International Journal of Pure and Applied Mathematics
Volume 60 No. 2 2010, 119-148

MARKETING COMPETITION IN THE MIDDLE OF PRODUCT LIFE-CYCLE: A DIFFERENTIAL GAME ANALYSIS

N.G. Medhin[1 §], Wei Wan[2]

[1]Department of Mathematics
North Carolina State University
Raleigh, NC 27695, USA
e-mail: ngmedhin@ncsu.edu

[2]Department of Mathematics
Claflin University
Orangeburg, SC 29115, USA
e-mail: wwan@claflin.edu

Abstract: The growth of market sale of a new product is followed by approximately stable market sales for some interval of time. The duration of this period depends on the kind of product. Because of the approximately constant sale volume, the basic dynamic of this stage is different from the first stage of product life cycle. In this paper a differential game model is used to get insight into the competition for one-product-market-share in the middle of the product life cycle in **n**-company competition environment. In the model, Lanchester competition dynamic will be adopted. And two kinds of competition strategies and their interactions will be considered. Both open-loop and closed-loop controls will be considered. Two algorithms are designed for the computation of the equilibrium, and numerical results are reported and discussed. Valuable results about the difference between open-loop and closed-loop control are obtained from the numerical results, which are beneficial for practical decisions.

AMS Subject Classification: 49K99, 49L99, 49M05, 49J21
Key Words: differential games, numerical methods, marketing competition

Received: January 27, 2010

§Correspondence author

1. Introduction

Differential Game originated in the work of R. Isaacs [20] in his research report from RAND Corporation around 1945. Berkovitz [1], [3] gives existence and optimality condition for differential game with fixed duration. Other typical theoretical investigation is from Avner Fridman [17]. After the existence and optimality condition is given, differential game is applied in many fields, such as economics, sociology, military, etc (see [24]). The first application in marketing is due to Lawrence Friedman [16], which is 2-dimensional differential game model, and numerical solution based on iterative method is given. In recent investigations of the application of differential game model in marketing, Gila E. Fruchter, Shlomo Kalish [19] focus on closed-loop control in a duopoly. Gary M. Ericson [13] does empirical analysis of closed-loop control, which shows closed-loop is better than open-loop control. In his another paper [14], he investigates a special method to solve closed-loop control, but his method can be successful only if the form of the objective function could be guessed. Prasad A. Naik [25] assumes there are more than two methods (controls) for each competitor, and investigates the interaction between controls. Other typical papers are as follows: Dockner and Jorgessen [11] study an open-loop competition based on a diffusion model. Dockner [12] investigates the link between pricing policy speed of diffusion, and number of competitors in the market. Teng, Jinn-Tsair, and Thompson [28] analyze numerically open-loop strategies in triopoly.

Above all, application of differential game models concentrates on the following aspects: the different effects brought by open-loop and closed-loop control; the methods of solving differential game models; the effects of different types of dynamics and objective functions on the equilibrium.

A general product has a typical product life cycle [26] (Figure 1). The length of each interval depends on the type of the product. Most of the references in marketing competition deal with the competition in the last stage of product life cycle. In this stage, the property of the dynamics of sale is that the sale decreases as time goes by. In Medhin and Wan [22], we discuss the competition in this stage and design a random algorithm to solve the differential game model, which is based on Quasi-Newton method. As to the first stage of product life cycle, there are a few references dealing with competition in this stage [11]. In this stage, the sale will increase as time goes by. In Medhin and Wan [23], the model deals with multiple-products competition in duopoly. We investigate the optimal competition strategy for each competitor in this stage. As to the second stage of product life cycle, however, there are few references investigating the

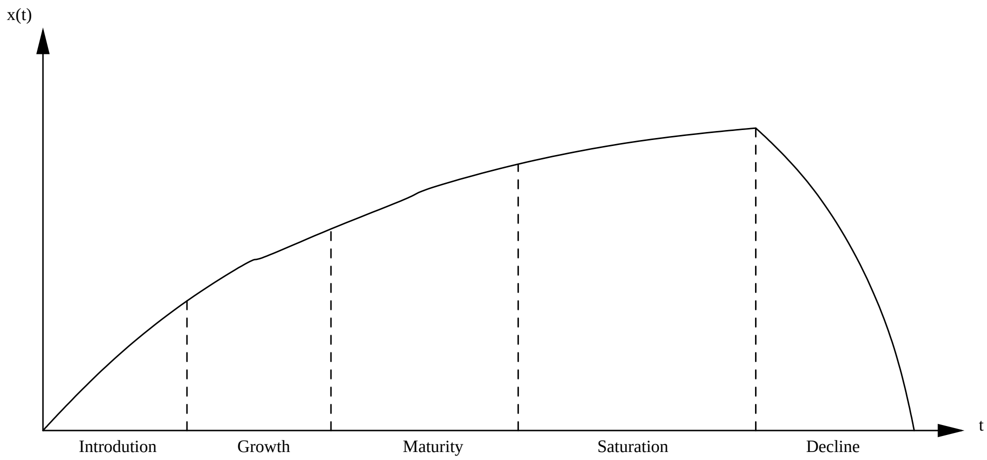

Figure 1: Sales vs time

competition strategy. The most important and different property of dynamics in the second phase of a product life cycle is that the market sale does not change much, and keeps at some approximate level. Furthermore, as to some products, such as TV, beverage market, etc., this stage is very long. Thus, it is necessary for us to investigate competition strategies for this specific stage. Thus, the results in this paper will fill this gap in marketing competition. In this paper we use differential game model to investigate the competition at the second stage. From our differential game model, we investigate the different effects brought by open-loop and closed-loop controls, the different interaction effects among different types of controls. Furthermore, we develop numerical algorithm to solve optimality conditions, and then solve the model. This algorithm is an efficient one for general differential game models. Furthermore, from the numerical results we obtain some general relations between these two kinds of controls, and practical guidelines for competition in this stage.

2. Model Development

In the investigation, in order to simplify the problem, we will assume that the total market sale is fixed. A constant total market share means that the loss of one player will be gains to others. This characteristic of the market will determine the special form of control (advertising, promotion) strategies. We will see these in the numerical results. Consider a one-product market where **n**-companies are involved. The market managers will use two kinds of controls: advertising and promotion to compete for market share and minimize cost. We use the index $i = 1, 2, \cdots, n$ to represent these **n** companies. The main notations are as follows:

$\mathbf{x}_i(t)$ – Market share of company i at time t. Since we are in the middle stage of product life-cycle, we impose $\sum_i^n x_i(t) = 1$;

$\mathbf{u}_i(t)$ – Control(advertising) of company i at time t;

$\mathbf{v}_i(t)$ – Control(promotion) of company i at time t;

$\mathbf{a}_i$ – Effectiveness of control-advertising of company i;

$\mathbf{b}_i$ – Effectiveness of control-promotion of company i;

$\mathbf{c}_i(t)$ – Interaction effectiveness of advertising and promotion;

δ_i – Advertising cost parameter for company i;

γ_i – Promotion cost parameter for company i;

ω_i – Company i's weight of final market share;

$\mathbf{p}$ – Price of this product.

In the second stage of the product life cycle, there is no diffusion effect and natural decrease effect, so we will use the Lanchester model to describe the dynamics of the system. The basic Lanchester model in two players situation is:

$$\dot{x} = \beta_1 u_1^{\alpha_1}(1 - x) - \beta_2 u_2^{\alpha_2} x.$$

This basic model tells us that the change of $\mathbf{x}$ is determined by two factors: one is positive effect, which is from the effect of player 1's own control $\mathbf{u}_1$ on the remaining market, the other is negative effect, which is from the effect of competitor's control $\mathbf{u}_2$ on the market. Now we extend this basic idea of Lanchester model into the n-player situation. The dynamics is the following:

$$\dot{x}_i = g_i(u_i, v_i)(1 - x_i) - x_i \sum_{k=1, k\neq i}^{n} g_k(u_k, v_k), \quad i = 1, \cdots, n,$$

where g_i is a function of (u_i, v_i). We interpret $g_i(u_i, v_i)$ as total effect of these two controls. We let $g_i(u_i(t), v_i(t)) = a_i u_i(t) + b_i u_i(t) + c_i(t) u_i(t) v_i(t)$. In Prasad A. Naik [25], he uses this form of g_i with $c_i(t)$ constant. In our model, we will first adopt this form with where $c_i(t)$ also depends on time, and then a more general form. The underlying assumption behind linear combination is that when company i does not employ control$(= 0)$, company i has no effects on the market. However, for real marketing competition, we should adopt another assumption, which is, when company i does not employ any control, it still keeps a basic share of the market. Thus, we take the form of $g_i(u_i(t), v_i(t)) = e^{a_i u_i(t)+b_i v_i(t)} + c_i(t)u_i(t)v_i(t)$. The non-linear combination will bring us difficulties to solve the model. However, regardless of the form of g_i, from the following discussion we can see that that our iterative algorithm is effective to solve differential game models. Let $G(u_1, u_2, \cdots, u_n, v_1, v_2, \cdots, v_n) = \sum_{i=1}^{n} g_i(u_i, v_i)$, then we get the following system:

$$\frac{d}{dt}\begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} g_1 \\ \vdots \\ g_n \end{bmatrix} - \begin{bmatrix} G & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & G \end{bmatrix}\begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}.$$

For performance, we will consider two criteria: one is profit, the other is market share at the final time, so the objective functions are of the form:

$$\min_{u_i,v_i} J_i = \int_{t_0}^{T} [\frac{\delta_i}{2}u_i^2(t) - (p - \frac{\gamma_i}{2}v_i^2(t))x_i(t)]dt - \omega_i \frac{x_i(T)}{\sum_{k=1}^{n} x_k(T)}.$$

Here, when calculating the revenue, we use the square of the controls. This is because there exists diminishing returns to the scale of adverting and promotion expenditures. This effect is included in squaring the control terms in the performance indices.

