

RaBVIt-SG, an algorithm for solving Feedback Nash equilibria in Multiplayers Stochastic Differential Games

Jorge Herrera de la Cruz^a, Benjamin Ivorra^b, Ángel M. Ramos^b

^a Department of Economic Analysis and Quantitative Economics
Complutense University of Madrid
Campus de Somosaguas, s/n, 28223 Pozuelo de Alarcón, Spain
email: jorherre@ucm.es

^bInstituto de Matemática Interdisciplinar (IMI),
Department of Mathematical Analysis and Applied Mathematics,
Complutense University of Madrid,
Plaza de las Ciencias, 3, Madrid 28040, Spain
email: ivorra@ucm.es;angel@ucm.es

July 18, 2019

Abstract

In a previous work, we have introduced an algorithm, called RaBVItG, used for computing Feedback Nash equilibria of deterministic multiplayer Differential Games. This algorithm is based on a sequence of Game Iterations (i.e., a numerical method to simulate an equilibrium of a Differential Game), combined with Value Iterations (i.e., a numerical method to solve a Dynamic Programming equation associated to a Differential Game). Here, our main objective is to present an extension of this algorithm, called RaBVIt-SG, considering stochastic multiplayer Differential Games. More precisely, we consider Differential Games including a multivariate Brownian term in the diffusive part of the Stochastic Differential Equations system. Hence, with respect to the previous deterministic version of the algorithm, we tackle the stochastic scheme approximating the diffusion by proper Markov chains in a finite dimensional space. To illustrate our approach, we consider a marketing problem based on a real data set. In particular, we study the relevance of the stochasticity in the optimal policies of the considered players and the impact of the number of players on the obtained solutions. Finally, we compare the solutions returned by RaBVIt-SG with the real observations and with the solution returned by RaBVItG considering a deterministic version of the game.

keywords: Stochastic Differential Games, Feedback-Nash equilibrium, Semi-Lagrangian discretization, Meshfree methods.

1 Introduction

In this paper, we aim to analyze the effect of uncertainties on multiplayer Feedback Nash Stochastic Differential Games (FNSDG). To do so, we need to solve the corresponding stochastic Hamilton-Jacobi-Bellman (HJB) system of equations (see, for instance, [3, 2]).

Currently, the development of algorithms to tackle Stochastic Differential Games (SDG) is a topic in expansion (see, e.g., early publications [14, 3] and recent contributions [29, 35, 7, 8]). Most of the works proposed in the literature focus on studying theoretical properties of some classes of SDGs. Additionally, some works apply SDG to marketing models and their solutions are obtained by using some heuristic approximations, avoiding to solve the stochastic HJB equation related to the problem (see, e.g., [26, 27, 42, 45]).

In a previous work, we have designed an algorithm, called RaBVItG (Radial Basis Value Iteration Game), in order to solve the HJB system arising when considering Feedback Nash Deterministic Differential Games (FNDDG). This algorithm is based on the combination of two main steps: Value Iteration (see [4, 41, 46]) and Game Iteration (see [23]). More precisely, RaBVItG includes a Value Iteration (VI) which aims to find a fixed point solution for the coupled system of value functions (one per player) involved in the considered FNDDG. Additionally, after each VI step, the algorithm performs a Game Iteration (GI) to find the corresponding game equilibrium (Nash, in our case) associated to a set of Value Functions. Furthermore, RaBVItG incorporates Function Approximation methods to simplify the model in order to approximate, using meshfree techniques, the Value Function for each player (see for instance, [34, 47]). Finally, focusing on the numerical implementation, RaBVItG is based on a Semi-Lagrangian discretization of time and space (see [12]) and uses a discrete version of the HJB equations that arises in our problem, using dynamic programming principle. A complete description of this algorithm and validation considering deterministic FNDG can be found in [23].

Here, to discuss the effect of uncertainties on Player's decisions, we model the dynamics of the state variables of the studied FNSDG by considering a stochastic System of Differential Equations (SDE), which incorporates a Brownian noise term (see [38]). To solve numerically this kind of FNSDG, we need to introduce a stochastic version of RaBVItG, called RaBVIt-SG. More precisely, we consider RaBVItG as a particular case of a broader general method which is able to work with uncertainties in the *diffusive* term of the SDE (i.e., the Brownian term introduced previously). In this case, we introduce a mesh-free scheme, based on Markov-Chains methods from [11] and [32], to deal with this diffusive term. Next, in order to illustrate our approach, we consider a particular FNSDG based on a real data base from a Marketing-Mix problem. The objective is to compute optimal advertising and pricing strategies, in order to maximize an expected profit function per player, with four retailers in competition. We analyze the effect of the different sources of uncertainties on the returned solutions, taking into account all the players in the game (and avoiding typical reductions to 2 players, see, e.g., [42]). Then, we compare the solutions returned by RaBVIt-SG with real data, with the ones obtained with RaBVItG by considering a deterministic version of the SDG, and with some results found in the literature.

This paper is organized as follows. In Section 2, we present the theoretical model, describe the game variables and the coupled optimization problems and define the considered equilibrium concepts. In Section 3, we describe the algorithm RaBVIt-SG used to solve the FNSDG introduced previously. In Section 4, we show the performance of this algorithm by considering a Marketing problem based on real data with 2 and 4 players. In particular, we discuss the impact of uncertainties on the solutions by comparing them with deterministic solutions and by using different stochastic sources. Finally, in Section 5, we summarize our main results and conclusions.

2 Mathematical Model

In this section, we describe the considered FNSDG. To do so, we first define the main functions and variables of a general Stochastic Differential Game. Secondly, we introduce the HJB equations associated to the SDG and the Feedback-Nash equilibria we aim to find.

2.1 Stochastic Differential Game

Here, we present some concepts related to the class of SDG tackled during this work. Considering the models proposed in [2] and [3], we assume some uncertainties related to the player's state variables and, hence, those players have to compute an expected value of their cost/benefit functions once they have scheduled their policies. For instance, in the particular cases

detailed in Section 4, the players exhibit uncertainties on the value of their future sales related to their prices and advertising policies. Thus, for each player, we compute the expected benefit related to these variables.

We consider a SDG with N players and a $p \in \mathbb{N}$ dimensional (with $p \neq N$) stochastic process $\{x(t)\}_{t \geq 0}$ (see, for instance, [38]), corresponding to the state of the system, governed by

$$dx(t) = g(x(t), u(t); \theta) dt + \sigma(x(t)) dW(t), \quad t \in [0, \infty), \quad (1)$$

and

$$x(0) = y \in \mathbb{R}^p, \quad (2)$$

where

- $x : [0, \infty) \rightarrow X$, with $X \subset \mathbb{R}^p$ is the set of admissible states;
- $g : X \times U \rightarrow \mathbb{R}^p$ denotes the *drift* of the Stochastic Differential Equation, satisfying
 - $|g(x_1, u) - g(x_2, u)| \leq L_g |x_1 - x_2|$, with $L_g > 0$,
 - $|g(x, u)| \leq M_g$, with $M_g > 0$;
- $\theta \in \Theta \subset \mathbb{R}^g$ is the multidimensional parameter vector, where Θ is a parameter space;
- $\sigma(\cdot) : X \rightarrow \mathbb{R}^{p \times p}$ is the covariance matrix green of the considered "noise" term in the equation.
- $\{W(t)\}_{t \geq 0}$ denotes a p -dimensional standard Wiener process defined on a complete filtered probability space $(\Omega, \mathcal{F}, \mathbb{P}, \{\mathcal{F}_t\}_{t \geq 0})$. Here, Ω is a set, $\{\mathcal{F}_t\}_{t \geq 0}$ is the natural filtration of $\{W(t)\}_{t \geq 0}$, \mathcal{F} is a sigma-algebra of subsets of Ω and \mathbb{P} is a mapping from \mathcal{F} into $[0, 1]$ satisfying that $\mathbb{P}(\emptyset) = 0$, $\mathbb{P}(\Omega) = 1$ and $\mathbb{P}(\cup_{m=1}^{\infty} A_m) = \sum_{m=1}^{\infty} \mathbb{P}(A_m)$, with $A_m \cap A_n = \emptyset$ for all $m \neq n$ (see [38] for more details).

We recall that, considering previous notations, $\omega \in \Omega$ is called *sample points*, $A \subset \Omega$ is called *event*, \mathbb{P} is called *probability measure* on Ω and $\mathbb{P}(A) \in [0, 1]$ is the *probability* of the event A . Furthermore, a p -dimensional Wiener process is a random variable $W : T \times \Omega \rightarrow \mathbb{R}^p$, with $T = \mathbb{R}^+$, that satisfies the following conditions:

- for any time $s \leq t$, $\mathbb{E}\{W(t, \cdot) - W(s, \cdot)\} = 0$ (independent increments),
- for any $s < t$, $W(t, \cdot) - W(s, \cdot)$ is multivariate normally distributed and has a similar distribution than $W(t-s, \cdot) - W(0, \cdot)$ (stationary increments),
- for almost every $\omega \in \Omega$, the functions $t \mapsto W(t, \omega)$ are continuous.

We assume that, given a progressively measurable process (see [32], Ch.1) $\{u(t)\}_{t \geq 0}$, there exists a unique progressively measurable process $\{x(t)\}_{t \geq 0}$ satisfying the controlled Stochastic Differential Equation (1). This process $\{u(t)\}_{t \geq 0}$ corresponds to the set of controls applied to the System (1)-(2) at time t with values in a compact metric space $U \subset \mathbb{R}^m$. In our case, we denote by \mathcal{U}_i the set of (progressively measurable) admissible control laws for player i that takes values in U_i . Thus, $U = U_1 \times U_2 \times \dots \times U_N$, with $U_i \subset \mathbb{R}^{m_i}$.

A solution of System (1)-(2) consists in a collection of sample paths of a stochastic process, and the associated probability of those paths.

Additionally, each player $i \in \{1, 2, \dots, N\}$ has an expected payoff functional conditioned to $x(0) = y$ over the time interval $[0, \infty)$, denoted by $J_i : X \times \mathcal{U} \rightarrow \mathbb{R}$, and given by:

$$J_i(y, u) = \mathbb{E} \left\{ \int_0^\infty e^{-\rho_i t} (f_i(x(t), u(t))) dt \mid x(0) = y \right\}. \quad (3)$$

where f_i and $\rho_i > 0$ are the benefit function and the discounting parameter of player i , respectively. Furthermore, f_i satisfies

- $f_i : X \times U_i \rightarrow \mathbb{R}$, with $r_i > 0$,
- $|f(x_1, u) - f(x_2, u)| \leq L_f |x_1 - x_2|$, being $L_f > 0$,
- $|f(x, u)| \leq M_f$, being $M_f > 0$.

