1, 2,\ \ i \neq i^0. \end{aligned}$$

Definition of later events and system states is longer but the main idea remains the same. This way one estimates customer rates $a_i(t)$, that defines server profit $u_i(t) = a_i(t) - x_i(t)$, where $i = 1, 2$.

The profit of each individual server is maximized assuming that their partners respect some agreement about the service price and rates. We call this the "Contract Vector". Service prices and rates obtained by maximizing individual profits of the servers transform the Contract Vector into the Fraud Vector. The optimization problem is to search for such Contract Vector that reduces the deviation of the Fraud vector from the Contract Vector. That makes the fraud less relevant. The fraud is irrelevant and the Nash equilibrium is achieved, if this deviation is zero. Then servers cannot increase their profits by changing service prices and server rates agreed by the contract.

The search is similar to differential game model. We fix the initial values, the Contract Vector $z^0 = (x_i(0)^0, b^0_{xi}, y_i(0)^0, b^0_{yi},\ i = 1, 2)$. The transformed values, the Fraud vector $z^1 = (x_i(0)^1, b^1_{xi}, y_i(0)^1, b^1_{yi},\ i = 1, 2)$, is obtained by maximizing the profits of each server i. The maximization is performed assuming that all partners $j \neq i$ honors the contract.

$$\begin{aligned} &\left(x_i(0)^1, b^1_{xi}, y_i(0)^1, b^1_{yi}\right) \\ &\quad = \arg \max_{x_i(0), b_{xi}, y_i(0), b_{yi}} U_i\Big(x_i(0), b_{xi}, y_i(0), b_{yi}, x_j(0)^0, b^0_{xj}, y_j(0)^0, b^0_{yj}, \\ &\qquad\qquad j = 1, 2,\ j \neq i\Big) \quad i = 1, 2. \end{aligned} \tag{21}$$

Here the symbol U_i denotes the accumulated profit (9). Formally, condition (21) transforms the vector $z^n = (x_i(0)^n, b^n_{xi}, y_i(0)^n, b^n_{yi},\ i = 1, 2) \in B \subset R^8$, $n = 0, 1, 2, \ldots$ into the vector z^{n+1}. To make expressions shorter denote this transformation by T

$$z^{n+1} = T(z^n), \quad n = 0, 1, 2, \ldots. \tag{22}$$

One may obtain the equilibrium at the fixed point z^n, where

$$z^n = T(z^n). \tag{23}$$

The fixed point z^n exists, if the feasible set B is convex and all the profit functions (8) are convex. We obtain the equilibrium directly by iterations, if the transformation T is contracting. If not, then we minimize the square deviation

$$\min \|z - T_t(z)\|^2. \tag{24}$$

The equilibrium is achieved, if the minimum (24) is zero.

The realization of economic duel provides the possibilities of analysis, demonstration and visualization. The main window of the economic duel problem is showed in Fig. 3.

