Neural, Parallel, and Scientific Computations 18 (2010) 451-460

NUMERICAL SOLUTION OF DIFFERENTIAL GAME MODELS

WEI WAN AND YUANYUAN PENG

Department of Mathematics, Claflin University, Orangeburg, SC 29115
Department of Mathematics, North Carolina State University, Raleigh, NC 27695

ABSTRACT. The applicability of differential game model is determined by the solvability of differential game. As to noncooperative differential game model, its optimality condition is Boundary Value Problems(BVPs) or Differential Algebraic Equations(DAEs). Thus, it is difficult to find analytical solution to this model except for some special case. In this paper, numerical method is developed to solve differential game models, and one iterative algorithm is given and discussed. And the algorithm is used to solve differential game model coming from competition in business competition. Numerical results are presented.

Keywords: differential games, numerical solution, optimality condition

MSc Subject classification: 91A23, 91A80, 91B50, 91B60

1. INTRODUCTION

Optimal control theory is a mathematical optimization method for deriving control policies. The method is largely due to the work of Lev Pontryagin and Richard Bellman. The abstract framework of optimal control goes as follows.

$$\min J(u(t)) = h(x(T), x(0)) + \int_0^T f(x(t), u(t), t)dt$$

subject to

$$(1) \qquad \begin{cases} \frac{dx(t)}{dt} = a(x(t), u(t), t) \\ u(t) \in \Omega \\ b(x(t), u(t), t) \leq 0 \ \text{ algebraic path constraints} \\ \phi(x(0), x(T)) \text{ boundary conditions} \end{cases}$$

In above optimal control model, there is one decision maker, who would like to optimize his objective function J by his control $u(t)$ under constraints of some ordinary differential system $\frac{dx(t)}{dt}$. Above optimal control can be solved analytically or numerically using Pontryagin's maximum principle, or by solving the Hamilton-Jacobi-Bellman equation. However, in some real dynamic system, there are more than one decision maker, and each decision maker uses his own control to optimize his own

Received April 15, 2010

objective function. In other words, in this system there exits competition. Game theory is a mathematical tool to investigate competition, which originates from Dr. John Nash. His game is called Matrix Game. In matrix game, the value of decision variable is the choice of row or column. As to Matrix Game, existence of Nash Equilibrium was solved, and Linear Programming was proved to be an efficient method to solve for its equilibrium. The Nash Equilibrium is define by the following inequalities under supposition that each decision maker wants to minimize his objective function:

$$\begin{cases} J_1(u_1, u_2^*, \cdots, u_n^*) \geq J_1(u_1^*, u_2^*, \cdots, u_n^*) \\ \vdots \\ J_n(u_1^*, u_2^*, \cdots, u_n) \geq J_n(u_1^*, u_2^*, \cdots, u_n^*) \end{cases} \tag{2}$$

where J_i is competitor i's objective value, u_i is competitor i's decision variable, and $(u_1^*, \cdots, u_n^*)$ is Nash Equilibrium. From (2), Nash Equilibrium means if any competitor deviates from equilibrium, then his objective value will be hurt.

Differential Game is a special Game model. Differential Game originated from R. Isaacs [9] in his research report from RAND Corporation around 1945. The Differential Game model is as follows:

$$\begin{cases} \min_{u_1} J_1 = h_1(x(T), x(0)) + \int_0^T f_1(x(t), u_1(t), \cdots, u_n(t), t)dt \\ \vdots \\ \min_{u_n} J_n = h_n(x(T), x(0)) + \int_0^T f_n(x(t), u_1(t), \cdots, u_n(t), t)dt \end{cases}$$

subject to

$$\begin{cases} \frac{dx(t)}{dt} = a(x(t), u_1(t), \cdots, u_n(t), t), \\ b(x(t), u_1(t), \cdots, u_n(t), t) \leq 0 \text{ algebraic path constraints} \\ \phi(x(0), x(T)) \text{ boundary conditions} \end{cases} \tag{3}$$

