Journal of Computational and Applied Mathematics 362 (2019) 41–54

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

Nash equilibrium approximation of some class of stochastic differential games: A combined Chebyshev spectral collocation method with policy iteration

Z. Nikooeinejad *, M. Heydari

Department of Mathematics, Yazd University, Yazd, Iran

ARTICLE INFO

Article history:
Received 26 August 2018
Received in revised form 10 May 2019

Keywords:
Nonzero-sum stochastic games
Nash equilibrium solution
Coupled HJ equations
Chebyshev spectral collocation method
Policy iteration
Operational matrix of derivative

ABSTRACT

This study relates to nonzero-sum stochastic games with perfect information. It proposes an efficient combined Chebyshev spectral collocation method (CSCM) with the policy iteration (PI) algorithm for solving nonlinear coupled Hamilton–Jacobi (HJ) equations. The proposed approach is comprised of two steps. First, the PI algorithm is used to reduce the nonlinear coupled HJ equations to a sequence of linear uncoupled PDEs. Then, these equations are approximated by the CSCM. The CSCM+PI is especially useful when the CSCM fails due to the increasing number of collocation points for solving the associated system of nonlinear algebraic equations. The main advantage of the resulting method is that it converts nonlinear coupled HJ equations to the systems of linear algebraic equations, which can be solved readily. Convergence analysis of this method is also provided in detail. To confirm the accuracy and validity of the proposed computational algorithm, several illustrative examples are presented.

1. Introduction

Decision-making in real-world problems often involves situations with two or more decision-makers. Also, they may consider uncertainties and nonlinearities. Researchers in the 1960s started working on what have been called stochastic differential (dynamic) games. These problems have drawn a great deal of attention. They have been widely applied to many problems in different fields ranging from physics to economics. The basic mathematical theory for stochastic differential games is established in many literatures [1–10].

In this paper, we confine ourselves to nonzero-sum stochastic games with perfect information. Also, we assume that all players act non-cooperatively and simultaneously. A feedback Nash equilibrium solution is desirable in this case. This solution can be obtained by solving the coupled second order HJ PDEs resulting from dynamic programming [8–11]. These equations do not have general closed-form solutions and are very difficult to solve for nonlinear problems. In most cases, the coupled HJ equations must be solved numerically.

One of the earliest computational methods for obtaining numerical solutions of stochastic optimal control and differential games is based on Markov chain approximation method (MCAM) [12–16]. The idea behind this method is to approximate the controlled diffusion process by an appropriate Markov chain on a finite state space. This method recomputes the transition matrix each time. Generally, it is not very easy to obtain numerical solution of stochastic differential games by the MCAM.

* Corresponding author.
E-mail addresses: znikooei@stu.yazd.ac.ir (Z. Nikooeinejad), m.heydari@yazd.ac.ir (M. Heydari).

https://doi.org/10.1016/j.cam.2019.05.014

A combined multigrid method with the PI algorithm has been developed to solve a class of zero-sum stochastic differential games [17]. This method seeks the solution on subdomains, and thus incurs the difficulty of dimensionality.

Adaptive dynamic programming (ADP) algorithms have been applied to solve a class of nonzero-sum deterministic games with infinite horizon [18–21]. In this approach, the PI algorithm and critic neural network (NN) are utilized to solve the coupled HJ equations. Because of the suitable choice of NN-activation functions, the number of neurons required, and the training time, implementing this approach can be difficult. To the best of the author's knowledge, this approach has not been applied for solving differential games with finite horizon.

In the last two decades, the spectral methods due to their extremely accuracy for solving ODEs and PDEs have been intensively studied [22–24]. The spectral methods based on the Chebyshev polynomials as basis functions are widely used because of their exponential convergence rate in the approximation of functions. In this approach, the solution of the problem is approximated by a linear combination of a finite set of Chebyshev polynomials. Using the derivative operational matrix and collocation points, the CSCM reduces the problem into a system of nonlinear algebraic equations. Compared to other numerical methods, it can be shown that the CSCM achieves high accuracy on the whole domain with relatively fewer grid points [25–28]. However, some difficulties may arise to implement on nonlinear problems. For instance, if we are not able to solve the system of nonlinear algebraic equations associated with the CSCM, then finding the approximate solution of the problem may fail as well.

Our aim in this paper, is to present a simple and efficient computational method for solving the coupled second order HJ equations arising in nonzero-sum stochastic games with finite horizon. This approach combines the PI algorithm and CSCM. First, the PI algorithm is used to reduce the nonlinear coupled HJ equations to a sequence of linear uncoupled PDEs. Then, these equations are approximated by the CSCM.

The remainder of the paper is organized as follows. In Section 2, we begin with the problem definition. Some preliminary details about Chebyshev polynomials are given in Section 3. In Section 4, we describe the technical difficulty of solving the coupled second order HJ equations by the CSCM. In this section, we develop the CSCM+PI for these equations and also, convergence results for the method are discussed. Finally, a brief conclusion is given in Section 6.

2. Stochastic differential games with perfect information

Let $(\Omega, \mathcal{F}, P)$ be a filtered probability space with filtration $\{\mathcal{F}_t\}$. Consider a stochastic formulation for n-person nonzero-sum differential game on a finite time interval $[t_0, t_f]$ described by the following Itô-sense stochastic differential equation (SDE):

$$dx = F(t, x, u_1, \dots, u_n)dt + \sigma(t, x)dw(t), \;\; x(t_0) = x_0, \;\; t \in [t_0, t_f], \tag{1}$$

where

$$F(t, x, u_1, \dots, u_n) = f(t, x) + \sum_{i=1}^{n} g_i(t, x)u_i, \tag{2}$$

and $\sigma(t, x)$ are the drift and diffusion terms, respectively. Also, x_0 is a given vector in $\mathbb{R}^k$, $x \in \Sigma \subset \mathbb{R}^k$ is the state process of system, $u_i \in U_i \subset \mathbb{R}^{m_i}$ is a feedback control function implemented by ith player for $i = 1, 2, \dots, n$.

Definition 2.1 (*[29], pp. 130*)**.** A control process $u_i = u_i(t, x)$ is admissible if it is adapted to $\{\mathcal{F}_t\}$ and satisfies a Lipschitz condition on the closure of $[t_0, t_f] \times \Sigma$.

Let $\mathcal{U}_i$ be the set of the admissible control functions $u_i(t, x)$ with values in U_i, function $F : \mathbb{R}\times\mathbb{R}^k\times\mathbb{R}^{m_1}\times\cdots\times\mathbb{R}^{m_n} \to \mathbb{R}^k$, function $\sigma : \mathbb{R} \times \mathbb{R}^k \to \mathbb{R}^{k\times k}$, and the vector $w = [w_1, w_2, \dots, w_k]^T$ is a k-dimensional Wiener process.