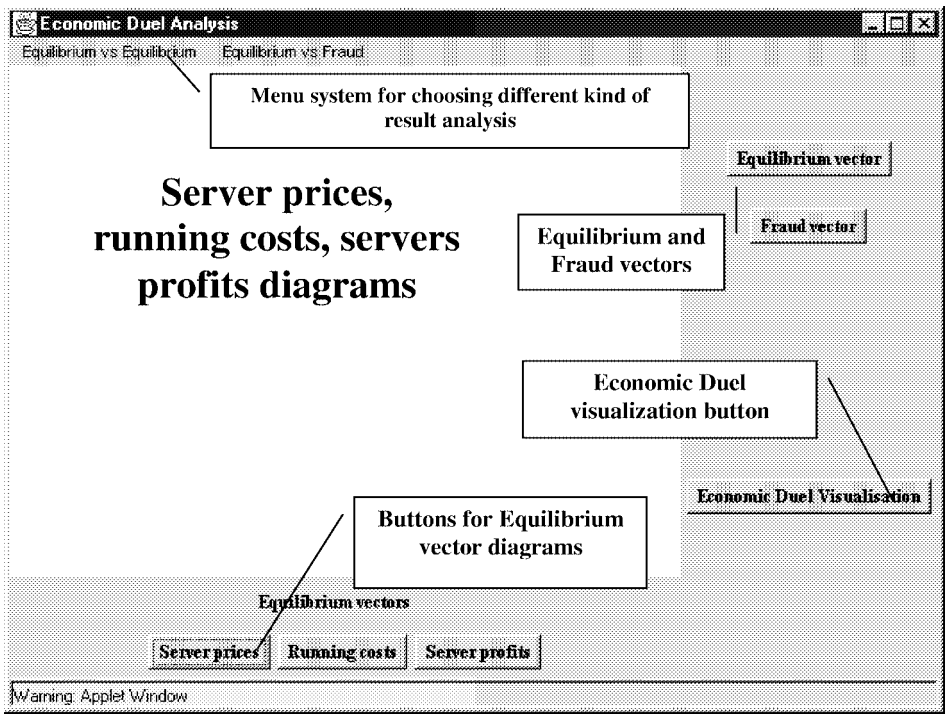


Fig. 3. *Economic Duel Analysis* main window.

There are some different modes for viewing Economic Duel results:

a) first server equilibrium vector vs second server equilibrium vector;

b) first server equilibrium vector vs second server fraud vector;

c) first server fraud vector vs second server fraud vector.

Program also provides possibilities of viewing various diagrams of server service price, running costs and server service accumulated profit of different simulation modes. It allows to compare different parameters of servers. This can be done using different control button and menu items. You can find some calculation examples in the Appendix 2.

Program results are displayed using concrete (discrete) time moments. It simplifies the visualization of the program, and helps to understand the whole principles of the task better (Figs. 4, 5, 6).

The achievement of fraud vector, the process, different parameters and vectors are showed in the part of the program called *Economic Duel Visualization*. The calculation

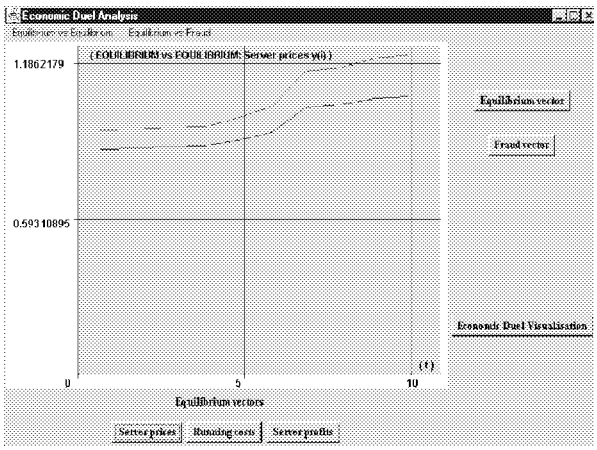


Fig. 4. Equilibrium vs Equilibrium – Server prices.

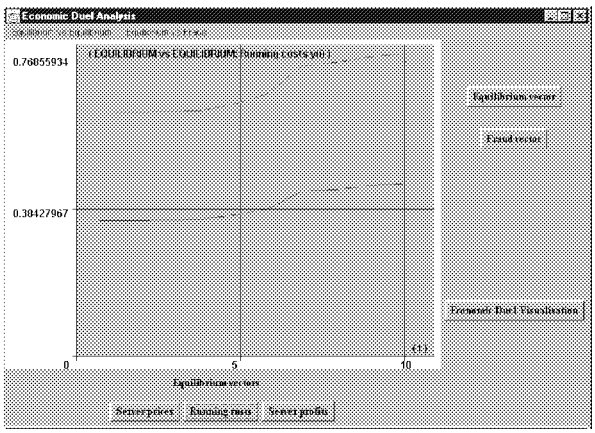


Fig. 5. Equilibrium vs Equilibrium – Running costs.

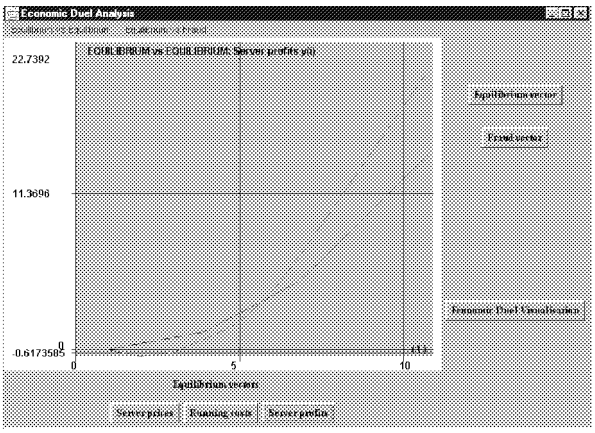


Fig. 6. Equilibrium vs Equilibrium – Server profits.