After the differential game model is set up, before we analyze the optimality condition and design numerical algorithm, we need to investigate the restriction on the coefficients in our system, since the sale of the product cannot be negative and the total market sale is constant. The following theorem is about the relationships between these coefficients.

Theorem 1. *Suppose* $0 \le u_i \le L$, $0 \le v_i \le K$, *and* $0 \le |c_i| \le M$, *then* $a_i L + b_i K \ge KLM, i = 1, 2, \cdots, n$ *guarantee that* $x_i(t) \ge 0$ *for* $0 \le t \le T, i = 1, 2, \cdots, n$.

Proof. From $\dot{x_i} = g_i - Gx_i$, we see that $\dot{x_i} = g_i$ if $x_i(t) = 0$.

In order to have $x_i(t) \ge 0$ for $0 \le t \le T$, the condition $g_i \ge 0$ should be satisfied. Then $g_i(u_i, v_i) = a_i u_i + b_i v_i + c_i(t)u_i v_i \ge 0$. Thus, since $a_i \ge 0, b_i \ge 0$, we have $\frac{a_i}{v_i} + \frac{b_i}{u_i} \ge -c_i(t)$. We need to enforce the condition: $\frac{a_i}{v_i} + \frac{b_i}{u_i} \ge |c_i(t)|$. Thus, suppose that $\inf_{v_i}\{\frac{a_i}{v_i}\} + \inf_{u_i}\{\frac{b_i}{u_i}\} \ge \sup|c_i(t)|$. Since $0 \le u_i \le L$, $0 \le v_i \le K$, and $0 \le |c_i| \le M$, the condition $\frac{a_i}{K} + \frac{b_i}{L} \ge M$, or $a_i L + b_i K \ge KLM, i = 1, 2, \cdots, n$ will guarantee that $x_i(t) \ge 0$. □

3. Model Analysis

We analyze the optimality condition for the above differential game mode, then design algorithm to solve it. First we consider open-loop controls. The controls are just function of time. The associated Hamiltonian for player i is:

$$H_i = \frac{\delta_i}{2} u_i^2 - (p - \frac{\gamma_i v_i^2}{2}) x_i + \sum_{k=1}^{n} \lambda_{ik}(g_k - G x_k).$$

The optimality condition implies that player i's control u_i, v_i will minimize H_i conditioning on other players' controls being optimal. Thus,

$$\frac{\partial H_i}{\partial u_i} = \delta_i u_i + (a_i + c_i v_i)[\lambda_{ii}(1 - x_i) - \sum_{k=1, k \neq i}^{n} \lambda_{ik} x_k] = 0$$

implies that

$$u_i = \frac{a_i + c_i v_i}{\delta_i}[\sum_{k=1, k \neq i}^{n} \lambda_{ik} x_k - \lambda_{ii}(1 - x_i)],$$

$$\frac{\partial H_i}{\partial v_i} = \gamma_i v_i x_i + (b_i + c_i u_i)[\lambda_{ii}(1 - x_i) - \sum_{k=1, k \neq i}^{n} \lambda_{ik} x_k] = 0$$

implies that

$$v_i = \frac{b_i + c_i u_i}{\gamma_i x_i}[\sum_{k=1, k \neq i}^{n} \lambda_{ik} x_k - \lambda_{ii}(1 - x_i)].$$

The co-state system for player i is drawn from $-\dot{\lambda_{ij}} = \frac{\partial H_i}{\partial x_j}$:

$$\frac{d}{dt}\begin{bmatrix} \lambda_{i1} \\ \vdots \\ \lambda_{ii} \\ \vdots \\ \lambda_{in} \end{bmatrix} = \begin{bmatrix} G & \cdots & 0 \\ \cdots & \cdots & \cdots \\ 0 & G & 0 \\ \cdots & \cdots & \cdots \\ 0 & \cdots & G \end{bmatrix} \begin{bmatrix} \lambda_{11} \\ \vdots \\ \lambda_{ii} \\ \vdots \\ \lambda_{in} \end{bmatrix} + \begin{bmatrix} 0 \\ \vdots \\ (p - \frac{\gamma_i}{2} v_i^2) \\ \vdots \\ 0 \end{bmatrix}.$$

The terminal condition is:

$$\begin{bmatrix} \lambda_{i1}(T) \\ \vdots \\ \lambda_{ii}(T) \\ \vdots \\ \lambda_{in}(T) \end{bmatrix} = \begin{bmatrix} \omega_i \frac{x_i(T)}{(\sum_{k=1}^{n} x_k(T))^2} \\ \vdots \\ -\omega_i \frac{\sum_{k=1, k \neq i}^{n} x_k(T)}{(\sum_{k=1}^{n} x_k(T))^2} \\ \vdots \\ \omega_i \frac{x_i(T)}{(\sum_{k=1}^{n} x_k(T))^2} \end{bmatrix}.$$

Thus, we get a system of Differential-Algebraic Equations (DAE) with n^2+n equations.

$$
\begin{aligned}
\dot{x_1} =&(a_1u_1 + b_1v_1 + c_1u_1v_1) - x_1\sum_{k=1}^{n}(a_ku_k + b_kv_k + c_ku_kv_k),\\
&\vdots\\
\dot{x_i} =&(a_iu_i + b_iv_i + c_iu_iv_i) - x_i\sum_{k=1}^{n}(a_ku_k + b_kv_k + c_ku_kv_k),\\
&\vdots\\
\dot{x_n} =&(a_nu_n + b_nv_n + c_nu_nv_n) - x_n\sum_{k=1}^{n}(a_ku_k + b_kv_k + c_ku_kv_k),\\
\dot{\lambda_{i1}} =&\lambda_{i1}\sum_{k=1}^{n}(a_ku_k + b_kv_k + c_ku_kv_k),\\
&\vdots\\
\dot{\lambda_{ii}} =&\lambda_{ii}\sum_{k=1}^{n}(a_ku_k + b_kv_k + c_ku_kv_k) + (p - \frac{\gamma_i}{2}v_i^2),\\
&\vdots\\
\dot{\lambda_{in}} =&\lambda_{in}\sum_{k=1}^{n}(a_ku_k + b_kv_k + c_ku_kv_k),\\
0 =&\delta_iu_i + (a_i + c_iv_i)[\lambda_{ii}(1 - x_i) - \sum_{k=1,k\neq i}^{n}\lambda_{ik}x_k],\\
0 =&\gamma_iv_ix_i + (b_i + c_iu_i)[\lambda_{ii}(1 - x_i) - \sum_{k=1,k\neq i}^{n}\lambda_{ik}x_k],\\
x_i(0) =&x_{i0},\\
\lambda_{ij}(T) =&\omega_i\frac{x_i(T)}{(\sum_{k=1}^{n}x_k(T))^2},\\
\lambda_{ii}(T) =&-\omega_i\frac{\sum_{k=1,k\neq i}^{n}x_k(T)}{(\sum_{k=1}^{n}x_k(T))^2},\\
j \neq i,& \qquad i = 1, 2, \cdots, n.
\end{aligned}
$$

Second, we consider the optimality condition for closed-loop control situation. In a closed-loop differential game, the control is a function of the states, i.e. $u = u(t, x(t))$. As to player i, his optimal controls u_i, v_i are as follows:

$$u_i = \frac{a_i + c_i v_i}{\delta_i}[\sum_{k=1,k\neq i}^{n} \lambda_{ik} x_k - \lambda_{ii}(1 - x_i)],$$
$$v_i = \frac{b_i + c_i u_i}{\gamma_i x_i}[\sum_{k=1,k\neq i}^{n} \lambda_{ik} x_k - \lambda_{ii}(1 - x_i)].$$

The state equations for closed-loop case is the same as the open-loop case. The co-state system for player i is as follows:

$$-\dot{\lambda}_{i1} = \frac{\partial H_i}{\partial x_1} + \sum_{j\neq i} \frac{\partial H_i}{\partial u_j}\frac{\partial u_j}{\partial x_1} + \sum_{j\neq i} \frac{\partial H_i}{\partial v_j}\frac{\partial v_j}{\partial x_1},$$
$$\vdots$$
$$-\dot{\lambda}_{ii} = \frac{\partial H_i}{\partial x_i} + \sum_{j\neq i} \frac{\partial H_i}{\partial u_j}\frac{\partial u_j}{\partial x_i} + \sum_{j\neq i} \frac{\partial H_i}{\partial v_j}\frac{\partial v_j}{\partial x_i},$$
$$\vdots$$
$$-\dot{\lambda}_{in} = \frac{\partial H_n}{\partial x_n} + \sum_{j\neq i} \frac{\partial H_i}{\partial u_j}\frac{\partial u_j}{\partial x_n} + \sum_{j\neq i} \frac{\partial H_i}{\partial v_j}\frac{\partial v_j}{\partial x_n},$$

where

$$\frac{\partial H_i}{\partial x_i} = -(p - \frac{\gamma_i}{2} v_i) - \lambda_{ii} G,$$
$$\frac{\partial H_i}{\partial x_j} = -\lambda_{ij} G, \quad \text{for} \quad j \neq i,$$
$$\frac{\partial H_i}{\partial u_j} = (a_j + c_j v_j)[\lambda_{ij}(1 - x_j) - \sum_{k\neq j} \lambda_{ik} x_k],$$
$$\frac{\partial H_i}{\partial v_j} = (b_j + c_j u_j)[\lambda_{ij}(1 - x_j) - \sum_{k\neq j} \lambda_{ik} x_k],$$
$$\delta_i \frac{\partial u_i}{\partial x_i} + c_i[\lambda_{ii}(1 - x_i) - \sum_{k\neq i} \lambda_{ik} x_k]\frac{\partial v_i}{\partial x_i} = \lambda_{ii}(a_i + c_i v_i),$$
$$c_i[\lambda_{ii}(1 - x_i) - \sum_{k\neq i} \lambda_{ik} x_k]\frac{\partial u_i}{\partial x_i} + \gamma_i x_i \frac{\partial v_i}{\partial x_i} = \lambda_{ii}(b_i + c_i u_i) - \gamma_i v_i,$$

$$\delta_i \frac{\partial u_i}{\partial x_j} + c_i[\lambda_{ii}(1-x_i) - \sum_{k \neq i} \lambda_{ik} x_k] \frac{\partial v_i}{\partial x_j} = \lambda_{ij}(a_i + c_i v_i),$$

$$c_i[\lambda_{ii}(1-x_i) - \sum_{k \neq i} \lambda_{ik} x_k] \frac{\partial u_i}{\partial x_j} + \gamma_i x_i \frac{\partial v_i}{\partial x_j} = \lambda_{ij}(b_i + c_i u_i).$$

4. Algorithm

We design two algorithms to solve the above model. The first algorithm is based on solving Boundary Value Problems (BVP), which is a second-order convergence algorithm. The second algorithm is based on full discretization, and has first-order convergence.