The payoff functionals associated with each player are defined as:

$$J_i(t, x, u_i, u_{-i}) = E\Big[\int_{t_0}^{t_f} L_i(t, x, u_i, u_{-i})dt + q_i(t_f, x(t_f))\Big], \tag{3}$$

where

$$L_i(t, x, u_i, u_{-i}) = x^T Q_i x + \sum_{j=1}^{n} u_j^T R_{ij} u_j, \tag{4}$$

$E[.]$ denoting the expectation operation, matrices $Q_i \geq 0$, $R_{ii} > 0$, and $R_{ij} \geq 0$ are symmetric. The non-own control vector $u_{-i} = (u_j : j \neq i)$ denotes strategies vector of all players except for Player i, and when the game terminates at time t_f, Player i receives a terminal payment of q_i.

We assume that each player wants to minimize his own payoff.

Definition 2.2 (*[30], pp. 2*)**.** The players' control actions $u_i^*(.) \in \mathcal{U}_i$, $i = 1, 2, \dots, n$ constitute a feedback Nash equilibrium if for all $(t, x) \in [t_0, t_f] \times \mathbb{R}^k$ and for all admissible control actions $u_i(.) \in \mathcal{U}_i$, $i = 1, 2, \dots, n$ the following inequalities hold:

$$J_i(t, x, u_i^*, u_{-i}^*) \leq J_i(t, x, u_i, u_{-i}^*), \quad \forall u_i \in \mathcal{U}_i. \tag{5}$$

The value functions $V_i(t,x),\ i=1,2,\dots,n$ associated with the Nash equilibrium point (u_i^*, u_{-i}^*) by starting at t and state x are defined as:

$$V_i(t,x) = J_i(t,x,u_i^*,u_{-i}^*).$$

Let the value functions $V_i(t,x) \in C^{1,2}([t_0,t_f]\times\Sigma), i=1,2,\dots,n$. By applying the principle of optimality and Itô calculus, an n-tuple of feedback strategies $\{u_i^*(t,x)\in\mathcal{U}^i; i=1,2,\dots,n\}$ provides a feedback Nash equilibrium solution to the game (1)–(4), if there exist smooth functions $V_i : [t_0,t_f]\times\Sigma\to\mathbb{R}$, for $i=1,2,\dots,n$ satisfying the following second order parabolic PDEs:

$$\begin{aligned}
&-\frac{\partial}{\partial t}V_i(t,x) - \frac{1}{2}\sigma(t,x)\sigma(t,x)^T\frac{\partial^2}{\partial x^2}V_i(t,x) = \\
&\min_{u_i}\left\{L_i(t,x,u_i,u_{-i}^*(t,x)) + \frac{\partial}{\partial x}V_i(t,x)^T F(t,x,u_i,u_{-i}^*(t,x))\right\},\\
&V_i(t_f,x) = q_i(t_f,x),\quad i=1,2,\dots,n.
\end{aligned}\tag{6}$$

Introducing the Hamiltonian functions:

$$\begin{aligned}
H^i(t,x,u_i,u_{-i}^*,\frac{\partial}{\partial x}V_i(t,x)) &= L_i(t,x,u_i,u_{-i}^*(t,x))\\
&\quad + \frac{\partial}{\partial x}V_i(t,x)^T F(t,x,u_i,u_{-i}^*(t,x)),\ i=1,2,\dots,n,
\end{aligned}$$

the associated optimal feedback control u_i^* can be obtained by:

$$\frac{\partial H^i}{\partial u_i} = 0 \Rightarrow u_i = u_i^*(t,x) = -\frac{1}{2}R_{ii}^{-1}g_i(t,x)^T\frac{\partial}{\partial x}V_i(t,x),\ i=1,2,\dots,n.\tag{7}$$

Substituting u_i^* from (7) into (6) yields:

$$\begin{aligned}
&\frac{\partial}{\partial t}V_i(t,x) + \frac{1}{2}\sigma(t,x)\sigma(t,x)^T\frac{\partial^2}{\partial x^2}V_i(t,x) + \frac{\partial}{\partial x}V_i(t,x)^T f(t,x)\\
&+x^TQ_ix - \frac{1}{2}\frac{\partial}{\partial x}V_i(t,x)^T\sum_{j=1}^{n}g_j(t,x)R_{jj}^{-1}g_j(t,x)^T\frac{\partial}{\partial x}V_j(t,x)\\
&+\frac{1}{4}\sum_{j=1}^{n}\frac{\partial}{\partial x}V_j(t,x)^Tg_j(t,x)R_{jj}^{-1}R_{ij}R_{jj}^{-1}g_j(t,x)^T\frac{\partial}{\partial x}V_j(t,x) = 0,\\
&V_i(t_f,x) = q_i(t_f,x),\quad i=1,2,\dots,n.
\end{aligned}\tag{8}$$

The n coupled HJ equations do not have exact analytical solutions and are very difficult to solve because of the nonlinear terms in (8).

3. Shifted Chebyshev polynomials and its derivative operational matrix

In this section, we review some basic properties of the shifted Chebyshev polynomials and its derivative operational matrix on the interval $[a,b]$.

Let $T_i(x), i = 0,1,2,\dots$ be the Chebyshev polynomials on interval $[-1,1]$. The shifted Chebyshev polynomials $T_i(\frac{2}{b-a}x - \frac{b+a}{b-a})$ on interval $[a,b]$ be denoted by $T_i^*(x)$ and can be obtained by using the following recurrence formula:

$$\begin{cases}
T_0^*(x) = 1,\\
T_1^*(x) = \frac{2}{b-a}x - \frac{b+a}{b-a},\\
T_{i+1}^*(x) = \left(4\left(\frac{x}{b-a}\right) - 2\left(\frac{b+a}{b-a}\right)\right)T_i^*(x) - T_{i-1}^*(x),\quad i=1,2,\dots.
\end{cases}\tag{9}$$

The derivative of the Chebyshev polynomials satisfies the following property [25]:

$$\frac{d}{dx}T_j(x) = \sum_{k=0,\ j-k\ odd}^{j-1}\frac{2jT_k(x)}{c_k},\quad j\ge 1,\tag{10}$$

where $c_0=2$, and $c_k=1,\ k\ge 1$. From the equality (10), we can find the derivative of shifted Chebyshev polynomials as follows:

$$\frac{d}{dx}T_j^*(x) = \sum_{k=0,\ j-k\ odd}^{j-1}\frac{4j}{c_k(b-a)}T_k^*(x),\quad j=0,1,2,\dots.$$