process is displayed using the command *Economic Duel Visualisation* (Figs. 7, 8, 9). It is the moments of client coming (*Next arrival* – Eq. 19, 20), servers total prices at the moment (*Server cost at the moment* – Eq. 17), waiting clients' queues (*Waiting queue of* n *clients*), client choices (Eq. 18), the whole monopolistic market situation and other specific things of the problem.

One of the main Economic Duel peculiarities is the monopolistic situation in the market. Let's say both servers take a credit from some credit institution, for example a bank. Server goes bankrupt if its profit becomes lower than credit limit. The remaining server

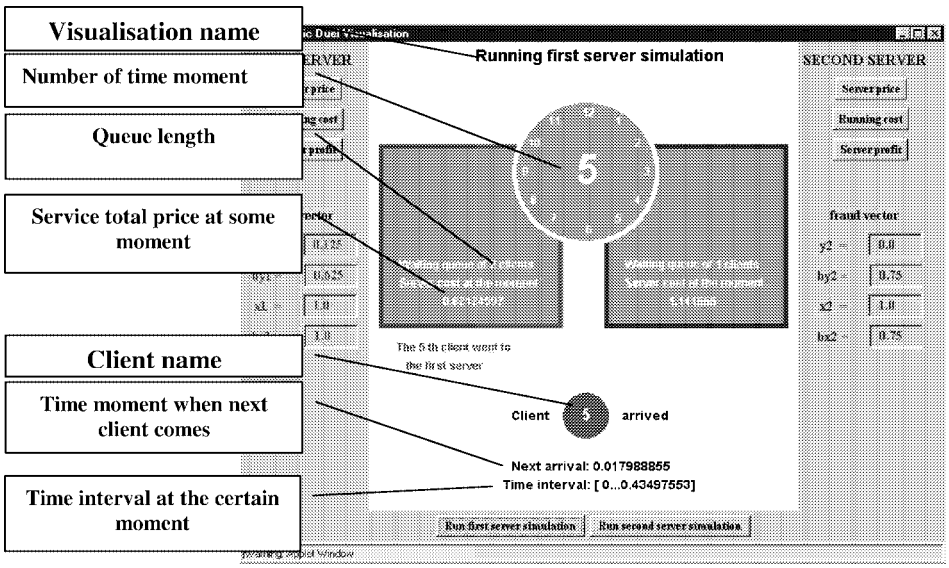


Fig. 7. Main components of started visualization window.

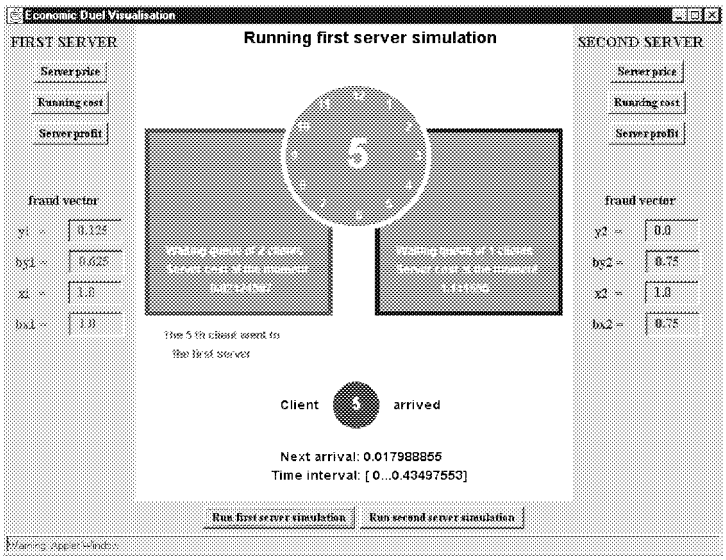


Fig. 8. The beginning of visualization.