For the above model, we can solve for controls u, v explicitly. After giving u, v in terms of the state and costate variables, we put the controls back into the state and costate equations to get the following Two Point Boundary Value Problems (BVP):

$$\begin{cases} \dot{x}_i(t) & = a(x_i(t), m_{ij}(t), t), \\ \dot{m}_{ij}(t) & = d(x_i(t), m_{ij}(t), t). \end{cases}$$

The boundary condition is:

$$\begin{cases} x_i(0) & = x_{i0}, \\ m_{ij}(T) & = h_{ij}(x_1(T), \cdots, x_n(T)). \end{cases}$$

As to the above BVP, we choose to guess the terminal values since we just need to guess n terminal values, which is much less than guessing the initial values; furthermore, the objective function of each player can give information about terminal value. For example, if a player wants more final market share, which leads to a bigger coefficient for final market share in the objective function, then we can expect his terminal value should be bigger than the initial value. Thus, the initial value problem (IVP) associated with this BVP is

$$\begin{cases} \dot{x}_i(t) & = a(x_i(t), m_{ij}(t), t), \\ \dot{m}_{ij}(t) & = d(x_i(t), m_{ij}(t), t). \end{cases}$$

The terminal condition is:

$$\begin{cases} x_i(T) & = s, \\ m_{ij}(T) & = h_{ij}(x_1(T), \cdots, x_n(T)), \end{cases}$$

where $\mathbf{s}$ is a parameter column. In Medhin and Wan (see [22]), we gave the condition for existence of solution the above BVP. The above model satisfies the existence condition in [22]. For each $s \in R^n$ there is a unique solution of the above IVP, which is denoted by $x_i(t;s), m_{ij}(t;s)$, then there is a unique $x_i(0;s)$. Thus, we can define a functional relationship between $x_i(T)$ and $x_i(0)$, that is, $x_i(0) = f_i(x_i(T))$, for which we cannot, in general, find analytic formula. Now suppose $x_i(t;s^*), m_{ij}(t;s^*)$ are solved for from the IVP where $s = s^*$, and if s^* is such that the boundary conditions are satisfied, that is $x_i(0;s^*) = x_{i0}$ or $f_i(x_i(T;s^*)) - x_{i0} = 0$, then $x_i(t;s^*)$ is the solution of the BVP. The first algorithm is designed to find such s^*, that is to find solution for following system:

$$\begin{cases} F_1(x_1(T),\cdots,x_n(T)) = & f_1(x_1(T),\cdots,x_n(T)) - x_1(0) = 0, \\ & \vdots \\ F_i(x_1(T),\cdots,x_n(T)) = & f_i(x_1(T),\cdots,x_n(T)) - x_i(0) = 0, \\ & \vdots \\ F_n(x_1(T),\cdots,x_n(T)) = & f_n(x_1(T),\cdots,x_n(T)) - x_n(0) = 0. \end{cases}$$

The first algorithm is based on Newton's method to update $x_i(T)$. As to the above functions F_i, using Taylor series expansion up to first-order about some estimate point $x_1(T)^{(k)},\cdots,x_n(T)^{(k)}$, we have the system of equations:

$$\begin{bmatrix} F_1(x_1(T),\cdots,x_n(T)) \\ \cdots \\ F_i(x_1(T),\cdots,x_n(T)) \\ \cdots \\ F_n(x_1(T),\cdots,x_n(T)) \end{bmatrix} = \begin{bmatrix} F_1(x_1(T)^{(k)},\cdots,x_n(T)^{(k)}) \\ \cdots \\ F_i(x_1(T)^{(k)},\cdots,x_n(T)^{(k)}) \\ \cdots \\ F_n(x_1(T)^{(k)},\cdots,x_n(T)^{(k)}) \end{bmatrix}$$

$$+ \begin{bmatrix} \frac{\partial F_1}{\partial x_1(T)} & \cdots & \frac{\partial F_1}{\partial x_n(T)} \\ \cdots & \cdots & \cdots \\ \frac{\partial F_i}{\partial x_1(T)} & \cdots & \frac{\partial F_i}{\partial x_n(T)} \\ \cdots & \cdots & \cdots \\ \frac{\partial F_n}{\partial x_1(T)} & \cdots & \frac{\partial F_n}{\partial x_n(T)} \end{bmatrix}_P * \begin{bmatrix} x_1(T) - x_1(T)^{(k)} \\ \cdots \\ x_i(T) - x_i(T)^{(k)} \\ \cdots \\ x_n(T) - x_n(T)^{(k)} \end{bmatrix} = \begin{bmatrix} 0 \\ \cdots \\ 0 \\ \cdots \\ 0 \end{bmatrix},$$

which we solve for $(x_1(T),\cdots,x_n(T))$ to get the updated vector estimate:

$$\begin{bmatrix} x_1(T)^{(k+1)} \\ \cdots \\ x_i(T)^{(k+1)} \\ \cdots \\ x_n(T)^{(k+1)}) \end{bmatrix} = \begin{bmatrix} x_1(T)^{(k)} \\ \cdots \\ x_i(T)^{(k)} \\ \cdots \\ x_n(T)^{(k)}) \end{bmatrix}$$

$$-\begin{bmatrix} \frac{\partial F_1}{\partial x_1(T)} & \cdots & \frac{\partial F_1}{\partial x_n(T)} \\ \cdots & \cdots & \cdots \\ \frac{\partial F_i}{\partial x_1(T)} & \cdots & \frac{\partial F_i}{\partial x_n(T)} \\ \cdots & \cdots & \cdots \\ \frac{\partial F_n}{\partial x_1(T)} & \cdots & \frac{\partial F_n}{\partial x_n(T)} \end{bmatrix}_P * \begin{bmatrix} F_1(x_1(T)^{(k)}, \cdots, x_n(T)^{(k)}) \\ \cdots \\ F_i(x_1(T)^{(k)}, \cdots, x_n(T)^{(k)}) \\ \cdots \\ F_n(x_1(T)^{(k)}, \cdots, x_n(T)^{(k)}) \end{bmatrix},$$

that is,

$$\mathbf{X}^{(k+1)}(T) = X^{(k)}(T) - J_k^{-1} F(X^{(k)}(T)),$$

where $P = (x_1(T)^{(k)}, \cdots, x_n(T)^{(k)})$.

In this updating process, we have the following issues:

1. How do we get the initial guess $x_i(T)^{(0)}$? Only when $|x_i(T)^{(0)} - x_i^*(T)|$ is small enough, is there quadratic convergence.

2. How do we get the *Jacobian* matrix of F? Although we know that the Jacobian matrix exists, we do not have analytic form for it. This is the case if the system is complex.

As to the initial guess, we can use information from initial values and the players' preference in the objectives to guess it. For example, we can expect that the terminal value for this player will be a little bigger than initial value if he prefers more market share than others.

To obtain the *Jacobian matrix* J_k, the following recipe is one way to approximate it:

$$\frac{\partial F_i}{\partial x_j} \simeq \frac{F_i(\mathbf{x}(\mathbf{k})) - F_i(\mathbf{x}(\mathbf{k-1}))}{x_j(k) - x_j(k-1)}.$$

From the above expression, we use difference of values of F_i between step k and step $k-1$ to approximate ΔF_i; use difference of values of x_j between step k and step $k-1$ to approximate Δx_j.

The following procedure is used in the first algorithm:

Step 1. First guess two terminal values for state variables $x_i(T), i = 1, 2, \cdots, n$.

Step 2. Compute corresponding costate vectors at time T. Since we have explicit formulae giving the controls in terms of the state and costate variables, we proceed to solve the coupled state and costate equations backwards in time from T to 0.

Step 3. Approximate Jacobian matrix J_k, and use Newton method to get new approximation x_{k+1}

Step 4. Check stopping criteria. If it is not satisfied, go back to Step 3.

The following algorithm is set up using this process.

Algorithm 1. 1. Guess two terminal values for state variables: $\widetilde{\mathbf{x}}_{\mathbf{0}}(\mathbf{T})$ and $\widetilde{\mathbf{x}}_{\mathbf{1}}(\mathbf{T})$, which are $n-$ dimensional vectors.

2. Compute terminal values for the costate variables: $\lambda_{\mathbf{0}}(\mathbf{T})$ and $\lambda_{\mathbf{1}}(\mathbf{T})$ corresponding to $\widetilde{\mathbf{x}}_{\mathbf{0}}(\mathbf{T})$ and $\widetilde{\mathbf{x}}_{\mathbf{1}}(\mathbf{T})$.

3. Solve backward the state and costate equations by Runge-Kutta method and get $\widetilde{\mathbf{x}}_{\mathbf{1}}(\mathbf{0}), \widetilde{\mathbf{x}}_{\mathbf{2}}(\mathbf{0})$, then compute $\mathbf{F}_{\mathbf{0}}, \mathbf{F}_{\mathbf{1}}$.

4. If $\| \mathbf{F}_{\mathbf{0}} \| < \epsilon$ or $\| \mathbf{F}_{\mathbf{1}} \| < \epsilon$, then stop, output the optimal solution; if not, $k = 1$ go to Step 5.