Considering all the previous assumptions and definitions, the class of SDG proposed here is of the form:

Given $y \in \mathbb{R}^p$, each player $i = 1, \dots, N$ intends to maximize $J_i(y, u)$, by controlling the component $u_i \in \mathcal{U}_i$ of $u \in \mathcal{U}$, governed by

$$\begin{cases} J_i(y, u) = \mathbb{E} \left\{ \int_0^\infty e^{-\rho_i t} (f_i(x(t), u(t))) dt | x(0) = y \right\}, \\ dx(t) = g(x(t), u(t); \theta) dt + \sigma(x(t)) dW(t), \\ x(0) = y. \end{cases} \quad (4)$$

2.2 HJB equations of the considered SDG

Let $v_i : X \rightarrow \mathbb{R}$, be a continuously differentiable mapping called *value function* of our SDG, and defined by

$$v_i(y) = \sup_{u \in \mathcal{U}} J_i(y, u).$$

For each $y \in X$, we assume the existence of, at least, one optimal control $u_y^* \in \mathcal{U}$ such that

$$v_i(y) = J_i(y, u_y^*) = \mathbb{E} \left\{ \int_0^\infty e^{-\rho_i t} f_i(x_y(t, u_y^*), u_y^*) dt | x(0) = y \right\}, \quad (5)$$

where $x_y(t, u_y)$ is an admissible trajectory satisfying (1)-(2).

We denote by $x_y^*(t) = x_y(t, u_y^*)$ the optimal trajectory associated to u_y^* .

It can be proven (see, for instance [32] and [10]) that the HJB equations associated to this stochastic optimal control problem is

$$\rho_i v_i(y) = \max_{u \in U} \left\{ f_i(y, u) + \nabla_x v(y)^t g(y, u) + \frac{1}{2} \text{tr} [\sigma(y) \sigma(y)^t \nabla_{xx} v(y)] \right\}.$$

where $\text{tr}(\cdot)$ is the trace operator

Now, let $S_i : X \rightarrow U_i$, called the *feedback-map* per player $i = 1, \dots, N$ of the HJB equations, defined by

$$S_i(y) \in \arg \max_{u \in U} \left\{ f_i(y, u) + \nabla_x v(y)^t g(y, u) + \frac{1}{2} \text{tr} [\sigma(y) \sigma(y)^t \nabla_{xx} v(y)] \right\},$$

where $y \in \mathbb{R}^p$.

We consider an optimal control defined by

$$u_y^*(t) = S_i(x_y^*(t)), \quad (6)$$

for almost every $t > 0$.

Considering the *feedback-map*, we build an optimal decision $u_y^*(t) = S_i(x_y^*(t))$ related to the corresponding optimal trajectory, by solving

$$dx(t) = g(x(t), [S_1(x(t)), \dots, S_N(x(t))]; \theta) dt + \sigma(x(t)) dW(t), \quad t \in [0, \infty), \quad (7)$$

and

$$x(0) = y. \quad (8)$$

2.3 Stochastic Feedback-Nash Equilibrium of the considered SDG

Once we have defined the stochastic HJB equation, and its associated feedback-map, we build, step by step, the Feedback Nash Equilibrium associated to the considered problem.

We define a *Stochastic Feedback N-tuple* as : $\mathcal{S} \equiv (\mathcal{S}_1, \dots, \mathcal{S}_{i-1}, \mathcal{S}_i, \mathcal{S}_{i+1}, \dots, \mathcal{S}_N)$ and a *Stochastic Feedback (N-1)-tuple* as : $\mathcal{S}_{-i} \equiv (\mathcal{S}_1, \dots, \mathcal{S}_{i-1}, \mathcal{S}_{i+1}, \dots, \mathcal{S}_N)$. Then, $\mathcal{S}_* \equiv (\mathcal{S}_{*,1}, \dots, \mathcal{S}_{*,i-1}, \mathcal{S}_{*,i}, \mathcal{S}_{*,i+1}, \dots, \mathcal{S}_{*,N})$ is a *Stochastic Feedback- Nash Equilibrium (SFNE)* if the following relationship holds (see [3]):

$$J_i(y; \mathcal{S}_*) \geq J_i(y; [u_i, \mathcal{S}_{*,-i}]) \text{ for every } y \in \mathbb{R}^p, \mathcal{S}_i : X \rightarrow U_i \text{ and } i \in \{1, \dots, N\}, \quad (9)$$

where, for the sake of simplicity, given a *Stochastic Feedback N-tuple* \mathcal{S} , $J_i(y, \mathcal{S}_i)$ denotes $J_i(y, \mathcal{S}_i(x(t)))$, and $[u_i, \mathcal{S}_{*,-i}]$ is the m -vector of controls obtained by replacing the i -th component of $\mathcal{S}_{*,i}$ by u_i .

Assuming that y and $\mathcal{S}_{*,-i}$ are fixed, as done in the deterministic case (see [23]), we compute u_i that maximizes $J_i(y; [u_i, \mathcal{S}_{*,-i}])$ by considering the following value function $v_i : X \rightarrow \mathbb{R}$

$$v_i(y; \mathcal{S}_{*,-i}) = \max_{u_i \in \mathcal{U}_i} J_i(y; [u_i, \mathcal{S}_{*,-i}]), \quad (10)$$

which is also the solution of the following HJB equation

$$\rho_i v_i(y; \mathcal{S}_{*,-i}) = \max_{u_i \in \mathcal{U}_i} \left\{ f_i(y, [u_i, \mathcal{S}_{*,-i}]) + \nabla_x v(y)^t g(y, [u_i, \mathcal{S}_{*,-i}]) + \frac{1}{2} \text{tr} [\sigma(y) \sigma(y)^t \nabla_{xx} v(y, [u_i, \mathcal{S}_{*,-i}])] \right\}. \quad (11)$$

3 Numerical implementation of RaBVit-SG

Now, we present the numerical methods involved to solve the FNSDG presented previously. More precisely, in Section 3.1, we introduce the Semi-Lagrangian discretization of the SDG problem. Next, in Section 3.2, we develop the pseudocode of the algorithm, focusing on the Game and Value iterations, used to approximate the solution to the considered problem.

3.1 Time and space discretization

We first introduce the scheme used to discretize Equations of system (4). To do so, we consider the following discrete-time version of the cost function, in (4)

$$J_i^h(y, u^h) = \mathbb{E} \left\{ h \sum_{k=0}^{\infty} e^{-\rho_i k} f_i(x_k, u_k) \mid x_0 = y \right\}, \quad (12)$$

where

- h is the time step,
- x_k is an approximated value of $x(t_k)$, with $t_k = kh$, which is given by the following discrete-time stochastic dynamic system, known as the Euler-Maruyama scheme (see [24]):

$$x_{k+1} = x_k + hg(x_k, u_k; \theta) + \sigma(x_k) \xi_k \quad \text{and,} \quad x_0 = y, \quad (13)$$

with ξ_k , the increment of the standard Brownian motion $\{W(t)\}_{t \geq 0}$ on the interval $[t_k, t_{k+1}]$

- $u_k = [u_{1,k}, \dots, u_{N,k}] \in U^h$, with $u_{i,k} \in U_i^h$ and considering the assumption that the controls are constant on $[t_k, t_{k+1})$,
- $u^h = [u_1^h, \dots, u_N^h] \in \mathcal{U}^h$, with $u_i^h = \{u_{i,0}, u_{i,1}, \dots\}, i = 1, \dots, N$.

For the sake of simplicity, we rewrite Equation (13) as $x_{k+1} = x_k + \delta_d$ with δ_d , for $d = 1, \dots, 2^p$, given by

$$\delta_d(x_k, u) = hg(x_k, u; \theta) + \sigma(x_k) \begin{pmatrix} \pm\sqrt{h} \\ \dots \\ \pm\sqrt{h} \end{pmatrix}.$$

Indeed, according to [5] and [11], Equation (13) can be interpreted as a Markov chain, where the Gaussian random variable ς^k in (13) can be replaced by a two-point distributed variable with probability distribution $\mathbb{P}(\xi_k = \sqrt{h}) = \mathbb{P}(\xi_k = -\sqrt{h}) = \frac{1}{2}$, whenever the focus of (12) is on expected values (see, for instance, [32]).

Next, we define discrete-time versions of the stochastic feedback $\mathcal{S}^h \equiv (\mathcal{S}_1^h, \dots, \mathcal{S}_N^h)$ and $\mathcal{S}_{-i}^h \equiv (\mathcal{S}_1^h, \dots, \mathcal{S}_{i-1}^h, \mathcal{S}_{i+1}^h, \dots, \mathcal{S}_N^h)$, by considering

$$\mathcal{S}_i^h(y) \in \arg \max_{u_i^h \in \mathcal{U}_i^h} \left\{ hf_i(y, [u_i^h, u_{-i}^h]) + \frac{(1 - \rho_i h)}{2^p} \sum_{d=1}^{2^p} v_i(y + \delta_d(y, [u_i^h, u_{-i}^h])) \right\}, \quad (14)$$

where $[u_i; u_{-i}]$ denotes the pair of the control of player i (i.e., u_i) and the controls associated to the rest of players (denoted by $-i$), respectively. Next, we discretize the value function (5). More precisely, given $y \in X$, we consider $v_i^h : X \rightarrow \mathbb{R}$ defined by

$$v_i^h(y) = \max_{u_i^h \in \mathcal{U}_i^h} J_i^h(y, [u_i^h, u_{-i}^h]), \quad (15)$$

In that case, Equation (15), for players $i = 1, \dots, N$, can be rewritten in its feedback form, as

$$v_i^h(y; \mathcal{S}^h) = h \sum_{k=0}^{\infty} e^{-r_i k} f_i(x_k^*, \mathcal{S}^h(x_k^*)), \quad (16)$$

with, $x_0^* = y$, $x_{k+1}^* = x_k^* + \delta_d(x_k^*, \mathcal{S}^h(x_k^*))$ and $\delta_d = hg(x_k^*, \mathcal{S}^h(x_k^*); \theta) + \sigma(x_k^*) \begin{pmatrix} \pm\sqrt{h} \\ \dots \\ \pm\sqrt{h} \end{pmatrix}$.

Hence, $\mathcal{S}_*^h = (\mathcal{S}_{*,1}^h, \dots, \mathcal{S}_{*,N}^h)$ is a Discrete Stochastic Feedback Nash if the following property holds:

$$J_i^h(y; \mathcal{S}_*^h) \geq J_i^h(y; [u_i^h, \mathcal{S}_{*,-i}^h]) \text{ for each } y \in X \text{ and } u_i^h \in \mathcal{U}_i^h, \quad (17)$$

In this work, we aim to get an approximation of v_i^h satisfying (16), denoted by $v_{i,j}^h$, by generating a time and spatial discretization of the HJB Equation 11.