Now suppose that the $1 \times (N+1)$-shifted Chebyshev vector $\boldsymbol{T}_N^*(x)$ is defined in the following form:

$$\boldsymbol{T}_N^*(x) = [T_0^*(x), T_1^*(x), \ldots, T_N^*(x)], \tag{11}$$

in which the elements $T_i^*(x), i = 0, 1, 2, \ldots, N$ are the shifted Chebyshev polynomials on interval $[a, b]$. Let $\frac{d}{dx}\boldsymbol{T}_N^*(x) = \boldsymbol{T}_N^*(x)D$, in which $D = (D_{ij})_{(N+1)\times(N+1)}$ is the operational matrix of derivative on $[a, b]$ and D_{ij} for $i, j = 0, 1, 2, \ldots, N$ is defined as:

$$D_{ij} = \begin{cases} \frac{4j}{c_i(b-a)}, & \text{if } i+j \text{ is odd, and } j > i, \\ 0, & o.w., \end{cases} \tag{12}$$

where $c_0 = 2$, and $c_i = 1,\ 1 \le i \le N$.

In a spectral method, a smooth continuous function $g(x, y)$ defined on $[a, b] \times [c, d]$ can be approximated by the shifted Chebyshev polynomials as:

$$\begin{aligned} g(x, y) \simeq g_{M,N}(x, y) &= \sum_{i=0}^{M}\sum_{j=0}^{N} g_{ij}T_i^*(x)T_j^*(y) \\ &= (\boldsymbol{T}_M^*(x) \otimes \boldsymbol{T}_N^*(y))G, \end{aligned} \tag{13}$$

where G is the $(M+1) \times (N+1)$-vector as:

$$G = [g_{00}, g_{01}, \ldots, g_{0N}, \ldots, g_{M0}, g_{M1}, \ldots, g_{MN}]^T.$$

From (12) and (13), we can write the partial derivative of $g(x, y)$ as follows:

$$\begin{aligned} \frac{\partial^k}{\partial x^k} g(x, y) &\simeq \left(\frac{\partial^k}{\partial x^k}\boldsymbol{T}_M^*(x) \otimes \boldsymbol{T}_N^*(y)\right) G \\ &= \left(\boldsymbol{T}_M^*(x)\mathbf{D}_{M+1}^k \otimes \boldsymbol{T}_N^*(y)\right) G \\ &= (\boldsymbol{T}_M^*(x) \otimes \boldsymbol{T}_N^*(y))(\mathbf{D}_{M+1}^k \otimes \mathbf{I}_{N+1})G, \end{aligned} \tag{14}$$

where $\mathbf{I}_{N+1}$ is the $(N+1)\times(N+1)$-identity matrix, D_{M+1}^k is the $(M+1)\times(M+1)$-shifted Chebyshev operational matrix of derivative and superscript k that denotes the power of matrix $D_{(M+1)\times(M+1)}$.

Theorem 3.1. *If the function $g(x, y)$ has second order continuous derivatives then*

$$\begin{aligned} |g_{i,0}| &\le \frac{2\gamma_{2,0}}{(i-1)^2}, \quad |g_{i,1}| \le \frac{8\gamma_{2,0}}{\pi(i-1)^2}, \quad i > 1, \\ |g_{0,j}| &\le \frac{2\gamma_{0,2}}{(j-1)^2}, \quad |g_{1,j}| \le \frac{8\gamma_{0,2}}{\pi(j-1)^2}, \quad j > 1, \end{aligned}$$

where $g(x, y) = \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} g_{ij}T_i(x)T_j(y)$, $g_{M,N}(x, y) = \sum_{i=0}^{M}\sum_{j=0}^{N} g_{ij}T_i(x)T_j(y)$, $\gamma_{2,0} \ge \max\{|\frac{\partial^2 g}{\partial x^2}(x, y)| : x, y \in [-1, 1]\}$, and $\gamma_{0,2} \ge \max\{|\frac{\partial^2 g}{\partial y^2}(x, y)| : x, y \in [-1, 1]\}$.

Proof. See [27].

Theorem 3.2. *If the function $g(x, y)$ has second order continuous partial derivatives then*

$$|g(x, y) - g_{M,N}(x, y)| \le \sqrt{6}\left(\frac{20\gamma_{0,2}^2}{(N-1)^2} + \frac{20\gamma_{2,0}^2}{(M-1)^2} + \frac{\pi^2\gamma_{1,1}^2}{6N} + \frac{\pi^2\gamma_{1,1}^2}{6M}\right)^{\frac{1}{2}} \to 0, \tag{15}$$

and $\lim_{M,N\to\infty} g_{M,N}(x, y) = g(x, y)$ uniformly in $[-1, 1]$.

Proof. See [27].

4. Approximate solution of *n*-player nonzero-sum stochastic games

In this section, we apply the CSCM to reduce (8) into a system of nonlinear algebraic equations. Then, we propose the CSCM+PI algorithm to improve the numerical efficiency of the CSCM for solving the coupled HJ equations arising in nonzero-sum stochastic games. Convergence results for the method are discussed in this section.

4.1. Numerical solution of (8) via the CSCM

Assume that the value functions $V_i(t,x)$ for each player $i = 1, 2, \ldots, n$ are continuously differentiable. On a compact set $[t_0, t_f] \times \Sigma$, the value functions $V_i(t,x)$ can be approximated as:

$$V_i(t,x) \simeq \sum_{m=0}^{M}\sum_{n=0}^{N} v_{mn}^{i} T_m^*(t)T_n^*(x) = \boldsymbol{T}_{M,N}(t,x)W_i, \tag{16}$$

where $\Sigma = [x_{min}, x_{max}]$ is state domain,

$$\boldsymbol{T}_{M,N}(t,x) = \boldsymbol{T}_M^*(t) \otimes \boldsymbol{T}_N^*(x),$$

and also

$$W_i = [v_{00}^i, v_{01}^i, \ldots, v_{0N}^i, \ldots, v_{M0}^i, v_{M1}^i, \ldots, v_{MN}^i]^T,\ \ i = 1, 2, \ldots, n, \tag{17}$$

are unknown vectors that should be obtained.

Remark 4.1. In general, the state may reach somewhere outside the domain due to not only the drift term but also the diffusion term, unless some additional conditions are provided. Here, for the computation purpose, we suppose that the state domain Σ is bounded.