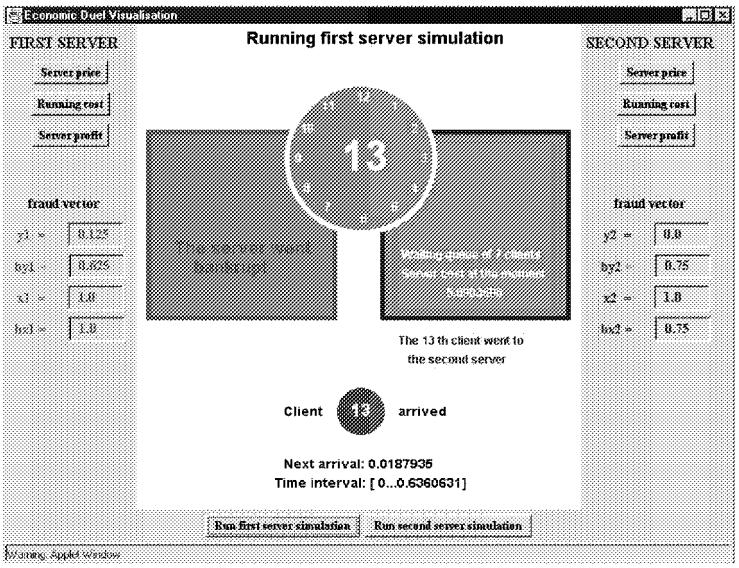


Fig. 9. The end of visualization.

becomes monopolist of the market.

In the monopolistic case, the profit of the remaining server i

$$u_i^* = \max_{x_i, y_i} (a_i^* y_i - x_i) . \tag{25}$$

Here the customer rate a_i^* is from the condition that a customer, arriving at a time t, goes to the monopolistic server i, if $h_i(t) \leqslant h_0$, where

$$h_i(t) = n_i(t)/x_i, \tag{26}$$

$n_i(t)$ is the number of customers waiting for services at the time t. A customer abandons the service system, if $h_i(t) > h_0$.

Such situation is showed in Figs. 10, 11, 12.

The first (red) server went bankrupt, because it achieved his critical profit limit, which was equal to 1. The server went bankrupt with its profit -1.3719476. The second (blue) server became the monopolist of the market.

We can see, that seeking for the monopolist case the server for some time may work with a unfavorable profit, waiting for other market server, competitor, going to bankrupt. The main purpose of the server is to maximize its final period profit in order to become market monopolist.

Usually analyzing dynamic problems it is not enough to use first order differential equation. Second order differential equations are used:

$$\mathrm{d}^2 y_i(t)/\mathrm{d}\,t^2 = g_{yi}\mathrm{d}\,y_i(t)/\mathrm{d}\,t + b_{yi}y_i(t), \tag{27}$$

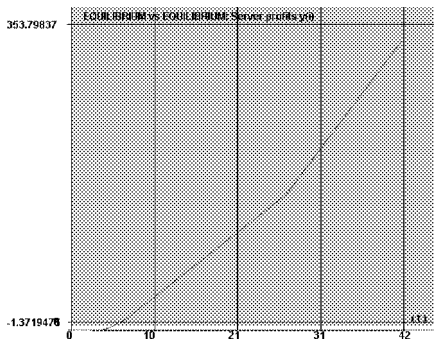


Fig. 10. Bankrupt situation – server profit diagrams.

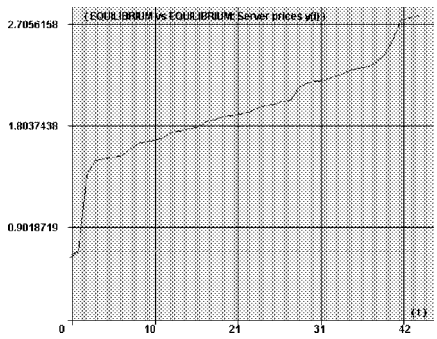


Fig. 11. Bankrupt situation – server price diagrams.

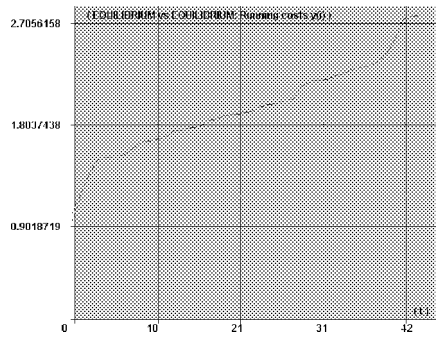


Fig. 12. Bankrupt situation – running costs diagrams.

$$\mathrm{d}^2 x_i(t)/\mathrm{d}\,t^2 = g_{xi}\mathrm{d}\,x_i(t)/\mathrm{d}\,t + b_{xi}x_i(t), \quad (z^n \in \mathrm{R}^{12}). \tag{28}$$

But in such case we have much more difficult problem which is needed more studying and analysis.