More precisely, we consider a set of Q arbitrary points $\{y_j\}_{j=1}^Q \in \hat{X}$, where \hat{X} is a discrete version of X (see Section 3.2 for a particular discrete scheme). Then, we consider $y^\# = (y_j + \delta_d(y_j, [u_i^h, \mathcal{S}_{-i}^h]))$ and $\bar{y}^\# = (y_j + \bar{\delta}_d(y_j, [u_i^h, \mathcal{S}_{-i}^h]))$, two points not necessarily in \hat{X} . For those points, we define the following discrete version of (11)

$$\rho_i v_{i,j}^h(y_j; \mathcal{S}_{*,-i}^h) = \max_{u_i^h \in \mathcal{U}_i^h} \left\{ hf_i(y_j, [u_i^h, \mathcal{S}_{*,-i}^h]) + \frac{(1 - \rho_i h)}{2^p} \sum_{d=1}^{2^p} I[V_i](y_j + \delta_d(y_j, [u_i^h, \mathcal{S}_{*,-i}^h])) \right\}$$

where $I[V_i](\cdot)$ is an approximation of $v_{i,j}^h(y^\#; \mathcal{S}_{*,-i}^h)$ at points $y^\#$ computed by a collocation algorithm (see, for instance, [34]) using a mesh-free method based on scattered nodes. In this work, $I[V_i](\cdot)$ is of the form

$$I[V_i](y^\#) = \sum_{j=1}^P \lambda_j \Phi(\|y^\# - y_j\|_2), \quad y^\# \in \Omega \quad (18)$$

where $\lambda_j \in \mathbb{R}$, $\|\triangle\|_2$ is the Euclidean norm and $\Phi(r)$ is a real-valued Radial Basis Function (see, e.g., [13]). Here, we use the Gaussian RBF, given by

$$\Phi(r) = \exp\left(-\frac{\|y^\sharp - y_j\|_2}{2\sigma^2}\right) \quad (19)$$

with $\sigma^2 > 0$. In order to determine λ_j , for $j = 1, \dots, Q$, we use the interpolation conditions

$$A\bar{\lambda}_i = V_i, \quad (20)$$

where $A \in \mathcal{M}_{Q \times Q}$ such that $[A]_{jl} = \Phi(\|y_j - y_l\|)$, for $j, l = 1, \dots, Q$, $\bar{\lambda}_i = [\lambda_1, \dots, \lambda_Q]^T$ and $V_i = [v_i^h(y_1), \dots, v_i^h(y_Q)]^T$ for $i = 1, \dots, N$.

3.2 Pseudocode of the considered algorithm

Here, for describing our algorithm, we use similar notations as the ones proposed in our previous work [23]. Regarding this previous work, to deal with the stochastic component added to the model, we introduce new steps and instructions to the deterministic version of RaBVIItG.

As in the deterministic case, the algorithm is mainly based on two fixed-point schemes. To describe the pseudocode presented below, we first need to introduce some useful vector and matrix notations. More precisely, the values of the game for all the players computed in the set of points \tilde{X} are stored in an array called V , given by

$$V = [v_1^h(y_1), \dots, v_1^h(y_Q), \dots, v_N^h(y_1), \dots, v_N^h(y_Q)]^T \in \mathbb{R}^{Q \cdot N},$$

with $v_i^h(\cdot) \in \mathbb{R}^N$.

We also store in the array Λ the values associated to the controls of each player as follows

$$\Lambda = (\Lambda_1, \dots, \Lambda_i, \dots, \Lambda_N) \in \mathbb{R}^{Q \cdot (m_1 + \dots + m_N)},$$

with $\Lambda_i = \left[(u_i^1, \dots, u_i^{m_i})_1, \dots, (u_i^1, \dots, u_i^{m_i})_Q \right]^T \in \mathbb{R}^{Q \cdot m_i}$, where $m_i \in \mathbb{N}$ is the number of control variables of player i .

We also define

$$\Lambda_{-i} = (\Lambda_1, \dots, \Lambda_{i-1}, \Lambda_{i+1}, \dots, \Lambda_N) \in \mathbb{R}^{Q \cdot (m_1 + \dots + m_{-i} + m_{i+1} + \dots + m_N)}.$$

Additionally, let F be a vector that contains the values of the benefit functions of each player at every data points and given by

$$F(\Lambda) = \left[(f_1(x, \Lambda))_1, \dots, (f_1(x, \Lambda))_p, \dots, (f_N(x, \Lambda))_1, \dots, (f_N(x, \Lambda))_p \right]^T \in \mathbb{R}^{Q \cdot N},$$

where $x \in X$. Finally, we introduce the operators, $T = (T_1, \dots, T_{Q \cdot N}) : \mathbb{R}^{Q \cdot N} \rightarrow \mathbb{R}^{Q \cdot N}$, and $G_i = (G_{i,1}, \dots, G_{i,Q \cdot N}) : \mathbb{R}^{Q \cdot N} \rightarrow \mathbb{R}^{Q \cdot N}$, with

$$T_j(\mathbf{V}; \Lambda) = h\mathbf{F}(\Lambda)_j + \frac{(1 - \rho_i h)}{2^p} \sum_{d=1}^{2^p} (I[\mathbf{V}](\Lambda_d))_j, \quad j = 1, \dots, Q \cdot N$$

and

$$G_{i,j}(\Lambda_i; \mathbf{V}_i) = \arg \max_{\Lambda_i \in U_i} (J_i(y, [\Lambda_{i,j}, \Lambda_{-i,j}])) \quad i = 1, \dots, N, \quad j = 1, \dots, Q \cdot N.$$

In the previous expression, $I[\mathbf{V}](\Lambda) = (A \cdot \bar{\lambda}_1 \quad \dots \quad A \cdot \bar{\lambda}_N) \in \mathbb{R}^{Q \cdot N}$, is a $Q \cdot N$ vector where $\bar{\lambda}_i$ and A are defined in (20).

Considering those notations, we define the following iterative processes:

- **Game Iteration:** let $s = 0, 1, 2, \dots$ and $\Lambda_i^0 \in U_i$. For each player $i = 1, \dots, N$, we generate Λ_i^{s+1} , a candidate to optimal policy at the step $s+1$ of this process, following

$$(\Lambda_i^{s+1})_j = \beta (\Lambda_i^s)_j + (1 - \beta) \arg \max_{\Lambda_i \in U_i} \left(hF([\Lambda_i, \Lambda_{-i}^s])_j + \frac{(1 - \rho_i h)}{2^p} \sum_{d=1}^{2^p} I[V_i^r]([\Lambda_i, \Lambda_{-i}^s]_d)_j \right) \quad (21)$$

with $j = 1, \dots, Q$, $\beta \in [0, 1]$, a weight coefficient (see, e.g., [31]).

We iterate this process until satisfying the stopping criterium $\|\Lambda^{s+1} - \Lambda^s\| < \epsilon_1$, with $\epsilon_1 > 0$ given. Once this criterium is satisfied, we consider the following candidate to the Stochastic Feedback-Nash equilibrium

$$\Lambda^* = (\Lambda_1^{s+1}, \dots, \Lambda_N^{s+1}).$$

Starting from Λ^* , we perform the Value Iteration explained below.

- **Value Iteration:** Once we have obtained Λ^* at the end of the previous Game Iteration step, we update the value function at step $r+1$ (with $V_i^0 \in U_i$ given, for $i = 1, \dots, N$), as following

$$(V_i^{r+1})_j = \left\{ hF(\Lambda^*)_j + \frac{(1 - \rho_i h)}{2^p} \sum_{d=1}^{2^p} I[V_i^r](\Lambda^*)_d)_j \right\} \quad (22)$$

with $j = 1, \dots, Q \cdot N$.

Considering the Game Iteration and Value Iteration briefly described above, we now introduce in detail a pseudo-code version of our algorithm.

STEP 1. Initialize all parameters and counters:

- Set: N the number of players; h , the time-step; ρ_i , the discount parameter of player i ; $\beta \in [0, 1]$ the update parameter for relaxation algorithm in *Game Iteration*. ; and $It_1max \in \mathbb{N}$ and $It_2max \in \mathbb{N}$, the maximum number of iterations for the while loops defined below.
- Initialize: $r = 0$ (i.e., the *value iteration* step); $s = 0$ (i.e., the *game iteration* step); and $tolV \approx 0$ and $tolG \approx 0$, the tolerance value for the stopping criteria of the Value and Game Iterations, respectively.
- Define U_i and X .
- Define the variance-covariance matrix $\sigma(\cdot)$, where $\sigma(\cdot) \in \mathcal{M}_{P \times P}$ by using the experimental data.

STEP 2. Build an initial approximation:

- Generate a set \mathcal{X} of Q random scattered points for each player, with an uniform distribution obtained from \tilde{X}

$$\mathcal{X} = ((\mathcal{X}_1)_1, \dots, (\mathcal{X}_1)_Q, \dots, (\mathcal{X}_N)_1, \dots, (\mathcal{X}_N)_Q) \in \mathbb{R}^{Q \cdot N}$$

- Generate a set of convenient initial values and controls:

$$\mathbf{V}^r = ((V_1)_1, \dots, (V_1)_Q, \dots, (V_N)_1, \dots, (V_N)_Q) \in \mathbb{R}^{Q \cdot N},$$

and

$$\Lambda^s = (\Lambda_1, \dots, \Lambda_i, \dots, \Lambda_N) \in \mathbb{R}^{Q \cdot (m_1 + \dots + m_N)},$$

with $\Lambda_i \in U_i \subset \mathbb{R}^{m_i}$. If there is no a priori information, they are set to zero.

- We define \mathbf{A} , using radial basis functions (here, we apply Equation (19))

$$(\mathbf{A})_{jl} = \Phi(\|\mathcal{X}_j - \mathcal{X}_l\|), \text{ for } j = 1, \dots, Q \text{ and } l = 1, \dots, Q,$$

- Determine $\bar{\lambda}_i^0$ (for $i = 1, \dots, N$) satisfying (see (20)):

$$\mathbf{A}\bar{\lambda}_i^0 = \mathbf{V}_i^r$$

STEP 3. Perform the Game Iteration and Value Iteration loops

WHILE $\text{dif}V > \text{tol}$ and $r < It_1\text{max}$

WHILE $\text{dif}G > \text{tol}G$ and $s < It_2\text{max}$

- For $j = 1, \dots, Q \cdot N$, perform:

- * Set $y = \mathcal{X}_j$
- * For each player $i = 1, \dots, N$:
 - Set y^\sharp according to the stochastic scheme

$$y^\sharp = (y_{\text{gi},1}^\sharp, \dots, y_{\text{gi},2^p}^\sharp) = y + hg(y, [\Lambda_i, \Lambda_{-i}^s]_j; \theta) + \sigma(y) \begin{pmatrix} \pm\sqrt{h} \\ \dots \\ \pm\sqrt{h} \end{pmatrix}.$$

- Compute $[\mathcal{A}_i]_d = \Phi(\|\mathcal{X}_j - y_d^\sharp\|)$, for $d = 1, \dots, 2^p$
- Compute $(\Lambda_i^{s+1})_j = \beta(\Lambda_i^s)_j - (1 - \beta)\arg\max_{\Lambda_i \in U_i} \left(hF([\Lambda_i, \Lambda_{-i}^s]_j) + \frac{(1-\rho_i h)}{2^p} \left[\sum_{d=1}^{2^p} [\bar{\lambda}_i^r \mathcal{A}_i]_d \right]_j \right)$
- Set

$$(\mathbf{\Lambda}^{s+1})_j = (\Lambda_1^{s+1}, \dots, \Lambda_i^{s+1}, \dots, \Lambda_N^{s+1})_j$$

- * Compute $\text{dif}G = \|\mathbf{\Lambda}^{s+1} - \mathbf{\Lambda}^s\|$.
- SET $s = s + 1$

END WHILE

- Define $(\mathbf{\Lambda}^*)_j = (\Lambda_1^{s+1}, \dots, \Lambda_i^{s+1}, \dots, \Lambda_N^{s+1})_j$

- Set $y^\sharp = (y_{\text{vi},1}^\sharp, \dots, y_{\text{vi},2^p}^\sharp) = y + hg(y, [\mathbf{\Lambda}^*]_j; \theta) + \sigma(y) \begin{pmatrix} \pm\sqrt{h} \\ \dots \\ \pm\sqrt{h} \end{pmatrix}$

- Get $[\mathcal{A}_i^*]_d = \Phi(\|\mathcal{X}_j - y_{*d}^\sharp\|)$, for $d = 1, \dots, 2^p$

- Update the value function (Value Iteration process): $(\mathbf{V}_i^{r+1})_j = \left\{ hF(\mathbf{\Lambda}^*)_j + \frac{(1-\rho_i h)}{2^p} \left[\sum_{d=1}^{2^p} [\bar{\lambda}_i^t \mathcal{A}_i^*]_d \right]_j \right\}$

- Update $\bar{\lambda}_i^{r+1}$ by solving

$$\mathcal{A}_i \bar{\lambda}_i^{r+1} = \mathbf{V}_i^{r+1}$$

- Compute $\text{dif}V = \|\mathbf{V}^{r+1} - \mathbf{V}^r\|$
- $r = r + 1$

– $s = 0$.