5. Update $\widetilde{\mathbf{x}}_{\mathbf{k+1}}(\mathbf{T}) = \widetilde{\mathbf{x}}_{\mathbf{k}}(\mathbf{T}) - J_k^{-1}\mathbf{F}_{\mathbf{0}}(\widetilde{\mathbf{x}}_{\mathbf{k}}(\mathbf{T}))$, where

$$J_k = \begin{bmatrix} \frac{\partial F_1}{\partial x_1} & \cdots & \frac{\partial F_1}{\partial x_n} \\ \cdots & \cdots & \cdots \\ \frac{\partial F_n}{\partial x_1} & \cdots & \frac{\partial F_n}{\partial x_n} \end{bmatrix} \quad \text{and} \quad \frac{\partial F_i}{\partial x_j} \simeq \frac{F_i(\widetilde{\mathbf{x}}(\mathbf{k})) - F_i(\widetilde{\mathbf{x}}(\mathbf{k-1}))}{\widetilde{x}_j(k) - \widetilde{x}_j(k-1)}.$$

.

6. Compute terminal values for costate variables: $\lambda(\mathbf{T})$.

7. Solve backward the state and costate equations by Runge-Kutta method and get $\widetilde{\mathbf{x}}_{\mathbf{k+1}}(\mathbf{0})$, then compute $\mathbf{F}_{\mathbf{0}}$.

8. If $\| \mathbf{F}_{\mathbf{0}} \| < \epsilon$, then stop, output the optimal solution; if not, $k+1$, go to Step 5.

We design another iterative algorithms to solve the differential game model. The idea of the iterative algorithm is based on Medhin and Wan (see [23]). The process is as follows: first generating **n** controls for **n** players separately; second, solve the state equation forward; third, solve for $\frac{\partial u_i}{\partial x_j}, \frac{\partial v_i}{\partial x_j}, \frac{\partial H_i}{\partial x_j}, \frac{\partial H_i}{\partial u_j}$ for each mesh point; fourth, solve the costate system backward, then evaluate the objective values. Fifth, if stopping criteria are satisfied, then stop. Otherwise, each player updates his control by steepest descent direction of his own Hamiltonian, then repeat from the second step. This method has first order convergence, but the advantage is that we do not need to guess the initial condition for the state variable. Furthermore, this method can be used when the controls cannot be solved for explicitly, which is especially useful for closed-loop control case, since we can use discretized control and state variable values to solve for $\frac{\partial u_i}{\partial x_j}, \frac{\partial v_i}{\partial x_j}$. The following algorithm is set up using this process.

Algorithm 2. 1. Generate randomly a discrete approximation to control

$u_1(t), \cdots, u_n(t),\ v_1(t), \cdots, v_n(t), t \in [0, T]$, that is:

$$\begin{aligned} u_1(t) =& \quad u_1(t_k), \quad t \in [t_k, t_{k+1}), \quad k = 1, 2, \cdots, N, \\ &\vdots \\ u_n(t) =& \quad u_n(t_k), \quad t \in [t_k, t_{k+1}), \quad k = 1, 2, \cdots, N, \\ v_1(t) =& \quad v_1(t_k), \quad t \in [t_k, t_{k+1}), \quad k = 1, 2, \cdots, N, \\ &\vdots \\ v_n(t) =& \quad v_n(t_k), \quad t \in [t_k, t_{k+1}), \quad k = 1, 2, \cdots, N. \end{aligned}$$

2. Integrate the state equation from 0 to T with initial condition $x_i(0) = x_{i0}, i = 1, \cdots, n$.

3. Calculate $\lambda_{ij}(T), i = 1, \cdots, n$ using $x_i(T), i = 1, \cdots, n$ and integrate the costate equation backward.

4. Solve for $\frac{\partial u_i}{\partial x_j}, \frac{\partial v_i}{\partial x_j}, \frac{\partial H_i}{\partial x_j}, \frac{\partial H_i}{\partial u_j}$ for each interval $[t_k, t_{k+1})$.

5. Use the discrete value of state and costate variables to evaluate objective values: $J_i(0), i = 1, \cdots, n$.

6. Generate a new piecewise constant control given by

$$\begin{cases} u_1(t_{k+1}) = & u_1(t_k) - \tau_1 \frac{\partial H_1}{\partial u_1}(t_k), \quad k = 1, 2, \cdots, N, \\ \quad\cdots & \cdots \\ u_n(t_{k+1}) = & u_n(t_k) - \tau_n \frac{\partial H_n}{\partial u_n}(t_k), \quad k = 1, 2, \cdots, N, \\ v_1(t_{k+1}) = & v_1(t_k) - \delta_1 \frac{\partial H_1}{\partial v_1}(t_k), \quad k = 1, 2, \cdots, N, \\ \quad\cdots & \cdots \\ v_n(t_{k+1}) = & v_n(t_k) - \delta_n \frac{\partial H_n}{\partial v_n}(t_k), \quad k = 1, 2, \cdots, N, \end{cases}$$

where step length τ_i, δ_i will be chosen to decrease $H_i, i = 1, \cdots, n$.

7. Use updated controls to repeat Step 2 to Step 5 to get $J_i(1), i = 1, \cdots, n$.

8. If $|J_i(k) - J_i(k-1)| < \epsilon$, for $i = 1, \cdots, n$, then terminate the iterative procedure and output the optimal controls and state equations.

If the stopping criterion is not satisfied, generate a new piecewise constant control given by

$$\begin{cases} u_1(t_{k+1}) = & u_1(t_k) - \tau_1 \frac{\partial H_1}{\partial u_1}(t_k), \quad k = 1, 2, \cdots, N, \\ \quad\cdots & \cdots \\ u_n(t_{k+1}) = & u_n(t_k) - \tau_n \frac{\partial H_n}{\partial u_n}(t_k), \quad k = 1, 2, \cdots, N, \\ v_1(t_{k+1}) = & v_1(t_k) - \delta_1 \frac{\partial H_1}{\partial v_1}(t_k), \quad k = 1, 2, \cdots, N, \\ \quad\cdots & \cdots \\ v_n(t_{k+1}) = & v_n(t_k) - \delta_n \frac{\partial H_n}{\partial v_n}(t_k), \quad k = 1, 2, \cdots, N, \end{cases}$$

where step length τ_i, δ_i will be chosen to decrease $H_i, i = 1, \cdots, n$. Then go back to Step 7.

5. Numerical Results

In our experiment, there are two companies. The basic assumption is company 2 is stronger than company 1, which means the effectiveness of control of company 2 is bigger than that of company 1, that is, $a_1 < a_2, b_1 < b_2$. Then, in the differential game between these two companies, the coefficients ω_i, initial conditions $x_i(0)$ and $c_i(t)$ will affect the equilibrium. In the following, we will discuss the change of equilibrium with regards to these coefficients in both open open- and closed-loop controls.

5.1. Effects in Terms of ω_i and Initial Conditions

Scenario 1. We assume that

$$x_1(0) > x_2(0), \quad \omega_1 = \omega_2, \quad \delta_1 > \delta_2, \quad \gamma_1 > \gamma_2.$$

This means the small company has larger market share at the initial time, and both companies have the same preference for final market share. Figures 2 is numerical results for open-loop case. Figures 3 is numerical results for closed-loop case. Figures 4 is comparison between open and closed-loop. From Figure 4 we can see closed-loop control is always smaller than open-loop control although the difference decreases as time goes on. Furthermore, from Table 1 the objective value of the closed-loop control setup is better than that of the open-loop control setup.

Because the advertising and promotion costs obey $\delta_1 > \delta_2, \gamma_1 > \gamma_2$, the bigger company 2 will be able to use bigger controls-advertising, promotion to attract customers. Also because $a_1 < a_2$, $b_1 < b_2$, customers are sensitive to the bigger company's controls and are willing to stay with the bigger company. This can explain why company 2 uses big controls to attract customers and then decrease its controls quickly. However, for company 1, its control trajectories are concave. Thus it has to employ relatively bigger control first and decrease its controls following the bigger company's steps.

Scenario 2. We assume that

$$x_1(0) < x_2(0), \quad \omega_1 > \omega_2, \quad \delta_1 > \delta_2, \quad \gamma_1 > \gamma_2.$$

From the above data, we can see that the small company 1 has smaller market share at the initial time, and company 1 prefers final market share to

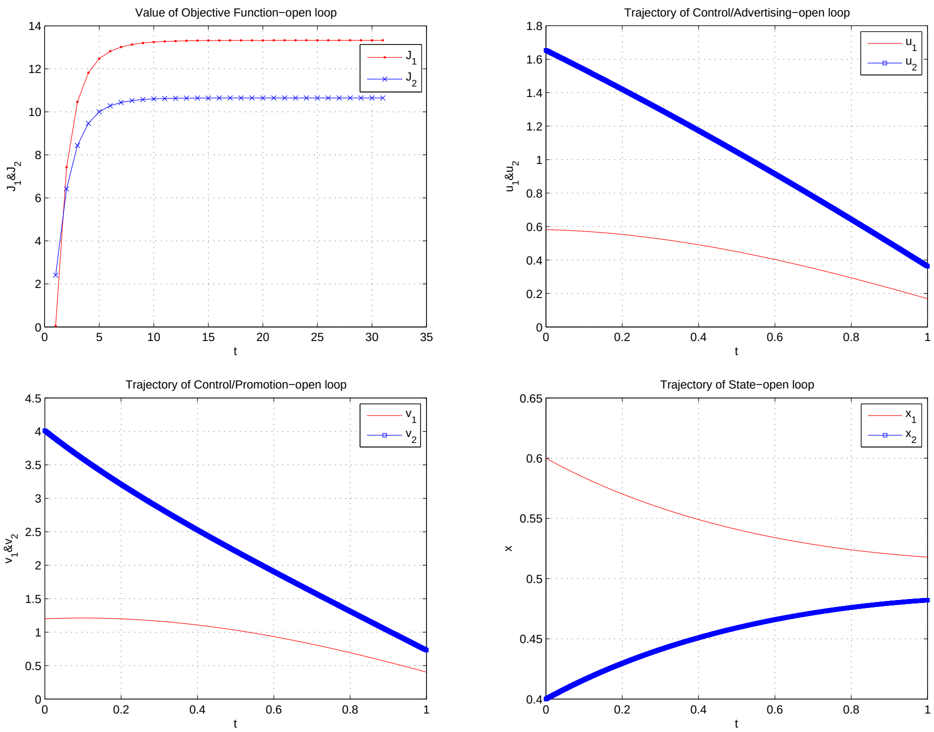

Figure 2: $\omega_1 = \omega_2$, open-loop

	J_1	J_2	X_1	X_2
Open-loop	13.3232	10.6441	$\cdots$	$\cdots$
Close-loop	13.3738	10.6890	$\cdots$	$\cdots$
Difference$^{L^2-norm}$	0.0505	0.0449	0.0087	0.0087