It is obvious that the game theory and dynamic problems are actual in our life and help to model different situations of our life. Realizations of Dynamic equilibrium problems are not only good tools for analyzing differential game models, but it also provides the possibility after adding some conditions to solve economical problems of real live, research differential game models. Realization – is the prototype of real modelling problems. Differential game programs are actual because of their theoretical principles. The demonstrations and applications can be fully used in distance education using Internet.

Running Applications through Internet

One of the main nowadays purposes is a fast and effective transaction of different kind of information. That is the advantages of Internet. Dynamic equilibrium game models' realizations can be run using Internet tools in full. For programming realizations of problems Java programming language was chosen. This language provides good possibilities to use all advantages of Internet. Java inherited lots of advantages from $C++$. As lots of other object oriented programming languages, Java has class libraries, which provide main instruments for programming. Java supports main Internet protocols, and other things, which allow this language to become popular. Platform independence is the main advantage creating similar projects. Applications can be run as Java*applets* and as Java *applications.* The advantage of running it as Java applet is obvious - it can be used over the net. The advantage of the Java application is that the newest Java features, which are supported by the Java Development Kit (JDK) can be used.

Therefore, differential Duel problems were plugged to *Global Minimizer for Java* (gmj) (Grybauskas, 1999) interface.

To run the program as Java application you have to install to your computer Java programming language. The archive of the program you can find in Internet http://www.javasoft.com.

You can find the programs realizations (archive *gmj_eco.zip*) and download it from the Internet too – *http://vaidila.vdu.lt/ i4rama/Dvikova/index.html*. After installing Java programming language and after extracting *gmj_eco.zip* file, run the program with the help of command *appletviewer*.

Appendix 1

There are some different calculations of Differential Duel.

Methods	**MIG**			**Bayes** (Initial points = 5)		
Iteration number	10	50	100	10	50	100
a_0	0	0	1	0	0	1
z_0	1	1	1	0	0	1
b_0	0	0	0	1	1	1
w_0	1	1	0	1	1	1
t_1^0	0.134	0.072	0.423	0.966	0.792	0.981
t_2^0	0.071	0.073	0.445	0.937	0.793	0.995
t_1^1	0.1	0.1	0.45	0.95	0.8	1
t_2^1	0.15	0.1	0.45	1	0.8	1
$\log F(x)$	-4.915	-6.513	-7.214	-5.452	-9.139	-7.88
$F(x)$	0.00733	0.00148	0.000736	0.004287	0.0001074	0.0003782

Appendix 2

There are some calculations of Economic Duel.

Methods	**MIG**			**Bayes** (Initial points = 5)		
Iteration number	10	50	100	10	50	100
y_1^0	0.757	0.61	0.838	0.75	0.903	0.873
by_1^0	0.201	0.971	0.197	0.25	0.853	0.949
x_1^0	0.603	0.929	0.873	0.75	0.981	0.945
bx_1^0	0.29	0.782	0.308	0.75	0.716	0.976
y_2^0	0.359	0.458	0.49	0.25	0.886	0.558
by_2^0	0.979	0.582	0.903	0.75	0.909	0.822
x_2^0	0.861	0.886	0.76	0.25	0.559	0.766
bx_2^0	0.921	0.984	0.59	0.75	0.875	0.859
y_1^1	0.75	0.875	0.875	1	0.875	1
by_1^1	0.625	0.875	0.375	0	1	0.875
x_1^1	1	1	0.875	1	1	1
bx_1^1	0.875	0.375	1	1	1	0.875
y_2^1	1	0.875	0.75	0.75	1	0.75
y_2^1	0.75	0.625	0.875	1	1	1
x_2^1	1	0.875	1	0.875	1	1
bx_2^1	1	1	0.625	0.875	0.75	1
$F(x)$	1.652	0.426	0.639	0.969	0.226	0.178

References

Cardaliaguet, P. (1995). A differential game with one target and two payers. *Bull. Polish Acad. Sci. Math.*, **43**(1), 69–74.