Let the solution $V_i(t,x)$ of the coupled HJ equations system (8) be approximated by $V_i(t,x) \simeq \boldsymbol{T}_{M,N}(t,x)W_i$. Then from (14), we have:

$$\begin{aligned}\frac{\partial}{\partial t}V_i(t,x) &\simeq \left(\frac{\partial}{\partial t}\boldsymbol{T}_M^*(t) \otimes \boldsymbol{T}_N^*(x)\right)W_i\\ &= (\boldsymbol{T}_M^*(t) \otimes \boldsymbol{T}_N^*(x))(\mathbf{D}_{M+1} \otimes \mathbf{I}_{N+1})W_i\\ &= \boldsymbol{T}_{M,N}(t,x)\widehat{\mathbf{D}}W_i,\end{aligned} \tag{18}$$

and

$$\begin{aligned}\frac{\partial^k}{\partial x^k}V_i(t,x) &\simeq \left(\boldsymbol{T}_M^*(t) \otimes \frac{\partial^k}{\partial x^k}\boldsymbol{T}_N^*(x)\right)W_i\\ &= (\boldsymbol{T}_M^*(t) \otimes \boldsymbol{T}_N^*(x))(\mathbf{I}_{M+1} \otimes \mathbf{D}_{N+1}^k)W_i\\ &= \boldsymbol{T}_{M,N}(t,x)\widetilde{\mathbf{D}}^kW_i,\end{aligned} \tag{19}$$

where $\widehat{\mathbf{D}} = \mathbf{D}_{M+1} \otimes \mathbf{I}_{N+1}$, $\widetilde{\mathbf{D}}^k = \mathbf{I}_{M+1} \otimes \mathbf{D}_{N+1}^k$. Now, our aim is to approximate the solutions for the time horizon $[t_0, t_f]$ and the state domain $\Sigma = [x_{min}, x_{max}]$. For this purpose, we define

$$t_r = t_0 + l_1\left(1 + \cos\left(\frac{(M-r)\pi}{M}\right)\right),\ \ r = 0, 1, \ldots, M, \tag{20}$$

$$\tau_s = x_{min} + l_2\left(1 + \cos\left(\frac{(N-s)\pi}{N}\right)\right),\ \ s = 0, 1, \ldots, N, \tag{21}$$

which are named as shifted Chebyshev–Gauss–Lobatto nodes, $l_1 = \frac{t_f - t_0}{2}$ and $l_2 = \frac{x_{max} - x_{min}}{2}$. The discretized n coupled HJ equations (8) at shifted Chebyshev–Gauss–Lobatto nodes $\{t_r\}_{r=0}^M$ and $\{\tau_s\}_{s=0}^N$ can be written as:

$$\begin{aligned}&\boldsymbol{T}_{M,N}^{r,s}\widehat{\mathbf{D}}W_i + \frac{\Omega^{r,s}}{2}\boldsymbol{T}_{M,N}^{r,s}\widetilde{\mathbf{D}}^2W_i + W_i^T\widetilde{\mathbf{D}}^T(\boldsymbol{T}_{M,N}^{r,s})^Tf^{r,s}\\ &-\frac{1}{2}W_i^T(\widetilde{\mathbf{D}}^2)^T(\boldsymbol{T}_{M,N}^{r,s})^T\sum_{j=1}^{n} g_j^{r,s}R_{jj}^{-1}(g_j^{r,s})^T\boldsymbol{T}_{M,N}^{r,s}\widetilde{\mathbf{D}}W_j\\ &+\frac{1}{4}\sum_{j=1}^{n} W_j^T\widetilde{\mathbf{D}}^T(\boldsymbol{T}_{M,N}^{r,s})^Tg_j^{r,s}R_{jj}^{-1}R_{ij}R_{jj}^{-1}(g_j^{r,s})^T\boldsymbol{T}_{M,N}^{r,s}\widetilde{\mathbf{D}}W_j\\ &= -\Gamma_i^s,\\ &i = 1, 2, \ldots, n,\ \ r = 0, 1, \ldots, M-1,\ \ s = 0, 1, \ldots, N,\end{aligned} \tag{22}$$

where $\boldsymbol{T}_{M,N}^{r,s} = \boldsymbol{T}_{M,N}(t_r, \tau_s)$, $\Omega^{r,s} = \sigma(t_r,\tau_s)\sigma(t_r,\tau_s)^T$, $f^{r,s} = f(t_r,\tau_s)$, $g_i^{r,s} = g_i(t_r,\tau_s)$ and $\Gamma_i^s = \tau_s^TQ_i\tau_s$.

Also, we suppose that $q_i^{r,s} = q_i(t_r, \tau_s)$ and discretize the boundary conditions as follows:

$$\boldsymbol{T}_{M,N}^{M,s} W_i = q_i^{M,s}, \quad i = 1, 2, \ldots, n, \; s = 0, 1, \ldots, N. \tag{23}$$

By discretizing the coupled HJ equations at collocation points, a system of algebraic equations is formed with only unknowns W_i, $i = 1, 2, \ldots, n$ as the coefficients vector of value functions $V_i(t, x)$. The difficulty in solving the resulted system (22) and (23) is related to nonlinear (quadratic) terms:

$$W_j^T \widetilde{\mathbf{D}}^T (\boldsymbol{T}_{M,N}^{r,s})^T g_j^{r,s} R_{jj}^{-1} R_{ij} R_{jj}^{-1} (g_j^{r,s})^T \boldsymbol{T}_{M,N}^{r,s} \widetilde{\mathbf{D}} W_j, \tag{24}$$

and

$$W_i^T (\widetilde{\mathbf{D}}^2)^T (\boldsymbol{T}_{M,N}^{r,s})^T \sum_{j=1}^{n} g_j^{r,s} R_{jj}^{-1} (g_j^{r,s})^T \boldsymbol{T}_{M,N}^{r,s} \widetilde{\mathbf{D}} W_j. \tag{25}$$

The application of the CSCM thus leads to the nonlinear algebraic system of Eqs. (22) and (23). This system can be reduced to the following form:

$$G_i(W_1, W_2, \ldots, W_n) = 0, \quad i = 1, 2, \ldots, n. \tag{26}$$

One of the disadvantages with applying the CSCM is that the practical success of this method is directly linked to successful algorithms for solving the nonlinear algebraic equations system (26). Generally, we use some iterative methods, such as Newton's method. For instance, in Newton's method, we need an initial guess for the solution. Also, calculating both the Jacobian matrix and its inverse can be quite time consuming.

4.2. Numerical solution of (8) via the CSCM+PI algorithm

We begin in this section to provide a numerical iterative method based on the CSCM and PI algorithm [18] for solving the coupled HJ equations system (8). For this purpose, we introduce the PI algorithm for nonzero-sum stochastic differential game (1) and (4).