In the above model, competitor i has its objective functional J_i, and his decision variable u_i is a measurable function taking value from compact set Ω. All decision variables are imbedded in the same dynamics, but each competitor tries to optimize his own objective. The key assumption for above Differential Game is: 1) all competitors make decisions at the same time; 2) the status of competitor is symmetric/same, which means each competitor has the same full knowledge of the dynamics, and knows the other competitors' objective functions. The solution of above model is defined as Nash Equilibrium, which is described by the set of inequalities (2). The classical proofs of existence and optimality condition of Nash Equilibrium are from Dr. Friedman [1], Dr. Berkovtiz, [6]. After the existence and optimality condition is given, differential game is applied in many fields, such as economics, sociology, military, etc. The first application in marketing is due to Lawrence Friedman [5]. In recent investigation of the application of differential game model in marketing, Gila E. Fruchter, Shlomo Kalish [4] focus on closed-loop control in a duopoly. In Gary M. Ericson's paper [3], he investigates a special method to solve closed-loop control, but his method can be successful only if the form of objective function could be guessed. Other typical papers are as follows: Dockner and Jorgessen[2]. Teng, Jinn-Tsair, and Thompson[10]. Negash Medhin and Wan ([7], [8]).

Since the optimality condition for non-cooperative differential game is Differential Algebraic Equation (DAE) or Boundary Value Problem (BVP), it is changeling to solve it, especially when the dimension of problem is high. In this paper, we discuss the optimality condition for differential game models, develop numerical methods to solve optimality condition and models, and apply differential game model to competition in business.

2. OPTIMALITY CONDITION AND NUMERICAL ALGORITHM

The solution of non-cooperative differential game model is defined by Nash Equilibrium (1). Nash equilibrium is reached when there is no incentive for each competitor to change his control any more. The Nash Equilibrium is called 'optimal' in the sense that if one of the player deviates from the Nash equilibrium, his cost will increase.

In the application of differential games, there are two types of controls that are commonly used: *Open-loop* and *Closed-loop*. Control $\mathbf{u}$ is called *open-loop* control if $\mathbf{u} = \mathbf{u}(\mathbf{t})$. Control $\mathbf{u}$ is called *closed-loop* if $\mathbf{u} = \mathbf{u}(\mathbf{t}, \mathbf{x}(\mathbf{t}))$. Open-loop controls are much easier to compute than closed loop controls, and the disadvantage in reality is that the competitors choose their controls at the beginning of the game and comply with their strategies in the game. The advantage of a closed-loop control is that the players adjust their controls according to the state. The disadvantage of closed-loop controls is that they are much more difficult to compute.

The necessary optimality condition for non-cooperative differential game model comes from Pontryagin's Minimum Principle. For open-loop control, the optimality condition is given by following theorem([1], [6]).

Theorem 1. *For an N-person differential game in fixed duration $[0, T]$, if $\{\mathbf{u}_\mathbf{i}^*(\mathbf{t}), \mathbf{i} \in \mathbf{N}\}$ provides an open-loop Nash equilibrium solution, and $\{\mathbf{x}^*(\mathbf{t}), \mathbf{0} \le \mathbf{t} \le \mathbf{T}\}$ is the corresponding state trajectory, there exist N costate functions $\mathbf{p}_\mathbf{i}(\cdot) : [\mathbf{0}, \mathbf{T}] \to \mathbf{R}^\mathbf{n}$, $\mathbf{i} \in \mathbf{N}$, such that the following relations are satisfied:*

$$\frac{dx^*}{dt} = a(t, x^*, u_1^*, \cdots, u_i^*, \cdots, , u_N^*), \quad x^*(0) = x_0$$

$$u_i^* = \underset{u_i \in \Omega_i}{\operatorname{argmin}}\, H_i(t, p_i, x^*, u_1^*, \cdots, u_{i-1}^*, \mathbf{u_i}, u_{i+1}^* \cdots, u_N^*)$$

$$\frac{dp_i}{dt} = -\frac{\partial H_i(t, p_i, x^*, u_1^*, \cdots, u_i^*, \cdots, u_N^*)}{\partial x}$$

$$p_i(T) = \frac{\partial h_i(x^*(T))}{\partial x}, \quad i \in N$$

where

$$H_i(t, p_i, x, u_1, \cdots, u_N) \triangleq f_i(t, x, u_1, \cdots, u_N) + p_i \cdot a(t, x, u_1, \cdots, u_N)$$

For closed-loop control, the optimality condition is given by following theorem ([1]).