Table 1: $\omega_1 = \omega_2$

profit. Figures 5 is numerical results for open-loop case. Figures 6 is numerical results for closed-loop case. Figures 7 is comparison between open and closed-loop. State trajectories are still almost the same between open and closed-loop with the difference $0.0068(L_2 - norm)$. We can still see there is little difference between open- and closed-loop controls, and open-loop controls are still always bigger than the closed-loop controls, and the difference decreases as time goes

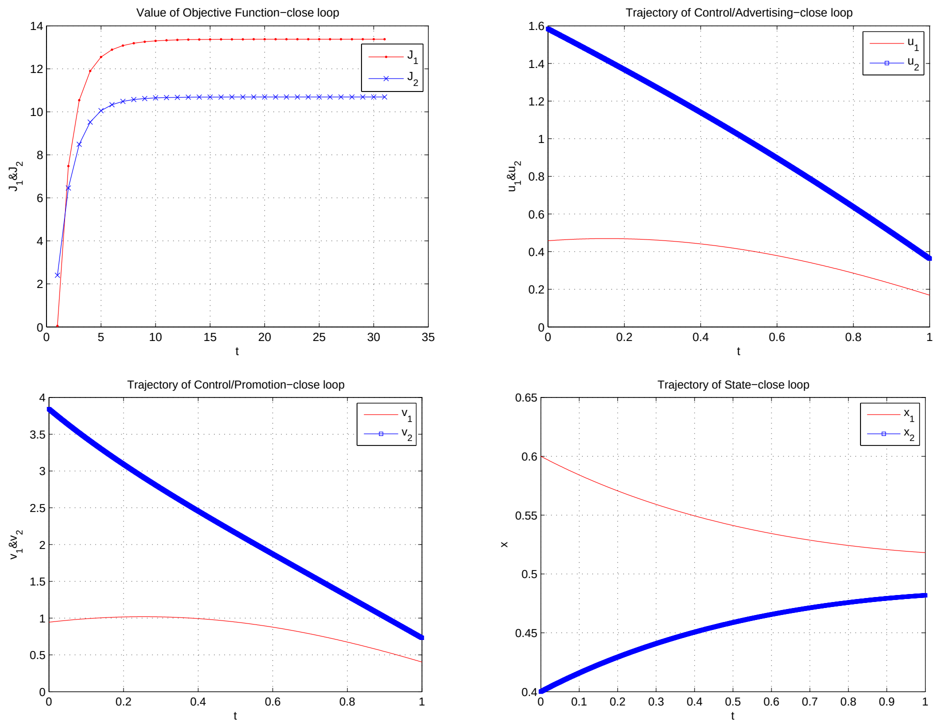

Figure 3: $\omega_1 = \omega_2$, closed-loop

	J_1	J_2	X_1	X_2
Open-loop	9.1280	13.4932	$\cdots$	$\cdots$
Close-loop	9.1734	13.5295	$\cdots$	$\cdots$
Difference$^{L^2-norm}$	0.0453	0.0362	0.0068	0.0068

Table 2: $\omega_1 > \omega_2$

on. Further, the objective value of closed-loop controls setup is still better than that of open-loop controls setup.

The controls-advertising, promotion of company 1 are much bigger than company 2 just because company 1 wants bigger final market share but it has smaller initial market share. Thus, company 1 needs to pay for his goal, which means that its optimal cost is higher than company 2's.

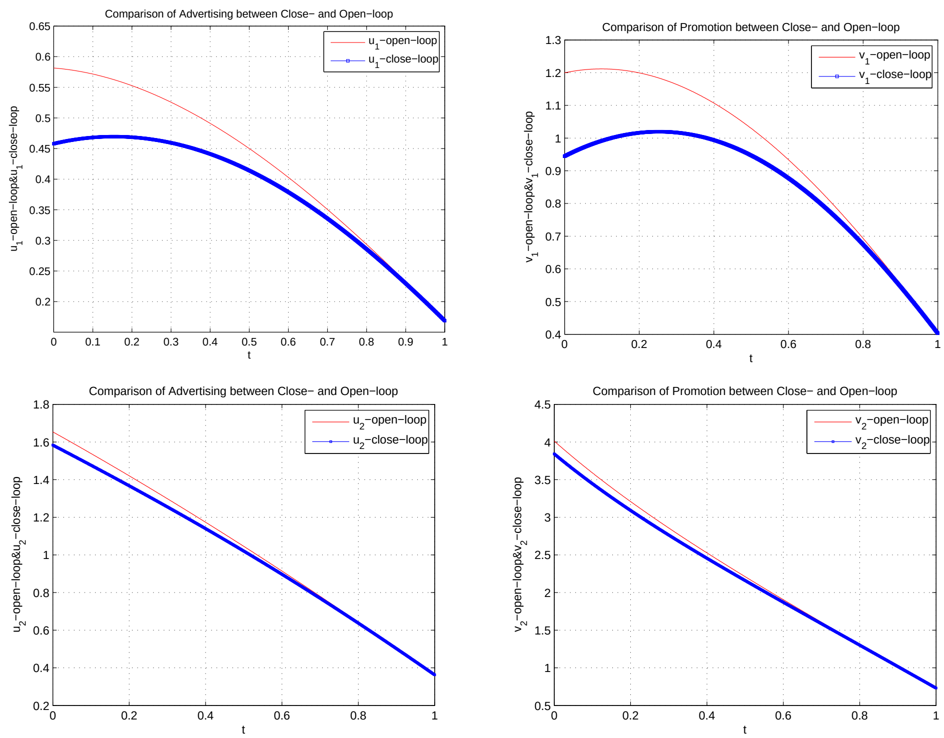

Figure 4: $\omega_1 = \omega_2$, comparison between open and closed-loop

Conclusion. From the numerical results, we can see that if a company wants more final market share, then it will use bigger control to to attract customers. Furthermore, if its initial market share is relatively smaller, then its controls will be even bigger. The effectiveness of closed-loop control is better than open-loop since decision maker can use updated state information to make decision.

5.2. Effect of Interaction of Two Controls

Scenario 3. In previous scenarios, we assumed that the effect of interaction of two controls of these two companies are the same, that is $c_1 = c_2$. In this scenario, we assume that $c_1 > c_2$, other coefficients are the same as Scenario 2, which means that player 1's controls will have bigger effect to attract customers.

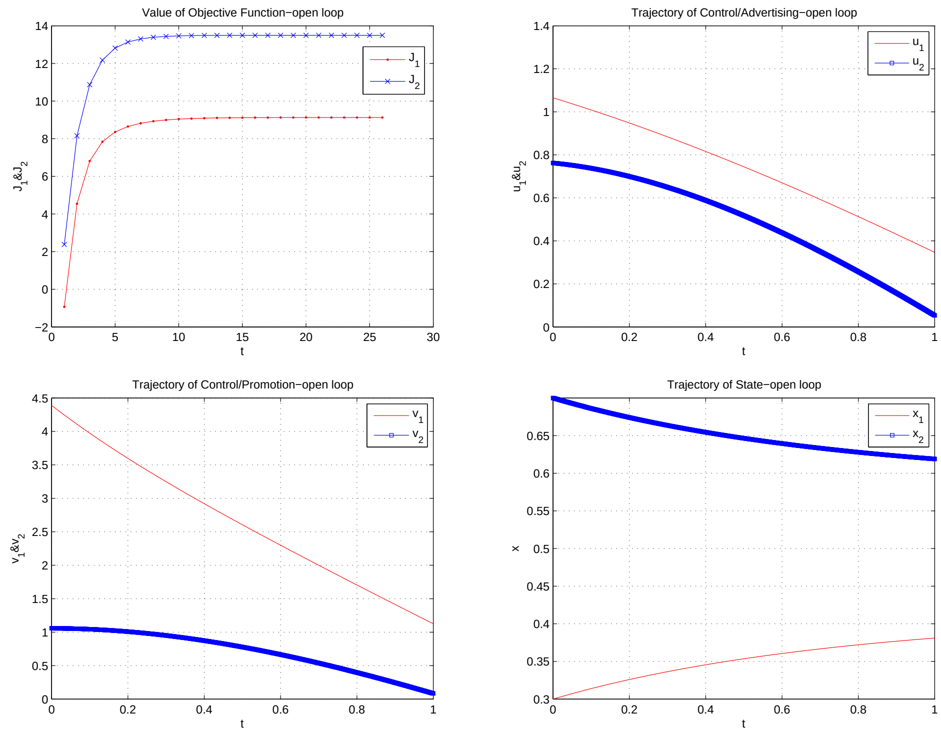

Figure 5: $\omega_1 > \omega_2$, open-loop

By comparing control trajectories in Figure 5 and Figure 8, player 1's control-promotion has a little change, but his control(advertising) increases most of the time. This is reasonable, because player 1 wants more final market share, the extra cost brought by increased control can be offset by the gain obtained from the final market share. However, as to player 2, both his controls just have a little change. By comparing objective function values in Table 3, we can see that player 1's objective function value becomes better, but player 2's becomes a little worse. This still make sense because the bigger interaction of two controls of player 1 will lead to more effective controls. Thus, we can conclude that the increase of effect of interaction of one player will lead to increase of his own advertising strategy, and improving his objective function value, but have little effect on promotion strategy. His competitor's controls have a little change, and objective function's value will become a littler worse.