END WHILE

STEP 4. Compute the outputs:

- In cases for which the algorithm converges, we return the equilibrium policies $\Lambda_1^*, \dots, \Lambda_N^*$ and their associated optimal values, V_1^*, \dots, V_N^* , depending both on \mathcal{X} .

4 Numerical Experiments

In this Section, we aim to solve, by using RaBVIt-SG, a 4-players Pricing- Advertising differential game in a Feedback Nash equilibrium, considering both stochastic and deterministic frameworks. This problem is based on a real business problem from four companies competing with a similar cosmetic product sold in Spanish supermarkets.

In order to show the importance of the number of players in the studied problem, we first analyze the competition between two players. Then, we solve the full problem considering all the competitors.

In Section 4.1, we present and analyze the data related to those problems. In Section 4.2, we detail the dynamical systems associated to the evolution of the sales, price and advertising and the parameter values of the models. Finally, in Sections 4.3 and 4.4, we discuss the results obtained with RaBVIt-G and RaBVIt-SG and analyse the effect of the number of players in the final outputs.

4.1 Dataset of the Problem

The data set presented below is confidential. Here, only partial information will be given. It contains the following inputs:

- $S_i(t)$: Sales in equivalent units (i.e., the total volume sold) of company i at week t .
- $p_i(t)$: price in Euro (€) per unit sold by company i at week t .
- $a_i(t)$: TV advertising expenditures in Euro (€) of company i at week t .

A graphical representation of those data is given in Figures 1-3.

Figure 1: Data of the problem presented in Section 4: $S_i(t)$, Sales in equivalent units of company i at week t .

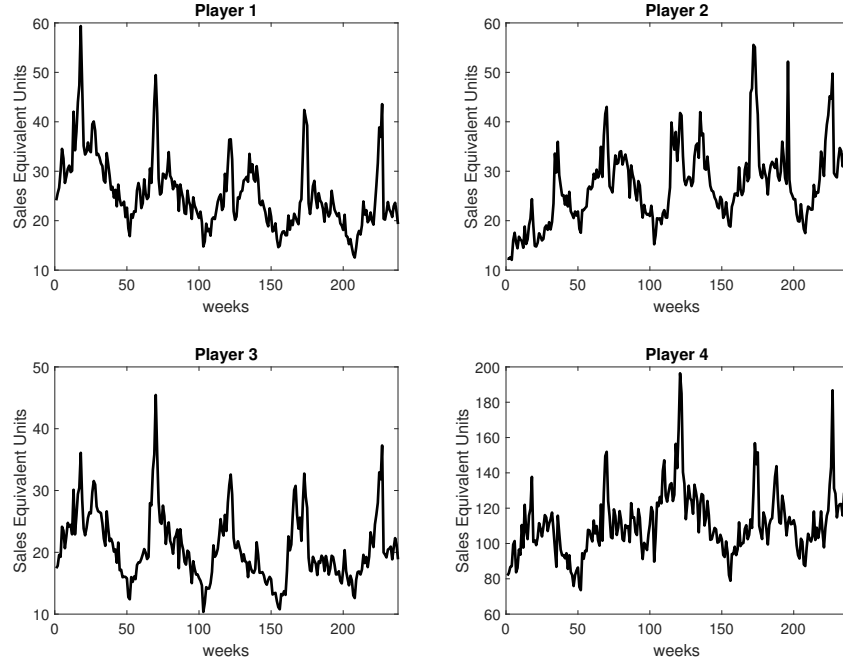


Figure 2: Data of the problem presented in Section 4: $p_i(t)$, price (€) per unit sold by company i at week t .

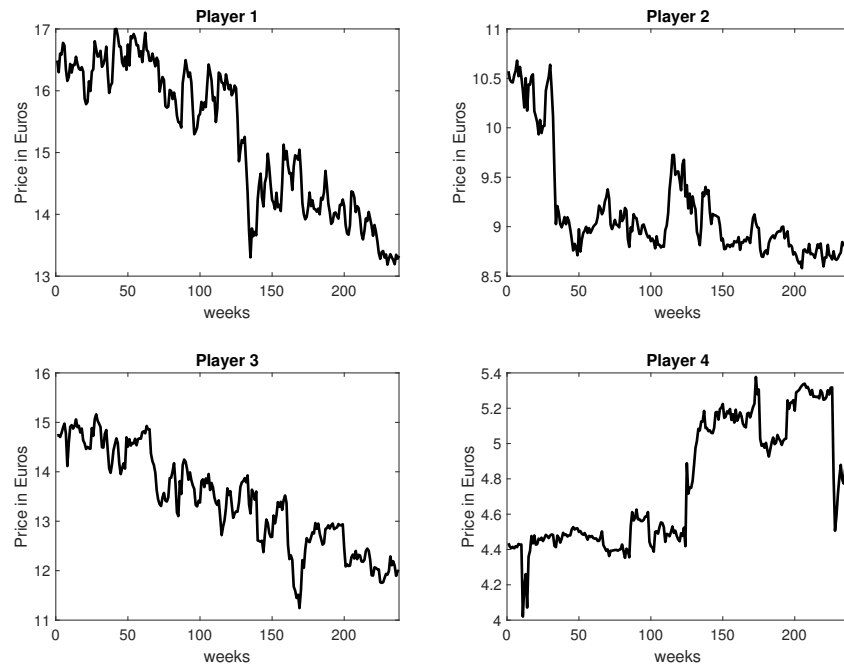
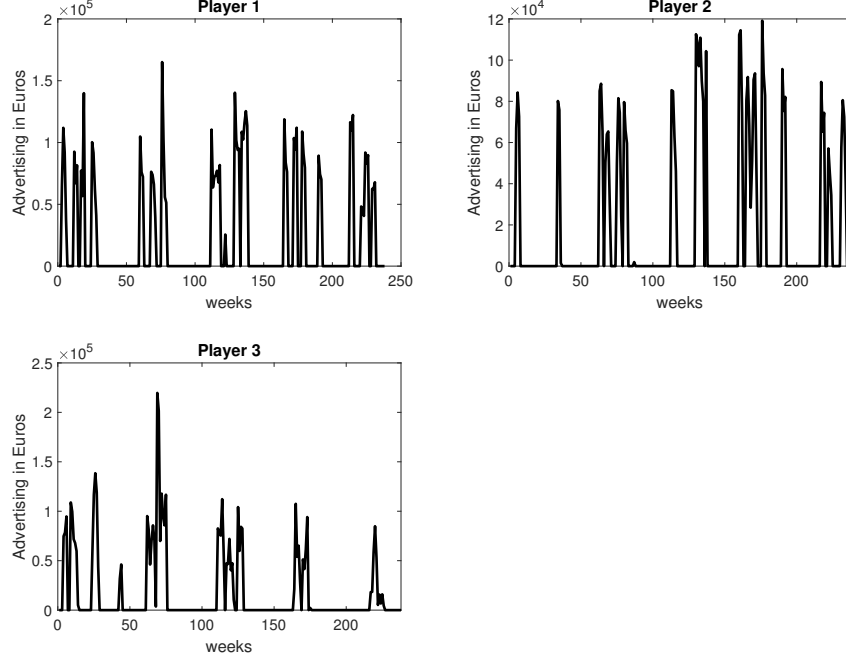


Figure 3: Data of the problem presented in Section 4: $a_i(t)$, TV advertising expenditures (€) of company i at week t .



As we can see on Figure 1, sales exhibit a seasonal pattern, meaning that during some weeks consumers follow different purchase decisions (generally affected by holidays, special events, etc. See, for instance, [44]). Omitting **P4** (as it is a *private label*) we observe a strong competition in sales between Player 1 (**P1**) and Player 2 (**P2**). **P1** starts being the sales leader up to week 50. From week 100 until the end of the sample, **P2** surpasses **P1**, competing with Player 4 (**P4**) the sales leadership up to the end of the sample.

Focusing on Figure 2, we can see that **P1**, **P2** and **P3** have a similar decreasing slope price strategy. On the opposite, the price of **P4** exhibit a general increasing behaviour. Additionally, **P1** is the most expensive competitor, whereas **P4**, which is the *private label* sold in supermarkets, has the lowest price strategy.

Finally, regarding Figure 3, **P1** has the most TV expenses, followed by **P2** and **P3**. **P4** has no TV presence. This dataset is processed as following:

- We use the first 200 data points (i.e., weeks), of a total of 250 points, to estimate the parameters of the dynamical system (see Section 4.2) by using a standard Time Series Multivariate Model (see, e.g., [19]). We call this set the "training" set. The rest of the available data (i.e., the last 50 weeks) is called the "testing" set. Then, considering this "training set", we run the RaBVIt-G and RaBVIt-SG in order to estimate optimal strategies within this sample.
- Next, we simulate the next 50 weeks and apply the strategies obtained from the "training" set to the "testing" set. In particular, we compare the efficiency of the results returned by applying the strategy proposed by RaBVIt-G and RaBVIt-SG with the real observations.

4.2 Parameterization of the Dynamical System

To our knowledge, there are three main popular optimal control (and their differential game versions) models for advertising: Nerlove-Arrow ([36]), Vidale- Wolfe ([48]) and Lanchester ([30]). Moreover, there exists several surveys with extensions and advances related to these models (see, for instance, [26], [22], [28]). From a general point of view, those models are well suited to work with sales, market shares, goodwill and advertising in a two players context (see, for instance, [43]). However, even if there exists some adaptations of these models including the pricing variable, we have numerically observed that they fit poorly the data described in Section 4.1.¹

For those reasons, here, we consider a model, presented in Equation (23), inspired from the work developed in [20] and [1], that takes into account the following multiplicative effects:

- The own sales of Company i and the sales of its competitors (i.e., Company j , with $j \neq i$) have effects on the sales of Player i .
- The advertising/pricing strategies of Company i and its competitors have effects on the sales of Player i . So that, we can estimate the advertising effects of competitors on a Player (see, for instance, [17, 18, 9]).
- The variation of sales depends on the level of all the variables. In general (see, e.g., [20]), this relationship is not linear, assuming that costumers do not react proportionally to price and advertising policies.