Theorem 2. *For an N-person differential game in fixed duration $[0, T]$, if $\{\mathbf{u_i^*(t, x(t))}, \mathbf{i \in N}\}$ provides an closed-loop Nash equilibrium solution, and $\{\mathbf{x^*(t), 0 \le t \le T}\}$ is the corresponding state trajectory, there exist N costate functions $\mathbf{p_i}(\cdot) : [\mathbf{0}, \mathbf{T}] \to \mathbf{R^n}$, $\mathbf{i \in N}$, such that the following relations are satisfied:]*

$$\frac{dx^*}{dt} = a(t, x^*, u_1^*, \cdots, u_i^*, \cdots, u_N^*), \quad x^*(0) = x_0$$

$$u_i^* = argmin_{u_i \in \Omega_i} H_i(t, p_i, x^*, u_1^*, \cdots, u_{i-1}^*, \mathbf{u_i}, u_{i+1}^* \cdots, u_N^*)$$

$$\frac{dp_i}{dt} = -\frac{\partial H_i(t, p_i, x^*, u_1^*(x(t), t), \cdots, u_i^*(x(t), t), \cdots, u_N^*(x(t), t)}{\partial x}$$

$$p_i(T) = \frac{\partial h_i(x^*(T))}{\partial x}, \quad i \in N$$

where

$$H_i(t, p_i, x, u_1, \cdots, u_N) \triangleq f_i(t, x, u_1, \cdots, u_N) + p_i \cdot a(t, x, u_1, \cdots, u_N)$$

In above necessary optimality conditions for both open- and closed-loop, the second equation is optimal control for each competitor. For example, competitor i's control u_i minimizes his own hamiltonian H_i conditioning on all other competitors' controls are optimal controls. The difference between open- and closed-loop control is costate equations in closed-loop case are partial differential equation with the form of $\frac{\partial u_i}{\partial x}$.

The algorithm based on solving BVP or DAE needs the controls u_i to be solvable explicitly from optimality condition([7]). The design of the following algorithm does not need this condition. It is an iterative algorithms. The idea of iterative algorithm is based on Medhin and Wan ([8]). The process is as follows: first generating $\mathbf{n}$ controls for $\mathbf{n}$ players separately; second, solve 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 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:

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:

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

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.

3. MODELING

Detail modeling background can be found at [7], [8]. In a general life cycle of a product, the market sale of some product will keep approximately constant for some time. A non-cooperative model differential game will be set up for this time interval. Thus, we assume that the total market sale is fixed. Furthermore, we suppose this is an 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.

The dynamics is as followings:

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

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. In our modeling, one assumption is when company i does not employ any control, it still keeps a basic share of the market. Thus, we take the form of

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

In the numerical experiment, there are two companies. And supposing that 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, $a_1 < a_2, b_1 < b_2$.

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

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

Based on above dynamics, objective functions, and optimality condition (Theorem 2), the explicit optimality condition is as follows:

$$\dot{x_1 7} = (e^{a_1 u_1 + b_1 v_1} + c_1 u_1 v_1) - x_1 \sum_{k=1}^{n} (e^{a_k u_k + b_k v_k} + c_k u_k v_k)$$

$$\cdots\cdots\cdots\cdots$$

$$\dot{x_i} = (e^{a_i u_i + b_i v_i} + c_i u_i v_i) - x_i \sum_{k=1}^{n} (e^{a_k u_k + b_k v_k} + c_k u_k v_k)$$