First, based on Definition 2.1, we initialize the algorithm with an admissible control set $\{u_1^0, u_2^0, \ldots, u_n^0\}$. For instance, by considering the boundary conditions in (6) and optimal feedback control in (7), selecting a linear approximation of control $u_i(t, x) = -\frac{1}{2} R_{ii}^{-1} g_i(t, x)^T \frac{\partial}{\partial x} q_i(t, x)(i = 1, 2, \ldots, n)$ can be an appropriate initial approximation.

For $k = 1$, with the n-tuple of policies $\{u_1^{k-1}, u_2^{k-1}, \ldots, u_n^{k-1}\}$, we solve the following system of PDEs:

$$\begin{aligned} &\frac{\partial}{\partial t} V_i^k(t, x) + \frac{1}{2} \sigma(t, x) \sigma(t, x)^T \frac{\partial^2}{\partial x^2} V_i^k(t, x) + \\ &\qquad L_i(t, x, u_i^{k-1}, u_{-i}^{k-1}) + \frac{\partial}{\partial x} V_i^k(t, x)^T F(t, x, u_i^{k-1}, u_{-i}^{k-1}) = 0, \\ &V_i^k(t_f, x) = q_i(t_f, x), \quad i = 1, 2, \ldots, n. \end{aligned} \tag{27}$$

For the kth iteration, the n-tuple of control policies can be updated as follows:

$$u_i^k(t, x) = -\frac{1}{2} R_{ii}^{-1} g_i(t, x)^T \frac{\partial}{\partial x} V_i^k(t, x), \; i = 1, 2, \ldots, n. \tag{28}$$

Remark 4.2. Although, by applying the PI algorithm, the nonlinear coupled HJ PDEs system (8) is reduced to a sequence of second order linear uncoupled PDEs system (27), but it remains difficult to find the analytical solution of this PDEs system. So, in practice the PI algorithm cannot be implemented alone and this system has to be solved numerically. On the other hand, based on the reasons stated in Section 4.1, we cannot implement the CSCM alone. In the sequel, we provide the CSCM+PI algorithm to solve the coupled HJ equations (8).

It should be noted that from the Step 5 of Algorithm 1, the application of the CSCM+PI method leads to the linear algebraic system of equations which can be solved by any standard linear solver.

4.3. Convergence analysis

In Section 4.2, we proposed the CSCM+PI algorithm for solving the coupled HJ PDEs. Here, we investigate the convergence of the proposed method in two parts. The first part of the convergence relates to property of the CSCM that is guaranteed according to Theorems 3.1 and 3.2. The second part of the convergence is provided by the property of the PI algorithm. For this purpose, we follow [18–21] and present the convergence of the PI algorithm for the coupled HJ equations arising in nonzero-sum stochastic game.

Algorithm 1 The CSCM+PI algorithm for n-player nonzero-sum stochastic games

Input: $t_0, t_f, x_{min}, x_{max} \in \mathbb{R}, M, N, n \in \mathbb{N}$, small positive number ε , $f, g_i, \Omega, L_i, q_i, R_{ij}, i, j = 1, 2, \ldots n$ and an initial admissible control set $\{u_1^0, u_2^0, \ldots, u_n^0\}$.
Step 1: Compute $\boldsymbol{T}_{M,N}(t,x) = \boldsymbol{T}_M^*(t) \otimes \boldsymbol{T}_N^*(x)$ using (11).
Step 2: Compute t_r and τ_s as shifted Chebyshev–Gauss–Lobatto nodes from (20) and (21), respectively.
Step 3: Compute $\widehat{\mathbf{D}} = \mathbf{D}_{M+1} \otimes \mathbf{I}_{N+1}$ and $\widetilde{\mathbf{D}}^2 = \mathbf{I}_{M+1} \otimes \mathbf{D}_{N+1}^2$ using (12).
Step 4: Set $V_i^0|_{M,N}(t,x) = 0,\ i = 1, 2, \ldots n,\ IN = 1$ and $k = 1$.
Step 5: **for each** $k \geq 1$ **do**
for i **from** 1 **to** n **do**
for r **from** 1 **to** $M-1$ **do**
for s **from** 0 **to** N **do**

$$\begin{aligned}\mathbf{Eq}[i][r][s] \;=\; &\Big(\boldsymbol{T}_{M,N}^{r,s}\widehat{\mathbf{D}} + \frac{\Omega^{r,s}}{2}\boldsymbol{T}_{M,N}^{r,s}\widetilde{\mathbf{D}}^2 \\ &+ \boldsymbol{T}_{M,N}^{r,s}\widetilde{\mathbf{D}}\Big(f^{r,s} + \sum_{j=1}^{n} g_j^{r,s}u_j^{k-1}\Big)\Big)W_i^k + L_i^{r,s}(t,x,u_i^{k-1},u_{-i}^{k-1});\end{aligned}$$

end for
end for
end for
for i **from** 1 **to** n **do**
for s **from** 0 **to** N **do**

$$\mathbf{Eq}[i][M][s] = \boldsymbol{T}_{M,N}^{M,s}W_i^k - q_i^{M,s};$$

end for
end for
Solve the following system of linear algebraic equations for the unknown vectors W_i^k:

$$\mathbf{Eq}[i][r][s] = 0,\ \ i = 1, 2, \ldots, n,\ \ r = 1, 2, \ldots, M,\ \ s = 1, 2, \ldots, N;$$

Compute the following value functions:

$$V_i^k|_{M,N}(t,x) = \boldsymbol{T}_{M,N}(t,x)W_i^k,\ \ i = 1, 2, \ldots, n;$$

Update the n-tuple of strategies as follows:

$$u_i^k|_{M,N}(t,x) = -\frac{1}{2}R_{ii}^{-1}g_i(t,x)^T\boldsymbol{T}_{M,N}(t,x)\widetilde{\mathbf{D}}W_i^k,\ \ i = 1, 2, \ldots, n;$$

if $\max\left\{\left|V_i^k|_{M,N}(t_0,x_0) - V_i^{k-1}|_{M,N}(t_0,x_0)\right|, i = 1, \ldots, n\right\} \leq \varepsilon$ **then**
$\{u_1^k, u_2^k, \ldots, u_n^k\}$ is a feedback Nash equilibrium of game
break
else

$IN = IN + 1;$
$k = k + 1;$

end if
end for
Output: Feedback Nash equilibrium $\{u_1^{IN}, u_2^{IN}, \ldots, u_n^{IN}\}$.