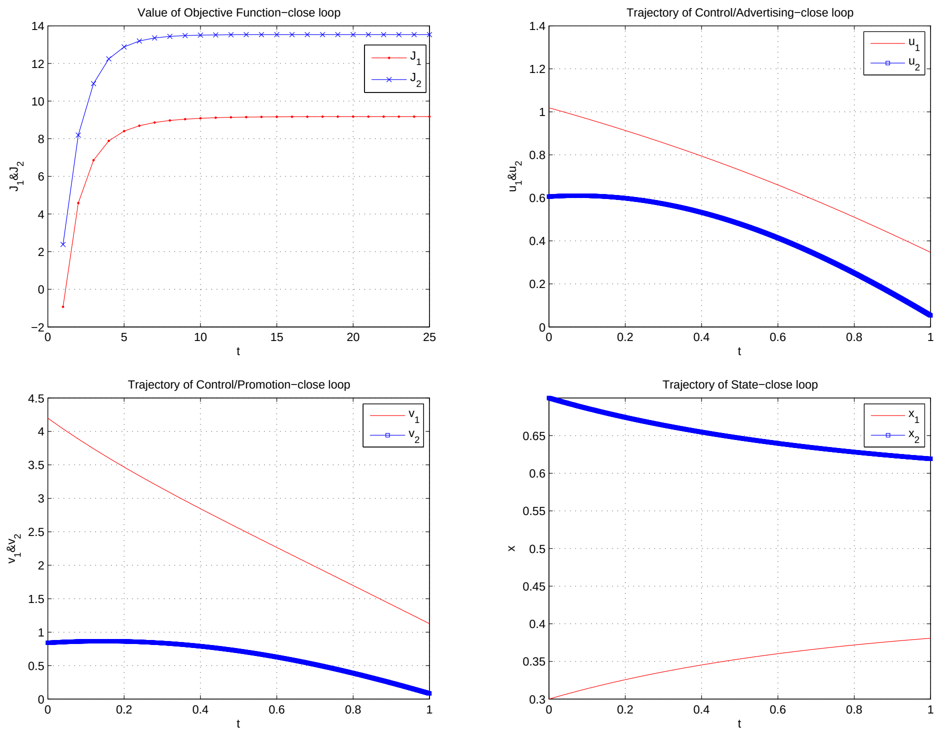

Figure 6: $\omega_1 > \omega_2$, closed-loop

	$J_1(open)$	$J_2(open)$	$J_1(close)$	$J_2(close)$
Scenario 2	18.3237	7.0478	18.3886	7.0855
Scenario 3	18.3828	6.9457	18.3911	7.0405
Difference(3 - 2)	0.0590	-0.1021	0.0025	-0.0449

Table 3: Comparison of Scenarios 2 and 3

Scenario 4. We increase the effect of interaction of player 1 in Scenario 4 further from that of Scenario 3. The results in Figure 8 and Figure 9 follow the conclusion of Scenario 3: player 1's control-promotion has a little change, and player 1 increases control-advertising further more, and his objective function's value is improved further too; For player 2, there is little change in his both controls, and there is no change to his objective function value. So we strengthen

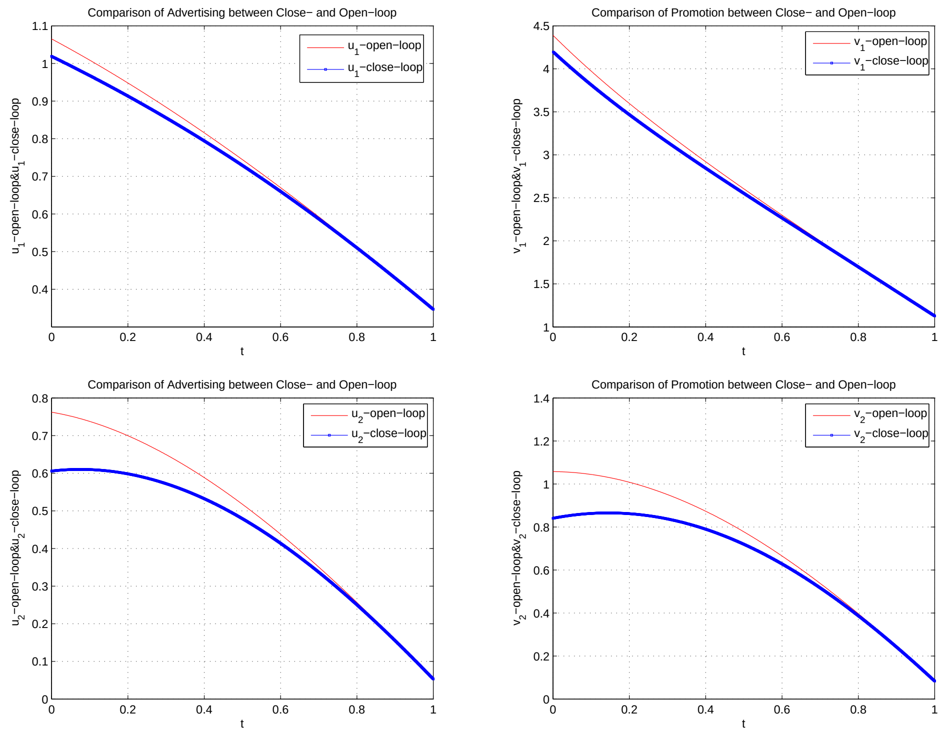

Figure 7: $\omega_1 > \omega_2$, comparison between open and closed-loop

the conclusion of Scenario 3.

Conclusion. From the numerical results, the increasing of effect of interaction of two controls of each player will lead to increasing of his controls, which will bring more profits to him. However, this does not affect the other player's controls much, but will hurt his objective value.

5.3. Time Dependent Control Interaction Coefficient

In all the previous scenarios, the interaction effectiveness of controls $(c_i(t))$ are constant. Now we will consider the cases when the $c_i(t)$ is function of time. In the following Scenarios 5 and 6, we set up two typical situations and discuss the change of equilibrium.

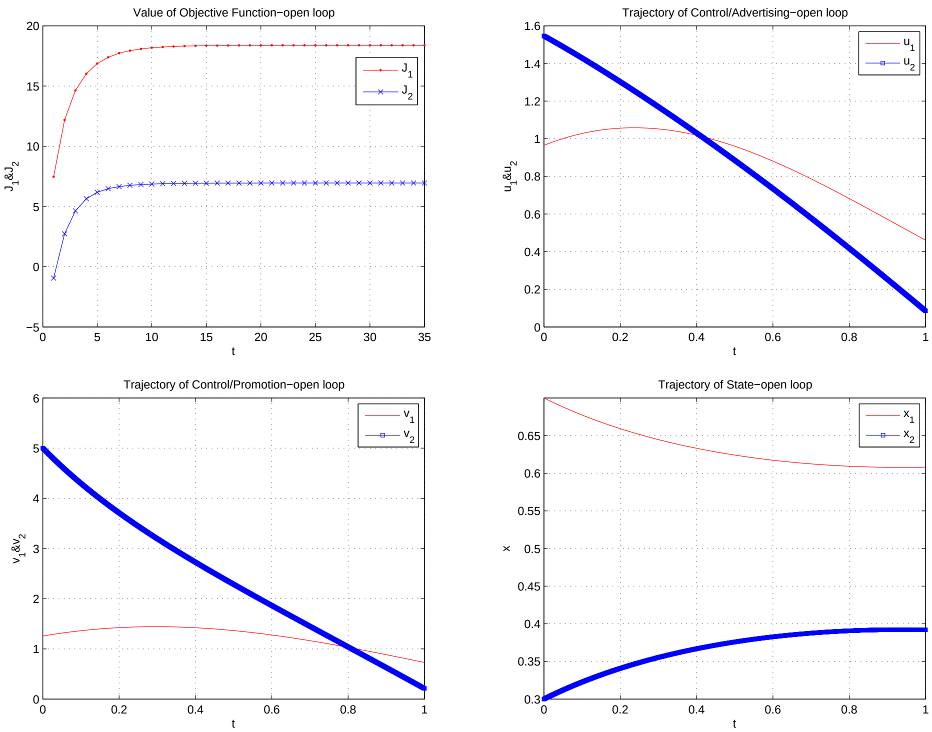

Figure 8: $c_1 > c_2$

Scenario 5. We choose the same coefficients as Scenario 3 except for:

$$c_1(t) = e^{-5-t},\ c_2(t) = e^{-5+t}.$$

Scenario 6. We choose the same coefficients as Scenario 3 except for:

$$c_1(t) = e^{-5+t},\ c_2(t) = e^{-5+t}.$$

In contrast to the coefficients of Scenario 3, where $c_1 = c_2 = e^{-5}$, interaction effectiveness in Scenario 5 will decrease as time goes on. In Scenario 6, the interaction effectiveness will increase as time goes on. From Figure 10, we can see that increase of interaction will lead to stronger competition between both companies, which means controls(advertising and promotion) of Scenario 6 are bigger than that of Scenario 3, and controls of Scenario 3 are bigger than that of Scenario 5. From Table 4, we can see that company 1's objective function value of Scenario 5 is largest and that of Scenario 6 is smallest, however company 2's objective function value of Scenario 5 is smallest and that of Scenario 6 is