Considering those multiplicative effects, we present the following stochastic dynamical system

$$\begin{cases} \frac{\partial S_1(t)}{\partial t} = \prod_{i=1}^N S_i^{\alpha_{1i}}(t) p_i^{\beta_{1i}}(t) a_i^{\eta_{1i}}(t) e^{\sigma_1(S(t))dW_1}, \\ \dots \\ \frac{\partial S_N(t)}{\partial t} = \prod_{i=1}^N S_i^{\alpha_{Ni}}(t) p_i^{\beta_{Ni}}(t) a_i^{\eta_{Ni}}(t) e^{\sigma_N(S(t))dW_N}. \end{cases} \quad (23)$$

We note that, in the deterministic case, we set $\sigma_i = 0$, for $i = 1, \dots, N$.

We estimate the parameters of System (23) by using the "training" dataset described previously and a Time Series Multivariate Model (see, e.g., [19], [25], [6]). More precisely, this fitting procedure has been performed by considering the 2 and 4 Players problems, denoted as 2D and 4D models, respectively.

In Table 1, we present all the obtained parameter values, for the 2D and 4D models. More precisely:

- Parameters denoted by α are related to the short run "carry over" effect over time in percentage on sales. Indeed, α_{ii} measures the "re-purchase" intentions of customers and the firm's ability to attract new customers over time. Furthermore, α_{ij} measures the migration between firms: if $\alpha_{ij} < 0$ the customers migrate on the short-run from Firm i to Firm j , or vice-versa if $\alpha_{ij} > 0$ (i.e., in the Marketing literature this is known as a *inter-firm* effect see, for instance, [20]).
- Parameters denoted by β correspond to the sensitivity of prices (in percentage on sales). In particular, we consider the own price sensitivity β_{ii} (generally, $\beta_{ii} < 0$, meaning that customers will buy less products from Player i if this firm increases its own price) and the cross-price sensitivity β_{ij} of customers from Firm i to prices proposed by Firm j (usually, $\beta_{ij} > 0$, meaning that customers will buy more products from Player i if its competitor, Player j , increases its price).
- Parameters denoted by η correspond to the advertising efficiency parameter (in percentage of sales). Here, we consider the advertising efficiency parameter η_{ii} as the short-run impact of advertising products of Player i on its own sales. We also take into account different kind of *synergy* effects whether $\eta_{ij} > 0$ or $\eta_{ij} < 0$ (i.e., measuring the impact of advertising from Company i on the sales of Company j).

¹Experiments available under request.

We note that the values of all those parameters are highly sensitive to the number of players, so that, the solution of the game will change according to this number.

Table 1: Values of the parameters obtained by the calibration process described in Section 4.2, for the 4D and 2D models.

Player footnotesize Model	Carry over	Price Sensitivity	
P1	4D	$\alpha_{11} = 0.75, \alpha_{12} = -0.25, \alpha_{13} = -0.08, \alpha_{14} = -0.12,$	$\beta_{11} = -0.06, \beta_{12} = -0.81, \beta_{13} = 0.28, \beta_{14} = 0.04$
P2	2D	$\alpha_{22} = 0.58, \alpha_{23} = -0.10$	$\beta_{22} = -1.0, \beta_{23} = 0.23,$
	4D	$\alpha_{21} = -0.03, \alpha_{22} = 0.52, \alpha_{23} = -0.06, \alpha_{24} = -0.09,$	$\beta_{21} = 0.50, \beta_{22} = -1.20, \beta_{23} = 0.25, \beta_{24} = 0.04,$
P3	2D	$\alpha_{32} = -0.10, \alpha_{33} = 0.43,$	$\beta_{32} = -0.43, \beta_{33} = -0.94,$
	4D	$\alpha_{31} = 0.06, \alpha_{32} = -0.16, \alpha_{33} = 0.41, \alpha_{34} = -0.03,$	$\beta_{31} = 0.19, \beta_{32} = -0.62, \beta_{33} = -1, \beta_{34} = 0.06,$
P4	4D	$\alpha_{41} = -0.15, \alpha_{42} = -0.28, \alpha_{43} = 0.18, \alpha_{44} = 0.41,$	$\beta_{41} = -0.32, \beta_{42} = -1.08, \beta_{43} = 0.03, \beta_{44} = -0.84,$

Player	Model	Advertising effectiveness
P1	4D	$\eta_{11} = 0.016, \eta_{12} = 0.012, \eta_{13} = 0.0008$
P2	2D	$\eta_{22} = 0.016, \eta_{23} = 0.0016$
	4D	$\eta_{21} = 0.0012, \eta_{22} = 0.025, \eta_{23} = 0.001$
P3	2D	$\eta_{32} = -0.001, \eta_{33} = 0.033$
	4D	$\eta_{31} = 0.008, \eta_{32} = 0.001, \eta_{33} = 0.035$
P4	4D	$\eta_{41} = 0.016, \eta_{42} = 0.012, \eta_{43} = 0.0008$

Finally, we have considered the following variance-covariance matrices, estimated from our data, of the diffusion term:

- for the 2D model:

$$\sigma = \begin{pmatrix} 1.0905 & 0.0033 \\ 0.0033 & 1.0784 \end{pmatrix},$$

- for the 4D model:

$$\sigma = \begin{pmatrix} 1.1171 & 0.0090 & 0.0099 & 0.0074 \\ & 1.1255 & 0.0084 & 0.0076 \\ & & 1.1125 & 0.0074 \\ & & & 1.0936 \end{pmatrix}.$$

4.3 2 Players model

First, we discuss the numerical results obtained by solving the stochastic and deterministic versions of the Feedback Nash problem presented previously by considering only Players 2 and 3. The main goal of this experiment is to compare the solutions returned by RaBVIt-G with the stochastic and deterministic approaches with the actual data.

In Figures 4 and 5, we represent both stochastic and deterministic solutions for pricing and advertising Feedback Nash Equilibria. We can observe on those figures how considering stochastic processes produces smoother solutions (for instance, see the behaviour of the P2 and P3 advertising solutions and P3 pricing equilibrium policies).

As far as we know, there are not enough results in the literature analyzing the impact of uncertainties on marketing decisions (see, for instance [37]). In general, the existing models are not devoted to analyze the effects of the uncertainty on Marketing decisions. Even so, it is not easy to find models comparing Feedback controls in a deterministic and stochastic differential games

Figure 4: Feedback results for pricing

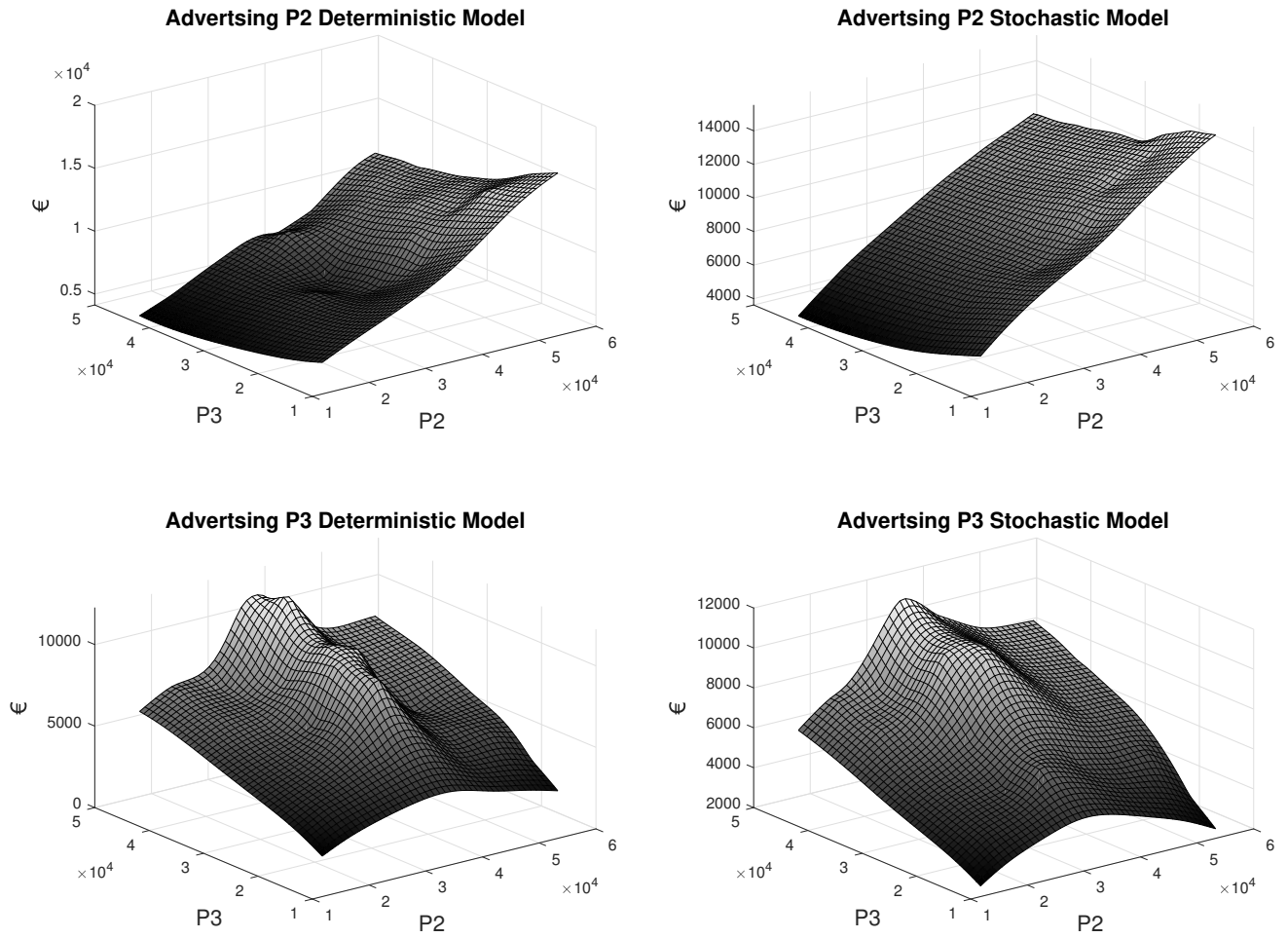
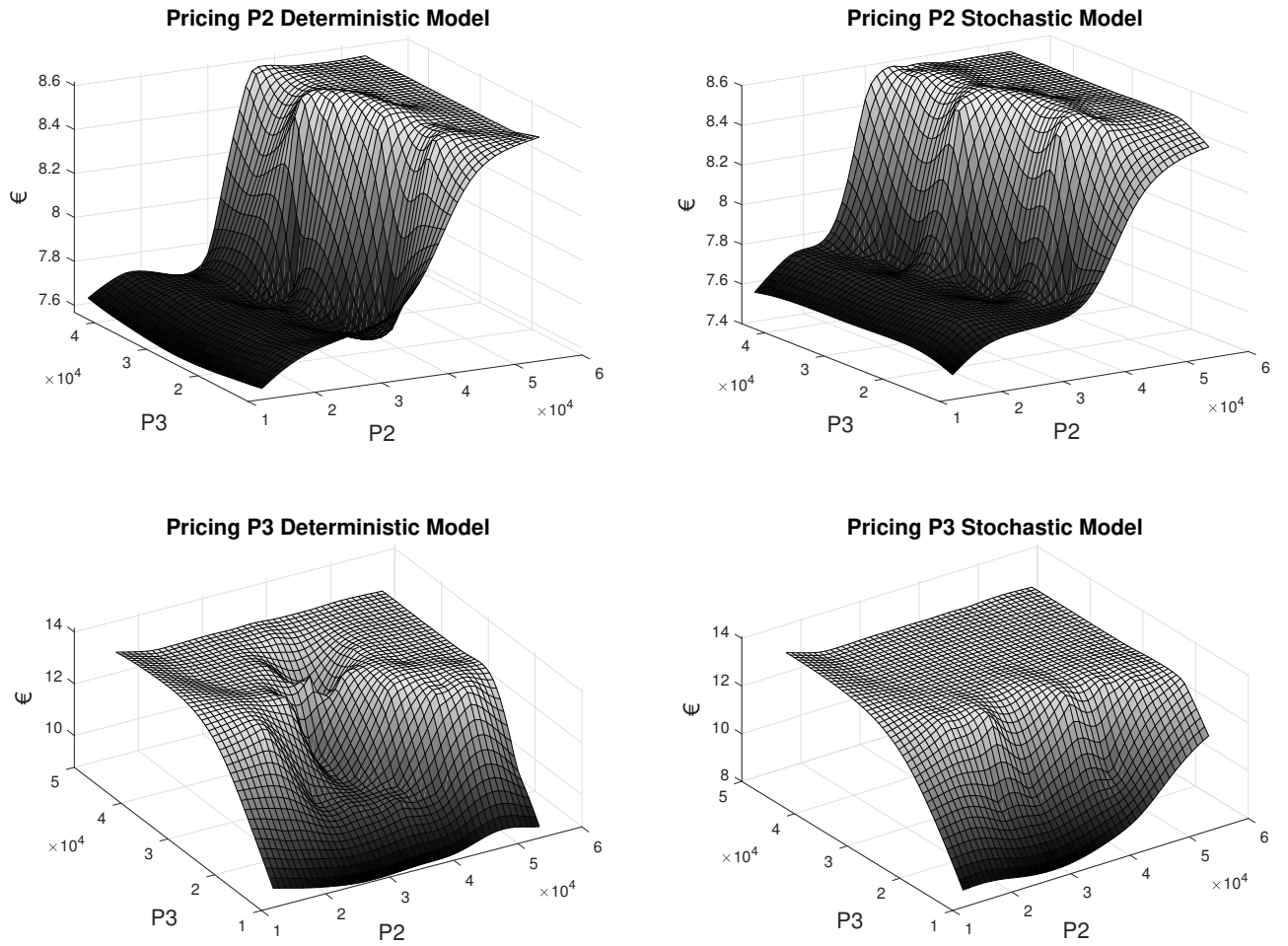