Consider a space $\mathcal{V} \subset \{V(t,x) : [t_0, t_f] \times \sum \to \mathbb{R}, V(t_f, x) = q(t_f, x)\}$ and define mappings $\mathcal{G}_i\, \underbrace{\mathcal{V} \times \mathcal{V} \times \cdots \times \mathcal{V}}_{n} \to \mathcal{V}$ as follows:

$$\begin{aligned}\mathcal{G}_i \;=\; &\frac{\partial}{\partial t}V_i(t,x) + \frac{1}{2}\sigma(t,x)\sigma(t,x)^T\frac{\partial^2}{\partial x^2}V_i(t,x) + \frac{\partial}{\partial x}V_i(t,x)^Tf(t,x) \\ &+ x^TQ_ix - \frac{1}{2}\frac{\partial}{\partial x}V_i(t,x)^T\sum_{j=1}^{n} g_j(t,x)R_{jj}^{-1}g_j(t,x)^T\frac{\partial}{\partial x}V_j(t,x) \\ &+ \frac{1}{4}\sum_{j=1}^{n}\frac{\partial}{\partial x}V_j(t,x)^Tg_j(t,x)R_{jj}^{-1}R_{ij}R_{jj}^{-1}g_j(t,x)^T\frac{\partial}{\partial x}V_j(t,x),\quad i = 1, 2, \ldots, n.\end{aligned} \tag{29}$$

Let $\mathcal{G}'_{iV_i}$ represent the Gâteaux derivative of $\mathcal{G}_i$ taken with respect to V_i. To compute the Gâteaux derivative, we introduce the following lemma.

Lemma 4.1. *Suppose that $\mathcal{G}_i$ be a mapping defined in (29). Then, $\forall V_i \in \mathcal{V}$, the Gâteaux differential of $\mathcal{G}_i$ at V_i is*

$$\mathcal{G}'_{iV_i}Y = \frac{\partial Y}{\partial t} + \frac{\sigma\sigma^T}{2}\frac{\partial^2 Y}{\partial x^2} + (\frac{\partial Y}{\partial x})^Tf - \frac{1}{2}(\frac{\partial Y}{\partial x})^T\sum_{j=1}^{n} g_jR_{jj}^{-1}g_j^T\frac{\partial}{\partial x}V_j. \tag{30}$$

Proof. According to the definition of $\mathcal{G}_i$, $\forall V_i \in \mathcal{V}$, we have

$$\begin{aligned}
&\mathcal{G}_i(V_i + sY) - \mathcal{G}_i(V_i) \\
&\quad = \frac{\partial}{\partial t}(V_i + sY) + \frac{1}{2}\sigma\sigma^T \frac{\partial^2}{\partial x^2}(V_i + sY) \\
&\quad + x^T Q_i x - \frac{1}{4}\frac{\partial}{\partial x}(V_i + sY)^T g_i R_{ii}^{-1} g_i^T \frac{\partial}{\partial x}(V_i + sY) \\
&\quad + \frac{\partial}{\partial x}(V_i + sY)^T f - \frac{1}{2}\frac{\partial}{\partial x}(V_i + sY)^T \sum_{j=1, j\neq i}^{n} g_j R_{jj}^{-1} g_j^T \frac{\partial V_j}{\partial x} \\
&\quad + \frac{1}{4}\sum_{j=1, j\neq i}^{n} \frac{\partial V_j^T}{\partial x} g_j R_{jj}^{-1} R_{ij} R_{jj}^{-1} g_j^T \frac{\partial V_j}{\partial x} \\
&\quad - \Big(\frac{\partial V_i}{\partial t} + \frac{1}{2}\sigma\sigma^T \frac{\partial^2 V_i}{\partial x^2} + x^T Q_i x \\
&\quad - \frac{1}{4}\frac{\partial V_i^T}{\partial x} g_i R_{ii}^{-1} g_i^T \frac{\partial V_i}{\partial x} + \frac{\partial V_i^T}{\partial x} f \\
&\quad - \frac{1}{2}\frac{\partial V_i^T}{\partial x}\sum_{j=1, j\neq i}^{n} g_j R_{jj}^{-1} g_j^T \frac{\partial V_j}{\partial x} + \frac{1}{4}\sum_{j=1, j\neq i}^{n} \frac{\partial V_j^T}{\partial x} g_j R_{jj}^{-1} R_{ij} R_{jj}^{-1} g_j^T \frac{\partial V_j}{\partial x}\Big) \\
&\quad = s\frac{\partial Y}{\partial t} + s\frac{\sigma\sigma^T}{2}\frac{\partial^2 Y}{\partial x^2} + s(\frac{\partial Y}{\partial x})^T f \\
&\quad - \frac{s}{2}(\frac{\partial Y}{\partial x})^T \sum_{j=1}^{n} g_j R_{jj}^{-1} g_j^T \frac{\partial V_j}{\partial x} - \frac{s^2}{4}(\frac{\partial Y}{\partial x})^T g_i R_{ii}^{-1} g_i^T \frac{\partial Y}{\partial x}.
\end{aligned}$$

So, the Gâteaux differential at V_i is

$$\begin{aligned}
\mathcal{G}'_{iV_i} Y &= \lim_{s\to 0} \frac{\mathcal{G}_i(V_i + sY) - \mathcal{G}_i(V_i)}{s} \\
&= \frac{\partial Y}{\partial t} + \frac{\sigma\sigma^T}{2}\frac{\partial^2 Y}{\partial x^2} + (\frac{\partial Y}{\partial x})^T f - \frac{1}{2}(\frac{\partial Y}{\partial x})^T \sum_{j=1}^{n} g_j R_{jj}^{-1} g_j^T \frac{\partial V_j}{\partial x}. \quad \square
\end{aligned}$$

In the following theorem, we will show that the PI algorithm for nonzero-sum stochastic game is also mathematically equivalent to quasi-Newton's iteration, which results in the convergence of the value function V_i^{k+1} to V_i^*, for $i = 1, 2, \ldots, n$, as $k \longrightarrow \infty$.

Theorem 4.3. *Let $\mathcal{G}_i$ and $\mathcal{G}'_{iV_i}$ be operators defined in (29) and (30), respectively. Then the iteration between (27) and (28) is equivalent to the following quasi-Newton's iteration*

$$V_i^{k+1} = V_i^k - (\mathcal{G}'_{iV_i^k})^{-1}\mathcal{G}_i, \ \ k = 0, 1, \ldots. \tag{31}$$