$$\cdots\cdots\cdots\cdots$$

$$\dot{x_n} = (e^{a_n u_n + b_n v_n} + c_n u_n v_n) - x_n \sum_{k=1}^{n} (e^{a_k u_k + b_k v_k} + c_k u_k v_k)$$

$$-\dot{\lambda_{i1}} = \frac{\partial H_i}{\partial x_1} + \sum_{j \neq i} \frac{\partial H_i}{\partial u_j} \frac{\partial u_j}{\partial x_1} + \sum_{j \neq i} \frac{\partial H_i}{\partial v_j} \frac{\partial v_j}{\partial x_1}$$

$$\cdots\cdots\cdots\cdots$$

$$-\dot{\lambda_{ii}} = \frac{\partial H_i}{\partial x_i} + \sum_{j \neq i} \frac{\partial H_i}{\partial u_j} \frac{\partial u_j}{\partial x_i} + \sum_{j \neq i} \frac{\partial H_i}{\partial v_j} \frac{\partial v_j}{\partial x_i}$$

$$\cdots\cdots\cdots\cdots$$

$$-\dot{\lambda_{in}} = \frac{\partial H_n}{\partial x_n} + \sum_{j \neq i} \frac{\partial H_i}{\partial u_j} \frac{\partial u_j}{\partial x_n} + \sum_{j \neq i} \frac{\partial H_i}{\partial v_j} \frac{\partial v_j}{\partial x_n}$$

where

$$\frac{\partial H_i}{\partial x_i} = -(p - \frac{\gamma_i}{2} v_i) - \lambda_{ii} G$$

$$\frac{\partial H_i}{\partial x_j} = -\lambda_{ij} G, \quad \text{for } j \neq i$$

$$\frac{\partial H_i}{\partial u_j} = (a_j e^{a_j u_j + b_j v_j} + c_j v_j) \left[\lambda_{ij}(1 - x_j) - \sum_{k \neq j} \lambda_{ik} x_k \right]$$

$$\frac{\partial H_i}{\partial v_j} = (b_j e^{a_j u_j + b_j v_j} + c_j u_j) \left[\lambda_{ij}(1 - x_j) - \sum_{k \neq j} \lambda_{ik} x_k \right]$$

$$\left\{ \delta_i + \left[\lambda_{ii}(1 - x_i) - \sum_{k \neq i} \lambda_{ik} x_k \right] a_i^2 e^{a_i u_i + b_i v_i} \right\} \frac{\partial u_i}{\partial x_i}$$

$$+ \left[\lambda_{ii}(1 - x_i) - \sum_{k \neq i} \lambda_{ik} x_k \right] (a_i b_i e^{a_i u_i + b_i v_i} + c_i) \frac{\partial v_i}{\partial x_i}$$

$$= \lambda_{ii}(a_i e^{a_i u_i + b_i v_i} + c_i v_i)$$

$$\left[\lambda_{ii}(1 - x_i) - \sum_{k \neq i} \lambda_{ik} x_k \right] (a_i b_i e^{a_i u_i + b_i v_i} + c_i) \frac{\partial u_i}{\partial x_i}$$

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

In the above Differential Algebraic Equations, we can not solve controls u_i, v_i explicitly, but it can be solved by our iterative algorithm efficiently. Numerical results are in *Figure 1*. The upper left graph is for objective function values. We can see that the algorithm was searching for the equilibrium for objective values, and converging. The upper right graph are optimal controls-advertising for these two players. The lower left graph are optimal controls-promotion for these two players. The lower right graph are sale trajectories.