Figure 5: Feedback results for advertising



setting, trying to decide at the same time pricing and advertising policies. In other fields, such as in the literature associated to the fishing industry (see, e.g., [33, 39, 40]) some works focuses on the study of feedback harvesting policies. They compare both deterministic and stochastic feedback optimal controls, finding that stochasticity implies solutions with smoother functions (corresponding to the optimal feedback policy depending on the stock levels) than the deterministic models. Additionally, the greater the stochasticity level is, the more conservative (i.e., few changes in policies) are the optimal policies recommended by the model.

In our case, considering those results from fisheries, the greater the uncertainty on future sales is, the more conservative should be the optimal policy, meaning lower advertising expenditures and stickier prices. However, according to [37], it also should depend on the risk aversion of the player (i.e., the willing of the company to apply volatile policies) and the influence of the competitor's parameters in the model outputs. Hence, we should expect different conclusions according to the number of players in the model.

In addition, taking into consideration the "testing" set, we simulate a 52 weeks horizon benefit by using the Feedback function obtained with RaBVIt-G and RaBVIt-SG and the dynamical system fitted to the "training sample". So that, at each time step, we obtain an optimal advertising and pricing recommendation for each player and the predicted sales estimated by the model. Since the model is stochastic, we also have a set of possible paths for all the time series involved in the analysis. We show in Figure 6 the time-distribution of simulated benefits and the range of main percentiles of its distribution. Furthermore, we also report in the same plot the actual benefits. As we can see on this figure, for P2, the real benefits are almost always inside the distribution, while P3's real benefits are generally outside the main percentiles. Hence, considering the real data, we do not have strong evidence that both players are under a Nash Equilibrium. This is not an illogical result since, as observed in similar experiments performed in [20] with static methods, Nash Equilibrium are always extreme (providing abrupt changes in price and advertising expenditures with respect to historical central measures) and do not fit with the real strategies followed by some huge companies. As a particular illustration, in [16], the authors do not find a high correlation between Coca-Cola and Pepsi real strategies regarding the feedback Nash solution of the model.

Finally, in Table (2) we report some results regarding the advertising expenses, the prices and the benefits obtained during the 50 weeks horizon simulation using both stochastic and deterministic models. We also report some of the real data.

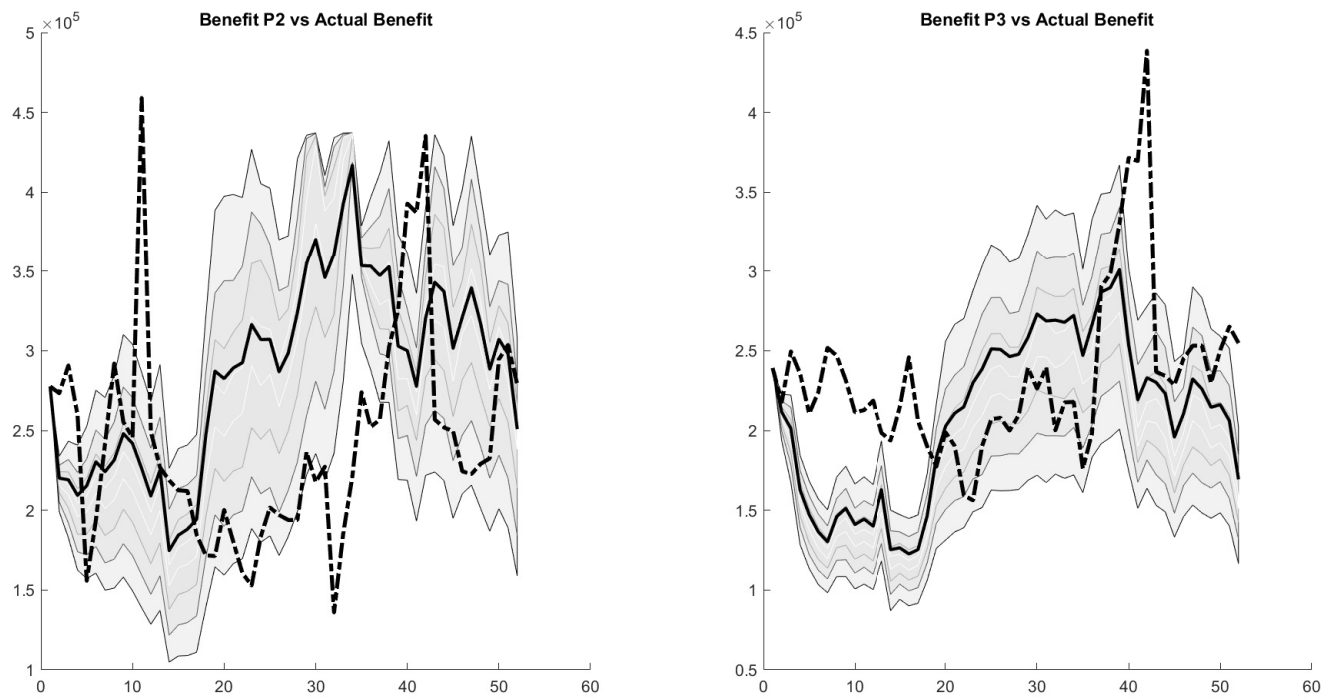
Table 2: Summary of 2 Players strategies obtained from deterministic model, stochastic model and real data. **Total Advertising** is the sum of the 52 weeks advertising (cost in euros), **Price** is the average price (in euros), **Benefit** is the sum of the profits (in thousand euros) and **Sales** are the total sales of the 52 weeks (in equivalent units). For the stochastic model, we provide into brackets the standard error.

		Total Advertising	Price	Benefit	Sales
P2	Stochastic	259960 (34076)	8.17 (0.24)	14.76M (4.12M)	34.90K (8.80K)
	Deterministic	270000	8.17	13.9M	35.2K
	Real	870000	8.77	12.9M	29.61K
P3	Stochastic	162840 (23981)	12.38 (0.68)	10.00M (3.54M)	16.44K (4.60K)
	Deterministic	170000	12.12	9.4M	17.9K
	Real	280000	12.31	12.4M	19.48K

For this particular simulation, we observe that:

- The total advertising expenses for the 52 weeks are both in the deterministic and stochastic models under the real policies. Indeed, we notice that, as said previously, the stochastic scenario assumes lower expenditures.

Figure 6: 2 players case: Time-distribution of simulated benefits for 52 weeks considering the "testing" set. Shaded parts correspond to different percentile ranges of the distribution of the benefits. Dotted lines are real data values and black line is the median



- Regarding the pricing policies, P2 applies the lowest prices (in both deterministic and stochastic models), which is similar to the real observation. Thus, P3, in this case, is the most expensive player.
- The effect on sales of both policies provides clearly higher sales (and also benefits) for P2 with respect to the real data. On the opposite, P3 loses sales and benefits.

However, those results may be incomplete and biased as we have omitted two players (P1 and P4) and, so, the obtained strategies could be non adequate.

4.4 4 Players model

In this section, we present and analyze the result returned by our algorithm when considering all the 4 players in the game. Hence, we use the full data set and the model parameters for 4 Players presented in Table 1. Then, we run RaBVIt-G and SG to estimate both deterministic and stochastic feedback Nash policies.

As stated in [15] in corollary 1, p. 1610, “using a 2-player advertising game instead of an N -player, by aggregating all other players into one rival, leads to overadvertising”. In addition, since in our framework we also consider price policies, we aim to study the effect on the solutions of taking into account all the players by comparing the results with the previous 2 players case.

In Figure 4.4, we plot the simulated 52-weeks time series result for the 4 players’ benefits and also report the real observations. First, it is important to note that, regarding the solution for 2 players depicted in Figure 6, the estimated variance values (i.e, σ) in the diffusion term are (for Players 2 and 3) higher than considering only two players (see Section 4.2). It affects the level of uncertainties of the model to compute the optimal policies.

In addition, the time distribution for benefits of Players 2 and 3 is quite different if we compare them to the 2 players case. This seems to indicate the importance of including all the players instead of a reduced set of players. This could be explained by the complex interaction between players strategies that are omitted when reducing the dimension of the game. Moreover, we observe that Benefits for Player 4 are close to real observation. We note that the Player 4 is a private label, which is not exposed to advertising. Thus, for P4, the output should be close to to the real observations.