Proof. According to (28) and Lemma 4.1, we have

$$\begin{aligned}
\mathcal{G}'_{iV_i^k} V_i^{k+1} &= \frac{\partial V_i^{k+1}}{\partial t} + \frac{\sigma\sigma^T}{2}\frac{\partial^2 V_i^{k+1}}{\partial x^2} + (\frac{\partial V_i^{k+1}}{\partial x})^T f \\
&- \frac{1}{2}(\frac{\partial V_i^{k+1}}{\partial x})^T \sum_{j=1}^{N} g_j(x) R_{jj}^{-1} g_j(x)^T \frac{\partial}{\partial x} V_j^k \\
&= \frac{\partial V_i^{k+1}}{\partial t} + \frac{\sigma\sigma^T}{2}\frac{\partial^2 V_i^{k+1}}{\partial x^2} + (\frac{\partial V_i^{k+1}}{\partial x})^T (f + \sum_{j=1}^{n} g_j u_j^k).
\end{aligned} \tag{32}$$

and

$$\mathcal{G}'_{iV_i^k} V_i^k = \frac{\partial V_i^k}{\partial t} + \frac{\sigma\sigma^T}{2}\frac{\partial^2 V_i^k}{\partial x^2} + (\frac{\partial V_i^k}{\partial x})^T (f + \sum_{j=1}^{n} g_j u_j^k). \tag{33}$$

From (29) and (28), we have

$$\mathcal{G}_i = \frac{\partial V_i^k}{\partial t} + \frac{\sigma\sigma^T}{2}\frac{\partial^2 V_i^k}{\partial x^2} + L_i(x, u_i^k, u_{-i}^k) + (\frac{\partial V_i^k}{\partial x})^T(f + \sum_{j=1}^{n} g_j u_j^k). \tag{34}$$

Thus,

$$\mathcal{G}'_{iV_i^k} V_i^k - \mathcal{G}_i = -L_i(x, u_i^k, u_{-i}^k). \tag{35}$$

Considering (27), we have $\mathcal{G}'_{iV_i^k} V_i^{k+1} = -L_i(x, u_i^k, u_{-i}^k)$, therefore, $\mathcal{G}'_{iV_i^k} V_i^{k+1} = \mathcal{G}'_{iV_i^k} V_i^k - \mathcal{G}_i$, which results $V_i^{k+1} = V_i^k - (\mathcal{G}'_{iV_i^k})^{-1}\mathcal{G}_i$. □

Now, we assume that $V_i|_{M,N}(t,x)$ and $V_i^k|_{M,N}(t,x)$ for $i = 1, 2, \ldots, n$ are the approximate solutions of value functions obtained by the CSCM and the CSCM+PI, respectively. For fixed k, according to Theorems 3.1 and 3.2, we have:

$$V_i^k|_{M,N}(t,x) \to V_i^k(t,x), \quad as \quad M, N \to \infty, \tag{36}$$

and also, based on Theorem 4.3, we can get:

$$V_i^k(t,x) \to V_i^*(t,x), \quad as \quad k \to \infty. \tag{37}$$

Thus, the convergence of the proposed method can be obtained as follows:

$$V_i^k|_{M,N}(t,x) \to V_i^*(t,x), \quad as \quad M, N, k \to \infty. \tag{38}$$

4.4. Multi-dimensional case

So far we have applied the CSCM+PI only for one-dimensional case of state variables. However, all the expressions given in Section 4 can be extended for the multi-dimensional case of state variables as follows.

Let $\mathbf{x} = (x_1, x_2, \ldots, x_k) \in \Sigma \subset \mathbb{R}^k$ be the k-dimensional state space. Therefore, the shifted Chebyshev vector $\mathbf{T}^*_{N_1,N_2,\ldots,N_k}(\mathbf{x})$ for multi-dimensional case can be expressed as:

$$\mathbf{T}^*_{N_1,N_2,\ldots,N_k}(\mathbf{x}) = \mathbf{T}^*_{N_1}(x_1) \otimes \mathbf{T}^*_{N_2}(x_2) \otimes \cdots \otimes \mathbf{T}^*_{N_k}(x_k),$$

where $\mathbf{T}^*_{N_i}(x_i)$ for $i = 1, 2, \ldots, k$ are the shifted Chebyshev vectors used in Section 3, and defined as:

$$\mathbf{T}^*_{N_i}(x_i) = \left[T_0^*(x_i), T_1^*(x_i), \ldots, T_{N_i}^*(x_i)\right], \quad i = 1, 2, \ldots, k,$$

where x_i indicates the ith entry of the state vector $\mathbf{x}$. The value functions $V_i(t, \mathbf{x})$, $i = 1, 2, \ldots, n$ can be approximated as:

$$V_i(t,\mathbf{x}) \simeq \mathbf{T}_{M,N_1,N_2,\ldots,N_k}(t,\mathbf{x}) W_i, \quad i = 1, 2, \ldots, n,$$

in which

$$\mathbf{T}_{M,N_1,N_2,\ldots,N_k}(t,\mathbf{x}) = \mathbf{T}^*_M(t) \otimes \mathbf{T}^*_{N_1,N_2,\ldots,N_k}(\mathbf{x}),$$

and

$$W_i = \left[v^i_{000\ldots00}, v^i_{000\ldots01}, \ldots, v^i_{000\ldots0N_k}, \ldots, v^i_{MN_1N_2\ldots N_{k-1}0}\right.$$
$$\left. v^i_{MN_1N_2\ldots N_{k-1}1}, \ldots, v^i_{MN_1N_2\ldots N_{k-1}N_k}\right]^T. \tag{39}$$

Also, we have:

$$\begin{aligned}\frac{\partial}{\partial t} V_i(t,\mathbf{x}) &\simeq \left(\frac{\partial}{\partial t}\boldsymbol{T}^*_M(t) \otimes \boldsymbol{T}^*_{N_1,N_2,\ldots,N_k}(x)\right) W_i \\ &= (\boldsymbol{T}^*_M(t) \otimes \boldsymbol{T}^*_{N_1,N_2,\ldots,N_k}(\mathbf{x})) \\ &\quad (\mathbf{D}_{M+1} \otimes \mathbf{I}_{N_1+1} \otimes \mathbf{I}_{N_2+1} \otimes \cdots \otimes \mathbf{I}_{N_k+1}) W_i \\ &= \boldsymbol{T}_{M,N_1,N_2,\ldots,N_k}(t,\mathbf{x}) \widehat{\mathbf{D}} W_i,\end{aligned}$$

and

$$\begin{aligned}
\frac{\partial^s}{\partial x_j^s} V_i(t,\mathbf{x}) &\simeq \left(\boldsymbol{T}_M^*(t) \otimes \frac{\partial^s}{\partial x_j^s} \boldsymbol{T}_{N_1,N_2,\ldots,N_k}^*(\mathbf{x}) \right) W_i \\
&= (\boldsymbol{T}_M^*(t) \otimes \boldsymbol{T}_{N_1}^*(x_1) \otimes \boldsymbol{T}_{N_2}^*(x_2) \otimes \cdots \otimes \boldsymbol{T}_{N_k}^*(x_k)) \\
&\quad (\mathbf{I}_{M+1} \otimes \mathbf{I}_{N_1+1} \otimes \cdots \otimes \mathbf{I}_{N_{j-1}+1} \otimes \mathbf{D}_{N_j+1}^s \otimes \mathbf{I}_{N_{j+1}+1} \otimes \cdots \otimes \mathbf{I}_{N_k+1}) W_i \\
&= \boldsymbol{T}_{M,N_1,N_2,\ldots,N_k}(t,\mathbf{x}) \widetilde{\mathbf{D}}_j^s W_i, \qquad (40)
\end{aligned}$$