4. **CONCLUSION**

Many real competition can be described and modeled as non-cooperative differential game. But the applicability of this model depends on its solvability. Based on the properties of optimality for differential game model, which is BVP or DAE, it is difficult to solve it analytically. In this paper, an iterative algorithm is developed to solve this model. Compared with other algorithms, the advantage of this algorithm is that it can be used for general differential game model. It does not need controls to

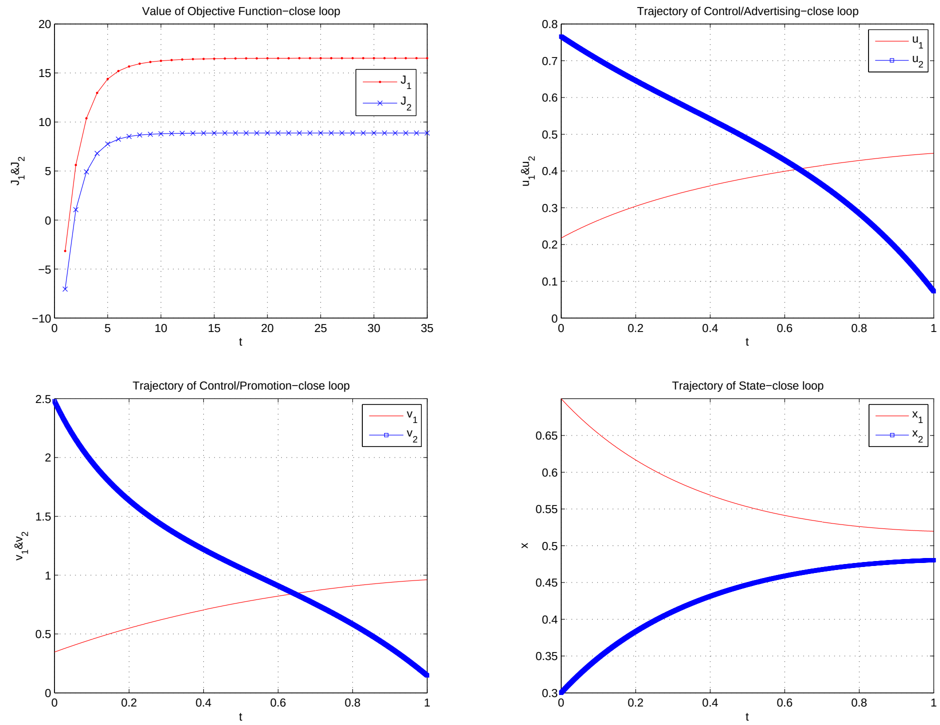

FIGURE 1. $\omega_1 > \omega_2$

be solved explicitly from optimality condition. But the disadvantage of this algorithm is that it just has first-order convergence.

REFERENCES

[1] Avner Friedman, *Differential Games*, Dover Publications, May 12, 2006.

[2] Dockner, Englbert and Steffen Jorgensen, *New-Product Advertising in Dynamic Oligopolies*, Methods and Models of Operations Research, Vol 36, 1992, pp. 459–473

[3] Gary M. Erickson, *An Oligopoly of Dynamic Advertising Competition*, 2007, To be published.

[4] Gila E. Fruchter, Shlomo Kalish, *Closed-loop Advertising Strategies in a Duopoly*, Management Science, Vol. 43, No. 1, pp. 54–63, Jan. 1997.

[5] Lawrence Friedman *Game-Theory Models in the Allocation of Advertising Expenditures*, Operations Research, Vol. 6, No.5, Sep.-Oct.,1958, pp. 699–709.

[6] Leonard D. Berkovitz, *The Existence of Value and Saddle Point in Games of Fixed Duration*, SIAM Journal on Control and Optimization. Vol. 23, No. 2, March 1985.

[7] Negash Medhin, Wei Wan, *Competition in the Last Stage of Product Life-cycle: A Differential Games Approach*, Communications in Applied Analysis, 12 (2008), pp. 113–136.

[8] Negash Medhin, Wei Wan, *Multi-New Product Competition in Duopoly: A Differential Game Analysis*, Dynamic Systems and Applications, Vol 18, No.2, June 2009.

[9] Rufus Isaacs, *Differential Games*, John Wiley and Sons, 1967.

[10] Teng, Jinn-Tsair and Gerald L. Thompson, *Oligopoly Models for Optimal Advertising When Production Costs Obey a Learning Curve*, Management Science, Vol. 29, No. 9, Sep. 1983, pp. 1087–1101