In Table 3, we can observe the different strategies returned by our algorithm. In particular, we see that there are noticeable differences between price strategies with respect to the 2 players stochastic models. Indeed, Player 2 has to increase its price while P3 has to decrease it. With respect to advertising policies, we can see on Table (3) that there are quite noticeable differences between 2 and 4 players models. For instance, the 95% confidence interval for Player 2 advertising expenditures is model is (193468,326452) for 2 players while it is (126220,178620) for 4 players. In the case of Player 3 (114878,210802) for the 2 players model and (91898,156282) for the 4 players. As stated previously, those results seem to indicate that a 2 players model recommends to overspending on advertising. It is also noticeable that, in general, deterministic policies are inside the 95% confidence intervals.

When comparing to the real policies taken by the companies, only P3 is “overpricing”. The rest of players could increase their prices, if they play a Nash Equilibrium. From a general point of view, there is a trend to overspend on advertising if we compare actual values with model recommendations with respect to real observations. The average benefits achieved by the stochastic models are slightly higher than the ones obtained with deterministic models. The main reason is the change in pricing policies combined with less advertising budget.

Figure 7: 4 Players case: Time-distribution of simulated benefits for 52 weeks considering the "testing" set. Shaded parts correspond to different percentile ranges of the distribution of the benefits. Dotted lines are real data values and black line is the median .

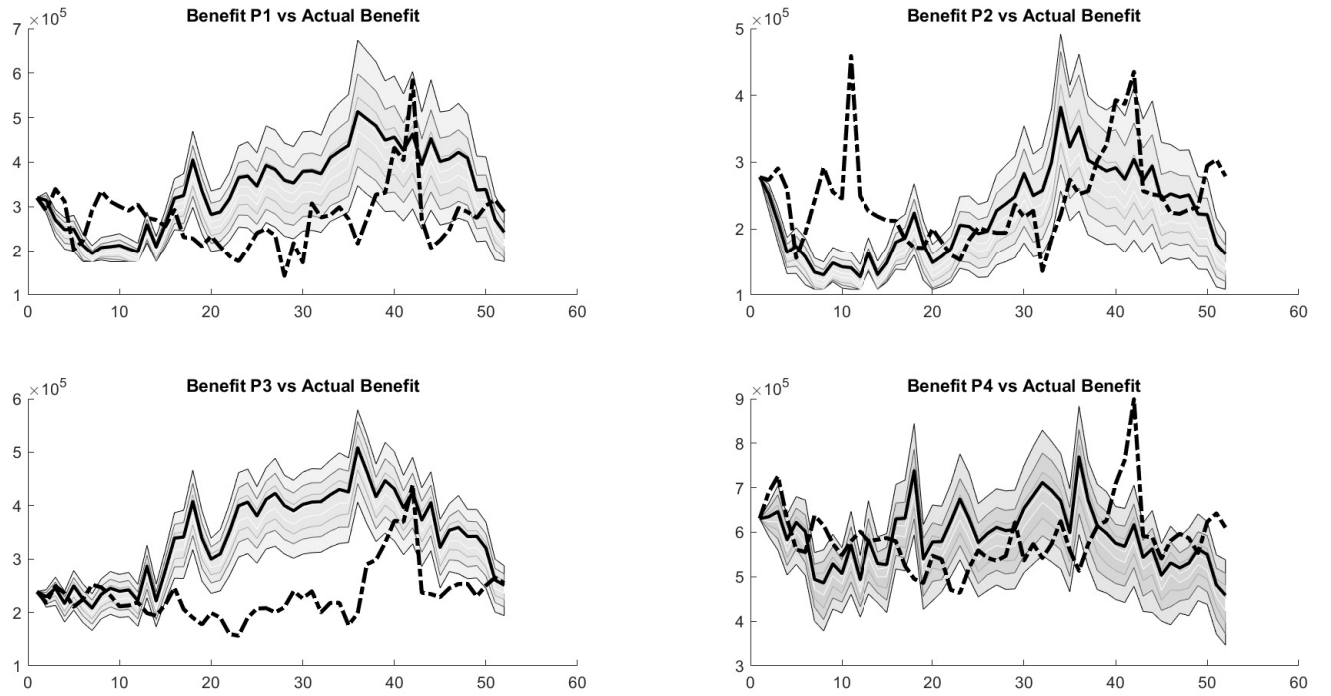


Table 3: Summary of the results obtained by the 4 and 2 players models. **Total Advertising** is the sum of the 52 weeks advertising (in euros), **Price** is the average price in euros, **Benefit** is the sum of the profits in euros and **Sales** are the total sales of the 52 weeks (in items). For the stochastic model, we provide into brackets the standard error. 2P refers to the results for the 2 players model.

		Total Advertising	Price	Benefit	Sales
P1	Stochastic	251960 (59181)	15.41 (0.002)	20.26M (8.05M)	25.58K (10K)
	Deterministic	250000	15.41	14.9M	25.85K
	Real	1.17M	13.80	14.5M	21.56K
P2	Stochastic 2P	259960 (33246)	8.17 (0.24)	14.76M (4.12M)	34.90K (8.80K)
	Deterministic 2P	270000	8.17	13.9M	29.61K
	Stochastic	152420 (13100)	9.08 (0.33)	16.20M (5.65M)	34.07K (11K)
	Deterministic	270000	8.17	13.9M	35.2K
	Real	870000	8.77	12.9M	29.61K
P3	Stochastic 2P	162840 (23981)	12.38 (0.68)	10.00M (3.54M)	16.44K (4.60K)
	Deterministic 2P	170000	12.12	9.4M	17.9K
	Stochastic	124090 (16096)	11.21 (0.52)	18.85M (5.31M)	32.00K (7.7K)
	Deterministic	104040	11.28	18.62M	31.44K
	Real	280000	12.31	12.4M	19.48K
P4	Stochastic	N/A	5.02 (0.31)	32.48M (9.37M)	122.9K (29.87K)
	Deterministic		5.24	33.50M	122.2K
	Real		5.12	31.1M	144.9K

5 Conclusions

In this work, we have developed a novel numerical algorithm (RaBVIt-SG) for solving multiplayer feedback-Nash stochastic differential games. This algorithm is as an extension of RABVIt-G, developed for deterministic cases.

RaBVIt-SG is an algorithm based on radial basis functions (known to work properly without mesh), value iteration (a classical approach in dynamic programming), and game iteration (to obtain, at each value iteration step, the corresponding stochastic feedback-Nash equilibrium). Additionally, we have considered a semi-Lagrangian discretization for the HJB equations involved in our differential games schemes. Next, we tackle the stochastic part of the problem by approximating the diffusion term with suitable Markov chains in a finite dimensional space.

To illustrate our approach, we have considered a marketing problem based on a real data set. In particular, we study the impact of the stochasticity in the optimal policies of the considered players and the impact of the number of players on the obtained solutions. Finally, we compare the solutions returned by RaBVIt-SG with the real observations and with the solutions returned by RaBVItG, considering a deterministic version of the problem. The results seem to indicate that: (i) in general the players do not follow a Feedback Nash policy, and (ii) the number of players is relevant in order to make decisions and estimate the optimal policies.

Acknowledgments

This work was carried out thanks to the financial support of the Spanish “Ministry of Economy and Competitiveness” under projects MTM2015-64865-P; the MOMAT UCM research Group (Ref. 910480); and the “Junta de Andalucía” and the European Regional Development Fund through the project P12-TIC301.

Appendix

Here, we present some numerical experiments to analyze the performance and the numerical convergence of RaBVIt-SG by considering two problems with analytical solution.

Test 1 - Stochastic Linear Quadratic

We consider the following two player ($N = 2$) scalar ($X \subseteq \mathbb{R}$) stochastic differential game for which we aim to minimize the following quadratic cost functional, with $i = \{1, 2\}$.

$$J_i(y, u) = \mathbb{E} \left\{ \int_0^\infty [q_i x(t)^2 + r_i u_i(t)^2] dt | x(0) = y \right\},$$

subject to the following stochastic linear differential equation:

$$dx(t) = [ax(t) + b_1 u_1(t) + b_2 u_2(t)] dt + \sigma x(t) dw(t),$$

where $x(0) = y$ and $y \in X$.

According to ([49]) the analytical feedback-Nash equilibria for this game are defined by the following linear expressions

$$u_1^*(t) = k_1 x(t), u_2^*(t) = k_2 x(t),$$

where k_1, k_2 are the solution to the following scalar Riccati algebraic equations:

$$2p_1(a + b_2 k_2) + q_1 + \sigma^2 p_1 + k_1 b_1 p_1 = 0, \quad k_1 = -\frac{1}{r_1} b_1 p_1,$$

$$2p_2(a + b_1 k_1) + q_2 + \sigma^2 p_2 + k_2 b_2 p_2 = 0, \quad k_2 = -\frac{1}{r_2} b_2 p_2.$$

In Table 4 we present the main results obtained by RaBVIt-SG from this particular problem. We use our algorithm for a set of time steps $h = \{0.1, 0.05, 0.01, 0.005\}$ and stochasticity levels $\sigma = \{0.1, 0.3, 0.5, 0.7, 0.8\}$. The state variable is discretized in 5 points in the domain $0 \leq x \leq 1$. We report on this table, the value

$$L_{error}^\infty = \max[k_1 x(t) - u_1, k_2 x(t) - u_2], \quad (24)$$

where u_1 and u_2 are the array of the optimal controls obtained by the algorithm. As we can see on this table, the error level decreases with the time step and is not significantly affected by the value of σ .

We note that, as it could be expected, the algorithm does not always converge. Indeed, it has been observed (for some experiments no reported here) that, for instance, for $\Delta t > 0.1$ or $\sigma > 0.9$ the convergence is not generally obtained. We also report the average CPU time obtained by using an Intel(R) Core(TM) i7-4600U CPU and 4 Gb of RAM.

Table 4: Results obtained with RaBVIIt-SG for Stochastic Linear Quadratic Problem. In each cell, we report two values: (i) the upper number represents the value of the L^∞ error (25) and (ii) the bottom number represents the number of iterations required to reach convergence.

	$\Delta t = 0.2$	$\Delta t = 0.1$	$\Delta t = 0.05$	$\Delta t = 0.01$
$\sigma = 0.1$	0.81 9	0.19 29	0.09 42	0.02 210
$\sigma = 0.3$	0.82 10	0.19 27	0.09 42	0.02 188
$\sigma = 0.7$	0.79 15	0.20 27	0.10 48	0.02 195
Avg CPU time	0.44	0.52	0.69	0.75

Test 2 - Vidale Wolfe Advertising Model

In literature associated to differential games in marketing (see [21]), the Vidale Wolfe model is used to find optimal feedback advertising policies. Its stochastic version corresponds to the following cost functions

$$J_1(x_0, u_1) = \mathbb{E} \left\{ \int_0^\infty [m_1 x(t) + c_1 u_1(t)^2] dt | x_0 \right\},$$

and

$$J_2(y_0, u_2) = \mathbb{E} \left\{ \int_0^\infty [m_2 y(t) + c_2 u_2(t)^2] dt | y_0 \right\},$$

subject to the following stochastic dynamic system (where x and y are the market shares)

$$\begin{aligned} dx &= [r_1 u_1 \sqrt{1-x} - r_2 u_2 \sqrt{x} - \delta(x-y)] dt + \sigma \sqrt{xy} dw, \\ dy &= [r_2 u_2 \sqrt{1-y} - r_1 u_1 \sqrt{y} - \delta(y-x)] dt - \sigma \sqrt{xy} dw, \end{aligned}$$

where $x(0) = x_0$ and $y(0) = 1 - x_0$ with $0 \leq x_0 \leq 1$ and $x(t) + y(t) = 1$ for all $t \geq 0$.