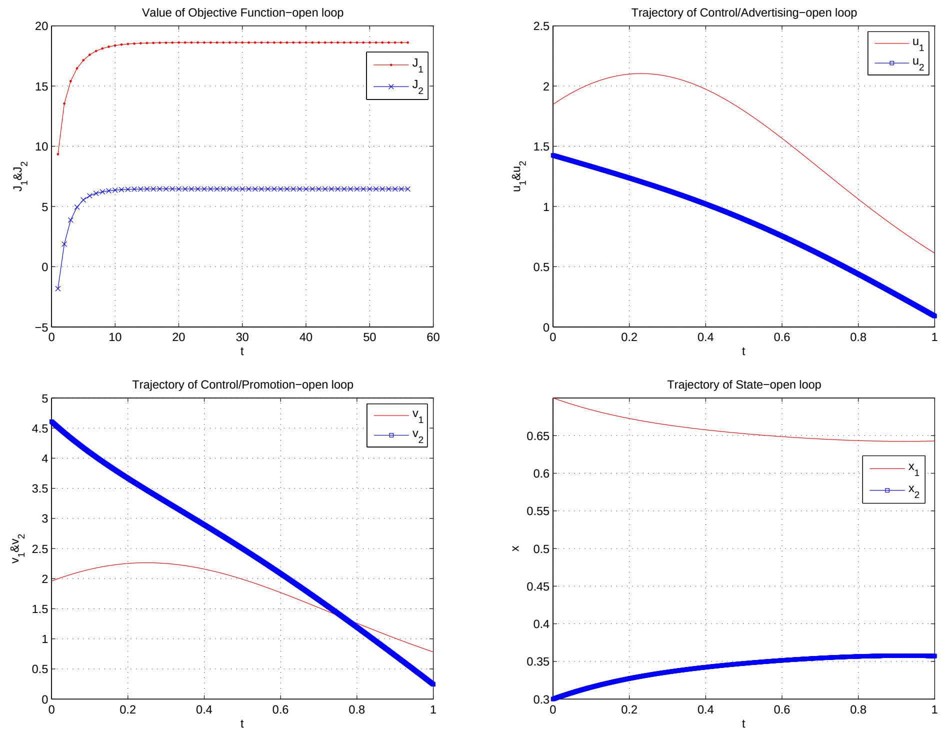

Figure 9: $c_1 \gg c_2$

	J_1	J_2
Scenario 6	18.32354	7.04788
Scenario 3	18.32375	7.04786
Scenario 5	18.32390	7.04784

Table 4: Comparison of objective value

largest. Thus, larger interaction of controls will lead to the stronger competition and the stronger competition will hurt the weaker company 1 and benefits the stronger company 2. Thus, in the competition for market, stronger company should consider strengthening the interaction of its controls.

Conclusion. In this section, we assumed control interaction coefficient $c_i(t)$ depends on time. In one scenario, $c_i(t)$ is increasing as time goes, in the other,

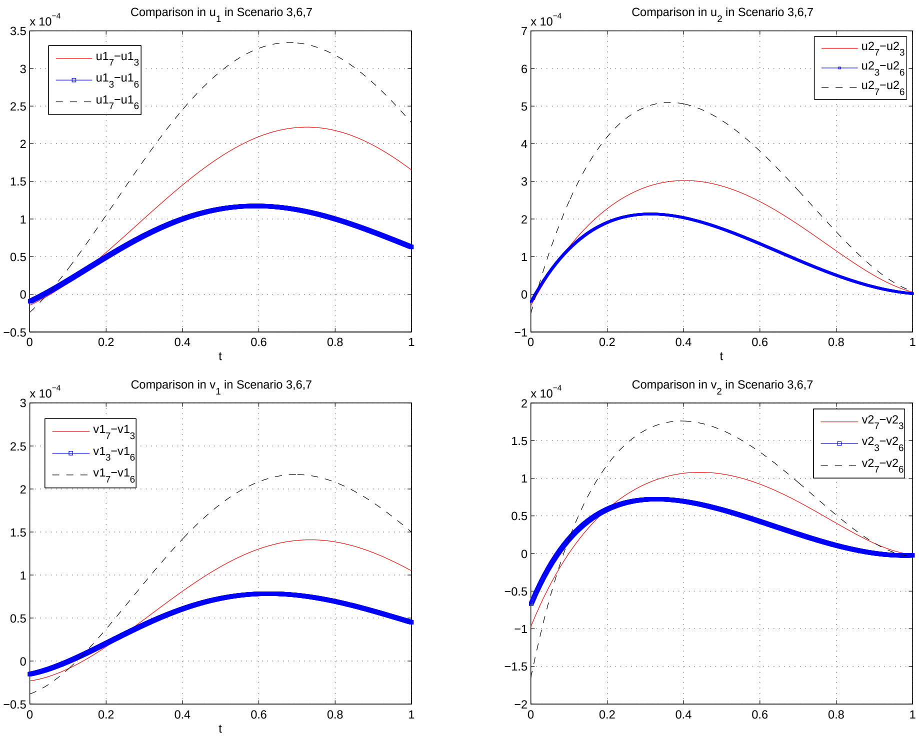

Figure 10: Comparison of controls

it is decreasing. From the numerical results, the increasing interaction leads to more furious competition. Furthermore, more furious competition benefits the stronger company. This is reasonable since the stronger company make the most of its conditions to compete.

6. Other Discussion

If we adopt another assumption that when company i does not employ any control, it still keeps a basic share of the market, then we adopt another form of $g_i(u_i, v_i)$:

$$g_i(u_i(t), v_i(t)) = e^{a_i u_i(t)+b_i v_i(t)} + c_i(t)u_i(t)v_i(t). \qquad (*)$$

Because of nonlinearity of $g_i(u_i(t), v_i(t))$, it is much harder to solve our model analytically, however, our algorithm is effective to solve it in both in open-loop and closed-loop cases. The following is a counterpart of Theorem 1 for the new g_i in $(*)$.

Theorem 2. *Suppose* $0 \le u_i \le L$ *and* $0 \le v_i \le L$, *and* $L \ge \max\{\frac{1}{a_i}, \frac{1}{b_i}\}, i = 1, 2, \cdots, n$ *then* $\max_t\{|c_i(t)|\} \le e^2 a_i b_i$ *guarantees that* $x_i(t) \ge 0$ *for* $0 \le t \le T, i = 1, 2, \cdots, n$.

For optimality conditions, we should solve the following system:

$$\begin{aligned}
\dot{x_1} =& (e^{a_1u_1+b_1v_1} + c_1u_1v_1) - x_1\sum_{k=1}^{n}(e^{a_ku_k+b_kv_k} + c_ku_kv_k),\\
&\vdots\\
\dot{x_i} =& (e^{a_iu_i+b_iv_i} + c_iu_iv_i) - x_i\sum_{k=1}^{n}(e^{a_ku_k+b_kv_k} + c_ku_kv_k),\\
&\vdots\\
\dot{x_n} =& (e^{a_nu_n+b_nv_n} + c_nu_nv_n) - x_n\sum_{k=1}^{n}(e^{a_ku_k+b_kv_k} + c_ku_kv_k),\\
-\dot{\lambda}_{i1} =& \frac{\partial H_i}{\partial x_1} + \sum_{j\neq i}\frac{\partial H_i}{\partial u_j}\frac{\partial u_j}{\partial x_1} + \sum_{j\neq i}\frac{\partial H_i}{\partial v_j}\frac{\partial v_j}{\partial x_1},\\
&\vdots\\
-\dot{\lambda}_{ii} =& \frac{\partial H_i}{\partial x_i} + \sum_{j\neq i}\frac{\partial H_i}{\partial u_j}\frac{\partial u_j}{\partial x_i} + \sum_{j\neq i}\frac{\partial H_i}{\partial v_j}\frac{\partial v_j}{\partial x_i},\\
&\vdots\\
-\dot{\lambda}_{in} =& \frac{\partial H_n}{\partial x_n} + \sum_{j\neq i}\frac{\partial H_i}{\partial u_j}\frac{\partial u_j}{\partial x_n} + \sum_{j\neq i}\frac{\partial H_i}{\partial v_j}\frac{\partial v_j}{\partial x_n},
\end{aligned}$$

where

$$\begin{aligned}
\frac{\partial H_i}{\partial x_i} =& -(p - \frac{\gamma_i}{2}v_i) - \lambda_{ii}G,\\
\frac{\partial H_i}{\partial x_j} =& -\lambda_{ij}G, \quad \text{for} \quad j \neq i,\\
\frac{\partial H_i}{\partial u_j} =& (a_je^{a_ju_j+b_jv_j} + c_jv_j)[\lambda_{ij}(1-x_j) - \sum_{k\neq j}\lambda_{ik}x_k],\\
\frac{\partial H_i}{\partial v_j} =& (b_je^{a_ju_j+b_jv_j} + c_ju_j)[\lambda_{ij}(1-x_j) - \sum_{k\neq j}\lambda_{ik}x_k],
\end{aligned}$$

$$
\begin{aligned}
&\{\delta_i+[\lambda_{ii}(1-x_i)-\sum_{k\neq i}\lambda_{ik}x_k]a_i^2e^{a_iu_i+b_iv_i}\}\frac{\partial u_i}{\partial x_i}\\
&\quad+[\lambda_{ii}(1-x_i)-\sum_{k\neq i}\lambda_{ik}x_k](a_ib_ie^{a_iu_i+b_iv_i}+c_i)\frac{\partial v_i}{\partial x_i}\\
&\quad=\lambda_{ii}(a_ie^{a_iu_i+b_iv_i}+c_iv_i),\\
&\lambda_{ii}(1-x_i)-\sum_{k\neq i}\lambda_{ik}x_k](a_ib_ie^{a_iu_i+b_iv_i}+c_i)\frac{\partial u_i}{\partial x_i}\\
&\quad+\{\gamma_ix_i+[\lambda_{ii}(1-x_i)-\sum_{k\neq i}\lambda_{ik}x_k]b_i^2e^{a_iu_i+b_iv_i}\}\frac{\partial v_i}{\partial x_i}\\
&\quad=\lambda_{ii}(b_ie^{a_iu_i+b_iv_i}+c_iu_i)-\gamma_iv_i,\\
&\{\delta_i+[\lambda_{ii}(1-x_i)-\sum_{k\neq i}\lambda_{ik}x_k]a_i^2e^{a_iu_i+b_iv_i}\}\frac{\partial u_i}{\partial x_j}\\
&\quad+[\lambda_{ii}(1-x_i)-\sum_{k\neq i}\lambda_{ik}x_k](a_ib_ie^{a_iu_i+b_iv_i}+c_i)\frac{\partial v_i}{\partial x_j}\\
&\quad=\lambda_{ij}(a_ie^{a_iu_i+b_iv_i}+c_iv_i)\\
&\lambda_{ii}(1-x_i)-\sum_{k\neq i}\lambda_{ik}x_k](a_ib_ie^{a_iu_i+b_iv_i}+c_i)\frac{\partial u_i}{\partial x_j}\\
&\quad+\{\gamma_ix_i+[\lambda_{ii}(1-x_i)-\sum_{k\neq i}\lambda_{ik}x_k]b_i^2e^{a_iu_i+b_iv_i}\}\frac{\partial v_i}{\partial x_j}\\
&\quad=\lambda_{ij}(b_ie^{a_iu_i+b_iv_i}+c_iu_i),\\
&x_i(0)=x_{i0},\\
&\lambda_{ij}(T)=\omega_i\frac{x_i(T)}{(\sum_{k=1}^n x_k(T))^2},\\
&\lambda_{ii}(T)=-\omega_i\frac{\sum_{k=1,k\neq i}^n x_k(T)}{(\sum_{k=1}^n x_k(T))^2},\\
&j\neq i,\quad i=1,2,\cdots,n.
\end{aligned}
$$

For the above differential algebraic equations, we cannot solve for the controls u_i, v_i explicitly, but we can solve for them by iterative Algorithm 2 efficiently. Numerical results are in Figure 11.