Grybauskas, M. (1999). *Global Minimizer for Java (GMJ), version 2.0. Developer's and Power User's Guide*. Institute of Mathematics and Informatics, Vilnius.

Ibragimov, G.I. (1997). On the differential game with fixed termination time. *Uzbek. Mat. Zh.*, **2**, 29–36.

Luo, Q. (1997). Existence theorems of random Nash Equilibrium point for random set-value maps. *Northeast Math. J.*, **13**(2), 247–252.

Mockus, J. (2000). *A Set Of Examples Of Global And Discrete Optimization. Application of Bayesian Heuristic Approach*. Kluwer Academic Publishers, Dordrecht-Boston-London.

Petrosjan, L.A. (1996). The time consistency (dynamic stability) in differential games with a discount factor. *Game theory and Applications*, 47–53.

Petrosjan's, L.A., V.V. Zakharov's (1997). *Mathematical Models in Environmental Policy Analysis*. Nova Science Publishers, Inc, Commack, NY.

Smlo'yakov, E.R. (1995). Multivalued noncooperative differential games. *Differ. Uravn.* **31**(12), 2013–2019; translation in *Differential Equations*, **31**(12), 1983–1989.

Smlo'yakov, E.R. (1995). Cooperative programmed differential games. *Differ. Uravn.*, **32**(11), 1499–1505, 1581; translation in *Differential Equations*, **32**(11) (1996), 1496–1502.

Smlo'yakov, E.R. (1997). Theorems on the existence of symmetric active equilibria in noncooperative programmed differential game. *Differ. Uravn.*, **33**(6), 776–781, 862; translation in *Differential Equations*, **33**(6) (1998), 779–784.

Smlo'yakov, E.R. (1998). Absolute equilibrium in differential game. *Dokl. Akad. Nauk*, **358**(5), 603–606.

Tarashina, S.I. (1998). Nash equilibria in differential pursuit game with one pursuer and m evaders. *Int. J. Math. Game Theory Algebra*, **7**(4), 189–197.

Zemskov, K.A., A.G. Pashkov (1997). Construction of optimal strategies in a differential pursuitevasion game with one pursuer and two evaders. *Prikl. Mat. Mekh.*, **61**(3), 403–412; translation in *J. Appl. Math. Mech.*, **61**(3), 391–399.

R. Matulevičius entered Vytautas Magnus University in 1994 . In 1998 he got the bachelor degree of the speciality of informatics (computer science) with business administration bases. 1998 he entered Vytautas Magnus University Informatics Faculty to the master studies and got his master diploma in 2000. His bachelor theses (1998): *Emulating states and states transition of processes in preemptive operating systems*, work of semester and his master theses about differential game models, simulations of economic bases (2000) – *Dynamical equilibrium research and application in duel problems using Internet*. Now he is a doctoral student at Norwegian University of Science and Technology (NTNU).

Dinaminės pusiausvyros tyrimas Dvikovos tipo uždaviniuose

Raimundas MATULEVIČIUS

Pradėjus nuo paprastesnio diferencialinio lošimo Dvikovos uždavinio buvo pereita prie sudėtingesnio ir praktiškesnio Ekonominės Dvikovos uždavinio modeliavimo. Pirmajame uždavinyje kalbama apie dviejų fizinių objektų dvikovą, antrasis uždavinys skirtas konkurencinio ekonominio modelio sukūrimui. Abiejuose uždaviniuose naudojamas Monte Karlo algoritmas. Fizinės dvikovos modelis iliustruoja diferencialinės žaidimų teorijos principus. Ekonominės Dvikovos uždavinys parodo rinkos dinamikos principus. Sprendimo metu matom, kad siekdamas išlikti rinkos lyderiu ir tapti monopolininku, objektas siekia kitų rinkos konkurentų bankrotų. Svarbiausias tokio veikiančio rinkoje objekto tikslas jo suminis laikotarpio pelnas, leidžiantis jam tapti monopolistu. Uždaviniuose naudojami skirtingi algoritmai, taip pat yra sukurta programinė įranga, leidžianti vykdyti uždavinius interneto aplinkoje.