According to [42], the analytic solution is given by

$$u_1^*(x) = \beta_1 r_1 \frac{\sqrt{1-x}}{2c_1},$$

and

$$u_2^*(y) = \beta_2 r_2 \frac{\sqrt{1-y}}{2c_2},$$

where, in our case, $\beta_1 = \beta_2 = \frac{\sqrt{(\rho_i + 2\delta)^2 + \frac{12r_i^2}{4c_i} m_i - (\rho_i + 2\delta)}}{\frac{3r_i^2}{2c_i}}$.

We use the following parameter values (similar for the ones chosen in the literature): $\rho_1 = \rho_2 = 0.1, m_1 = m_2 = 1, c_1 = c_2 = 0.1, r_1 = r_2 = 1$ and $\delta = 0.5$. We also, test our algorithm by using three stochasticity levels (0.5, 1 and 1.5). However, we note that the explicit solution is independent to the stochasticity level.

We report on Table 5, the number of iterations required to reach convergence and the infinity-norm of the error (similar to the one used in the Test 1)

$$L_{error}^\infty = \max [u_1^* - u_1, u_2^* - u_2], \quad (25)$$

where u_1 and u_2 are the array of the optimal controls obtained by the algorithm. We also provide the average CPU time needed for convergence. We observe that, for $\sigma > 2$ or $\Delta t > 0.2$, the algorithm does not reach convergence.

Table 5: Results obtained with RaBVIt-SG for a version of Vidale-Wolfe model. In each cell, we report two values: (i) the upper number represents the value of the L^∞ error (25) and (ii) the bottom number represents the number of iterations required to reach convergence.

	$\Delta t = 0.2$	$\Delta t = 0.1$	$\Delta t = 0.05$	$\Delta t = 0.01$
$\sigma = 0.5$	0.0087 6	0.0109 12	0.0062 25	0.0043 121
$\sigma = 1$	0.015 6	0.0085 11	0.0032 24	0.0047 121
$\sigma = 1.5$	0.013 11	0.0052 12	0.0035 24	0.0046 120
Avg CPU time	6.4	9.3	19.2	64.7

References

- [1] M Berk Ataman, Harald J Van Heerde, and Carl F Mela. The long-term effect of marketing strategy on brand sales. *Journal of Marketing Research*, 47(5):866–882, 2010.
- [2] Tamer Başar. A tutorial on dynamic and differential games. In *Dynamic games and applications in economics*, pages 1–25. Springer, 1986.
- [3] Tamer Basar and Geert Jan Olsder. *Dynamic noncooperative game theory*, volume 23. Siam, 1999.
- [4] Olivier Bokanowski, Maurizio Falcone, Roberto Ferretti, Lars Grüne, Dante Kalise, and Hasnaa Zidani. Value iteration convergence of"-monotone schemes for stationary hamilton-jacobi equations. *Discrete and Continuous Dynamical Systems-Series A*, 35(9):4041–4070, 2015.
- [5] Fabio Camilli and Maurizio Falcone. An approximation scheme for the optimal control of diffusion processes. *ESAIM: Mathematical Modelling and Numerical Analysis-Modélisation Mathématique et Analyse Numérique*, 29(1):97–122, 1995.
- [6] Marnik G Dekimpe and Dominique M Hanssens. The persistence of marketing effects on sales. *Marketing science*, 14(1):1–21, 1995.
- [7] Chao Deng, Xudong Zeng, and Huiming Zhu. Non-zero-sum stochastic differential reinsurance and investment games with default risk. *European Journal of Operational Research*, 264(3):1144–1158, 2018.
- [8] Jacob Engwerda. *LQ dynamic optimization and differential games*. John Wiley & Sons, 2005.
- [9] Gary M Erickson. A differential game model of the marketing-operations interface. *European Journal of Operational Research*, 211(2):394–402, 2011.
- [10] Giorgio Fabbri, Fausto Gozzi, and Andrzej Swiech. Stochastic optimal control in infinite dimension. *Probability and Stochastic Modelling*. Springer, 2017.
- [11] Maurizio Falcone. Numerical methods for differential games based on partial differential equations. *International Game Theory Review*, 8(02):231–272, 2006.
- [12] Maurizio Falcone and Roberto Ferretti. *Semi-Lagrangian approximation schemes for linear and Hamilton-Jacobi equations*, volume 133. SIAM, 2013.

- [13] Gregory E. Fasshauer. *Meshfree approximation methods with MATLAB*, volume 6. World Scientific, 2007.
- [14] Avner Friedman. Stochastic differential games. *Journal of differential equations*, 11(1):79–108, 1972.
- [15] Gila E Fruchter. The many-player advertising game. *Management Science*, 45(11):1609–1611, 1999.
- [16] Gila E Fruchter and Shlomo Kalish. Closed-loop advertising strategies in a duopoly. *Management Science*, 43(1):54–63, 1997.
- [17] Hubert Gatignon. Competition as a moderator of the effect of advertising on sales. *Journal of Marketing Research*, 21(4):387–398, 1984.
- [18] Luca Grosset and Bruno Viscolani. Age-structured linear-state differential games. *European Journal of Operational Research*, 254(1):269–278, 2016.
- [19] James Douglas Hamilton. *Time series analysis*, volume 2. Princeton university press Princeton, NJ, 1994.
- [20] Dominique M Hanssens, Leonard J Parsons, and Randall L Schultz. *Market response models: Econometric and time series analysis*, volume 12. Springer Science & Business Media, 2003.
- [21] Alain Haurie, Jacek B Krawczyk, and Georges Zaccour. *Games and dynamic games*, volume 1. World Scientific Publishing Company, 2012.
- [22] Xiuli He, Ashutosh Prasad, Suresh P Sethi, and Genaro J Gutierrez. A survey of stackelberg differential game models in supply and marketing channels. *Journal of Systems Science and Systems Engineering*, 16(4):385–413, 2007.
- [23] Jorge Herrera de la Cruz, Benjamin Ivorra, and Ángel M Ramos. Rabvitg:an algorithm for solving a class of multi-players feedback nash differential games. *Optimization Online*, 2019.
- [24] Desmond J Higham. An algorithmic introduction to numerical simulation of stochastic differential equations. *SIAM review*, 43(3):525–546, 2001.
- [25] Maxwell K Hsu, Ali F Darrat, Maosen Zhong, and Salah S Abosedra. Does advertising stimulate sales or mainly deliver signals? a multivariate analysis. *International Journal of Advertising*, 21(2):175–195, 2002.
- [26] Jian Huang, Mingming Leng, and Liping Liang. Recent developments in dynamic advertising research. *European Journal of Operational Research*, 220(3):591–609, 2012.
- [27] Jian Huang, Mingming Leng, and Liping Liang. Recent developments in dynamic advertising research. *European Journal of Operational Research*, 220(3):591–609, 2012.
- [28] Steffen Jørgensen and Georges Zaccour. Developments in differential game theory and numerical methods: economic and management applications. *Computational Management Science*, 4(2):159–181, 2007.
- [29] Ricardo Josa-Fombellida and Juan Pablo Rincón-Zapatero. New approach to stochastic optimal control. *Journal of Optimization Theory and Applications*, 135(1):163–177, 2007.
- [30] George E Kimball. Some industrial applications of military operations research methods. *Operations Research*, 5(2):201–204, 1957.
- [31] Jacek B Krawczyk and Stanislav Uryasev. Relaxation algorithms to find nash equilibria with economic applications. *Environmental Modeling & Assessment*, 5(1):63–73, 2000.

- [32] Harold Kushner and Paul G Dupuis. *Numerical methods for stochastic control problems in continuous time*, volume 24. Springer Science & Business Media, 2013.
- [33] Sturla Furunes Kvamsdal, Diwakar Poudel, and Leif Kristoffer Sandal. Harvesting in a fishery with stochastic growth and a mean-reverting price. *Environmental and resource economics*, 63(3):643–663, 2016.
- [34] Nam Mai-Duy and Thanh Tran-Cong. Approximation of function and its derivatives using radial basis function networks. *Applied Mathematical Modelling*, 27(3):197–220, 2003.
- [35] Paola Mannucci. Nonzero-sum stochastic differential games with discontinuous feedback. *SIAM journal on control and optimization*, 43(4):1222–1233, 2004.
- [36] Marc Nerlove and Kenneth J Arrow. Optimal advertising policy under dynamic conditions. *Economica*, pages 129–142, 1962.
- [37] Dung Nguyen. *Marketing decisions under uncertainty*, volume 6. Springer Science & Business Media, 2012.
- [38] Bernt Oksendal. *Stochastic differential equations: an introduction with applications*. Springer Science & Business Media, 2013.
- [39] Diwakar Poudel and Leif K Sandal. Stochastic optimization for multispecies fisheries in the barents sea. *Natural Resource Modeling*, 28(3):219–243, 2015.
- [40] Diwakar Poudel, Leif K Sandal, Sturla F Kvamsdal, and Stein I Steinshamn. Fisheries management under irreversible investment: does stochasticity matter? *Marine Resource Economics*, 28(1):83–103, 2013.
- [41] Warren B. Powell. *Approximate Dynamic Programming: Solving the curses of dimensionality*, volume 703. John Wiley & Sons, 2007.
- [42] Ashutosh Prasad and Suresh P Sethi. Competitive advertising under uncertainty: A stochastic differential game approach. *Journal of Optimization Theory and Applications*, 123(1):163–185, 2004.
- [43] Ashutosh Prasad and Suresh P Sethi. Competitive advertising under uncertainty: A stochastic differential game approach. *Journal of Optimization Theory and Applications*, 123(1):163–185, 2004.
- [44] Sonja Radas and Steven M Shugan. Seasonal marketing and timing new product introductions. *Journal of Marketing Research*, 35(3):296–315, 1998.
- [45] Suresh P Sethi. Deterministic and stochastic optimization of a dynamic advertising model. *Optimal Control Applications and Methods*, 4(2):179–184, 1983.
- [46] Yuval Tassa and Tom Erez. Least squares solutions of the hjb equation with neural network value-function approximators. *IEEE transactions on neural networks*, 18(4):1031–1041, 2007.
- [47] Leigh Tesfatsion and Kenneth Juddn (Eds.). *Handbook of computational economics (Vol. 2)*. Elsevier, 2006.
- [48] ML Vidale and HB Wolfe. An operations-research study of sales response to advertising. *Operations research*, 5(3):370–381, 1957.
- [49] Haiying Zhou, Huainian Zhu, and Chengke Zhang. Linear quadratic nash differential games of stochastic singular systems. *Journal of Systems Science and Information*, 2(6):553–560, 2014.