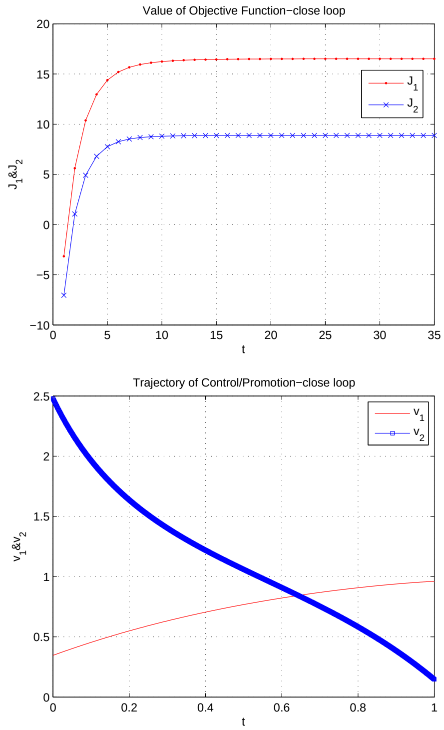

Figure 11: $\omega_1 > \omega_2$

7. Conclusion

This paper deals with **n** person noncooperative differential game model to analyze competition in the middle stage of a product life cycle. We adopted open-loop and closed-loop controls and solved our models from optimal conditions drawn from Pontryagin's type condition numerically. We analyzed the difference between open- and closed-loop controls, and the interaction of different controls from numerical results. As to the combination of controls, we can consider a more general $g_i(u_i(t), v_i(t)) = a_i(t)u_i(t) + b_i(t)u_i(t) + c_i(t)u_i(t)v_i(t)$, etc, and our iterative algorithm is still effective tool to solve this more general problem. The following are important observations:

1. The difference between open-and closed-loop controls: Erickson [13] and Chintagunta [10] have used statistical tools on the empirical data to conclude that closed-loop control trajectories fit the data better. In the literature, the

fact that closed-loop control is better than open-loop control comes in relation to *subgame perfect*. To be subgame perfect, an equilibrium must not depend on initial conditions [13]. To be specific, controls/strategies $u_i(x(t), t)$ that depend on current values of state variables as well as time and that do not depend on initial conditions are termed *feedback* strategies (Fershtman [15]). In this paper, we have strengthened the claim by numerical results that feedback controls are better than open-loop controls. In all situations of our model, the value of closed-loop controls is smaller than that of open-loop controls for all time, and the optimal objective values corresponding to closed-loop controls is always better than that corresponding to open-loop controls. We can define the difference of objective function value between open-loop and closed-loop control as *Current State Value*.

2. Here is a general observation about differential games. If Nash equilibrium exists for the differential game model, then the general competition strategy for each player is: Competing for resource with biggest efforts first, then decreasing efforts as time goes till equilibrium is reached. In all settings of our model, the whole "resource" is constant, i.e. $\sum_i^n x_i(t) = 1$. In the process of competing, players always try their best to compete for customers at first, and as time goes, when it appears that it is not possible to get more from the current situation, players compromise with each other, and reduce their effort to compete (i.e. reduce their controls), and reach equilibrium. One could anticipate this result even before the start of our numerical calculation. It supports a fact that a natural process is going from competition to coexistence (equilibrium). However, in practice since people do not know when equilibrium is reached, people will have to continue to compete hardest.

3. The optimality condition of differential game is BVP or DAE, so once the dimension is high, it is difficult to solve it analytically. In this paper, we have developed two algorithms, which can be used to solve general differential game models. The first algorithm is based on quasi-Newton method, has second order convergence; the second algorithm is based full-discritization, and is first order convergent algorithm.

References

[1] Leonard D. Berkovitz, A variational approach to differential games, In: *Advances in Game Theory* (AM-52 – Annals of Mathematics Studies, Eds: Melvin Dresher, Lloyd S. Shapley, Albert William Tucker), Princeton University Press (1964), 127-174.

[2] Leonard D. Berkovitz, A differential game without pure strategy solutions on an open set, In: *Advances in Game Theory* (AM-52 – Annals of Mathematics Studies, Ed-s: Melvin Dresher, Lloyd S. Shapley, Albert William Tucker), Princeton University Press (1964), 175-194.

[3] Leonard D. Berkovitz, The existence of value and saddle point in games of fixed duration, *SIAM Journal Control and Optimization*, **23**, No. 2 (1985).

[4] Leonard D. Berkovitz, Differential games of generalized pursuit and evasion, *SIAM Journal Control and Optimization*, **24**, No. 3 (1986).

[5] Leonard D. Berkovitz, Characterization of the values of differential games, *Applied Mathematics and Optimization*, **17** (1988), 177-183.

[6] Leonard D. Berkovitz, Differential games of survival, *Journal of Mathematical Analysis and Applications*, **129** (1988), 493-504.

[7] Leonard D. Berkovitz, A theroy of differential game, In: *Advances in Dynamic Games and Applications* (Ed-s: Tamer Başar, Alain Haurie), Birkhäuser Boston (1994).

[8] R. Cellini, L. Lambertini, Dynamic oligopoly with sticky prices: Closed-loop, feedback, and open-loop solutions, *Journal of Dynamical and Control Systems*, **10**, No. 3 (2004), 303-314, 341-369.

[9] P.K. Chintagunta, N.J. Vilcassim, An empirical investigation of advertising strategies in a dynamic duopoly, *Management Science*, **38**, No. 9 (1992), 1230-1244.

[10] P.K. Chintagunta, N.J. Vilcassim, Marketing investment decisions in a dynamic duopoly: A model and empirical analysis, *International Journal of Research in Marketing*, **11**, No. 3 (1994), 387-306.

[11] Englbert Dockner, Steffen Jorgensen, New-product advertising in dynamic oligopolies, *Methods and Models of Operations Research*, **36** (1992), 459-473.

[12] E.J. Dockner, G.E. Fruchter, Dynamic strategic pricing and speed of diffusion, *Journal of Optimization Theory and Applications*, **123**, No. 2 (2004), 331-348.

[13] Gary M. Erickson, Empirical analysis of closed-loop duopoly advertising strategies, *Management Science*, **38**, No. 12 (1992), 1732-1745.

[14] Gary M. Erickson, An oligopoly of dynamic advertising competition, *European Journal of Operational Research*, **197**, No. 1 (2009), 374-388.

[15] C. Fershtman, Identification of classes of differential games for which the open loop is a degenerate feedback Nash equilibrium, *Journal of Optimization Theory and Applications*, **55**, No. 2 (1987), 217-231.

[16] Lawrence Friedman, Game-theory models in the allocation of advertising expenditures, *Operations Research*, **6**, No. 5 (1958), 699-709.

[17] Avner Friedman, *Differential Games*, John Wiley and Son, New York (1971).

[18] Gila E. Fruchter, The many-player advertising game, *Management Science*, **45**, No. 11 (1999), 1609-1611.

[19] Gila E. Fruchter, Shlomo Kalish, Closed-loop advertising strategies in a duopoly, *Management Science*, **43**, No. 1 (1997), 54-63.

[20] R. Isaacs, Differential games I, II, III, IV, *RAND Corporation, Research Memoranda*, RM-1391, RM-1399, RM-1411, RM-1486 (1954-1955).

[21] Steffen Jorgensen, A survey of some differential games in advertising, *Journal of Economic Dynamic and Control*, **4** (1982), 341-369.

[22] N.G. Medhin, Wan Wei, Competition in the last stage of product life-cycle: A differential games approach, *Communications in Applied Analysis*, **12** (2008), 113-136.

[23] N.G. Medhin, Wan Wei, Multi-new product competition in duopoly: A differnetial game analysis, *Dynamic Systems and Applications*, **18**, No. 2 (2009).

[24] Alexander Mehlmann, *Applied Differential Game*, Plenum Press, New York (1988).

[25] Prasad A. Naik, Kalyan Raman, Russell S. Winer, Planning marketing-mix strategies in the presence of interaction effects, *Marketing Science*, **24**, No. 1 (2005), 25-34.

[26] Suresh P. Sethi, Dynamic optimal control models in advertising: A survey, *SIAM Review*, **19**, No. 4 (1977), 685-725.

[27] Gerhard Sorger, Competitive dynamic advertising: A modification of the case game, *Journal of Economic Dynamics and Control*, **13** (1989), 55-80.

[28] Jinn-Tsair Teng, Gerald L. Thompson, Oligopoly models for optimal advertising when production costs obey a learning curve, *Management Science*, **29**, No. 9 (1983), 1087-1101.