where

$$\begin{aligned}
\widehat{\mathbf{D}} &= \mathbf{D}_{M+1} \otimes \mathbf{I}_{N_1+1} \otimes \mathbf{I}_{N_2+1} \otimes \cdots \otimes \mathbf{I}_{N_k+1}, \\
\widetilde{\mathbf{D}}_j^s &= \mathbf{I}_{M+1} \otimes \mathbf{I}_{N_1+1} \otimes \cdots \otimes \mathbf{I}_{N_{j-1}+1} \otimes \mathbf{D}_{N_j+1}^s \otimes \mathbf{I}_{N_{j+1}+1} \otimes \cdots \otimes \mathbf{I}_{N_k+1}.
\end{aligned}$$

5. Illustrative examples

For the purpose of illustration, three examples are presented in this section. Example 5.1 is a stochastic control problem, which is a one-person version of an n-person stochastic differential game, and can be solved analytically. This allows us to evaluate the error between the control obtained from the CSCM+PI and that from the analytical solution. Also, we can verify the obtained control is indeed optimal. For an n-person stochastic linear-quadratic differential game in Example 5.2, we can solve these problems via guessing a solution of form:

$$V_i(t,x) = \frac{1}{2} x^T Z_i(t) x + x^T \zeta_i(t) + \xi_i(t), \quad i = 1, 2, \ldots, n, \qquad (41)$$

where $Z_i(t)$, $\zeta_i(t)$ and $\xi_i(t)$ for $i = 1, 2, \ldots, n$ satisfy the n coupled matrix Riccati differential equations. In general, solving this system of equations can be difficult. In Example 5.3, we consider an application of stochastic differential games in competitive advertising. All the computations associated with the proposed method have been performed by using Maple 17 with 32 digits precision.

Example 5.1 (*[31]*). Consider the following dynamic optimization problem with a single decision maker:

$$\min_u \left\{ E\left[\int_0^1 \left(\frac{1}{2} u^2(t) + \frac{1}{2} x(t) u(t) + \frac{5}{8} x^2(t) \right) dt \right] \right\}, \qquad (42)$$

subject to

$$dx(t) = \left(\frac{1}{2} x(t) + u(t) \right) dt + dw(t), \quad x(0) = 1. \qquad (43)$$

Here, the HJB equation is as follows:

$$\frac{\partial}{\partial t} V(t,x) + \frac{1}{2} x^2 - \frac{1}{2} \left(\frac{\partial}{\partial x} V(t,x) \right)^2 + \frac{1}{2} \frac{\partial^2}{\partial x^2} V(t,x) = 0, \quad V(1,x) = 0. \qquad (44)$$

The exact solutions of value function $V^*(t,x)$ and control variable $u^*(t,x)$ are as follows:

$$\begin{aligned}
V^*(t,x) &= \frac{1-\exp(2t-2)}{2+2\exp(2t-2)} x^2 + \frac{1}{2} \ln\left(\frac{1+\exp(2t-2)}{2\exp(t-1)} \right), \qquad (45) \\
u^*(t,x) &= \frac{-3+\exp(2t-2)}{2+2\exp(2t-2)} x. \qquad (46)
\end{aligned}$$

The exact solution for the performance index is $J^* = V(t_0, x_0) = 0.5976874930$. The computational results of J using the the CSCM+PI are listed in Table 1 for different values of M, N and k. Let us define

$$\begin{aligned}
e_{M,N} &= \|V|_{M,N} - V^*\|_\infty = \max_{(t,x)\in\Lambda} \left| V|_{M,N}(t,x) - V^*(t,x) \right|, \\
e_{M,N}^k &= \|V^k|_{M,N} - V^*\|_\infty = \max_{(t,x)\in\Lambda} \left| V^k|_{M,N}(t,x) - V^*(t,x) \right|,
\end{aligned}$$

where $(t,x) \in [0,1] \times [-1,1]$. The comparisons between $e_{M,N}$ and $e_{M,N}^k$ are given in Table 2. In Table 2, it can be seen that the CSCM+PI method improves the computation time for solving the HJB equation.

Example 5.2 (*[8]*). Consider the stochastic two-player linear-quadratic differential game

$$\min_{u_1} J_1 = \left\{ E\left[\int_0^1 \left(x^2(t) + u_1^2(t) \right) dt \right] \right\},$$

Table 1
The optimal cost functional J obtained using the CSCM+PI as compared with exact solutions for Example 5.1.

M,N	k	J	$\mid J-J^*\mid$
2,2	2	0.6237244897	$2.6E-1$
4,2	2	0.5976104518	$7.7E-4$
6,2	2	0.5976851596	$7.7E-5$
8,3	3	0.5976876635	$1.7E-7$
10,3	3	0.5976874893	$3.6E-9$
12,3	3	0.5976874932	$2.7E-10$

Table 2
The comparisons between $e_{M,N}$ and $e^k_{M,N}$ for the error of approximate solutions of value function obtained by the CSCM and the CSCM+PI for Example 5.1.

M,N	k	$e^k_{M,N}$	CPU time(s)	$e_{M,N}$	CPU time(s)
2,2	2	$2.6E-1$	1.266	$2.5E-2$	1.482
4,2	2	$7.7E-4$	1.296	$1.6E-4$	1.669
6,2	2	$7.7E-5$	1.453	$4.0E-6$	2.075
8,3	3	$1.7E-7$	1.295	Fail	–
10,3	3	$3.6E-9$	2.402	Fail	–
12,3	3	$2.7E-10$	3.853	Fail	–

and

$$\min_{u_2} J_2 = \left\{ E\left[\int_0^1 \left(2x^2(t)+u_2^2(t)\right)dt + x^2(1)\right]\right\}, \tag{47}$$

subject to

$$dx(t) = (x(t)+u_1(t)+u_2(t))dt + dw(t), \quad x(0)=1. \tag{48}$$

By introducing the Hamiltonian functions

$$H^1(t,x,u_1,u_2,\frac{\partial}{\partial x}V_1(t,x)) = x^2(t)+u_1^2(t)+\frac{\partial}{\partial x}V_1(t,x)(x(t)+u_1(t)+u_2(t)),$$
$$H^2(t,x,u_1,u_2,\frac{\partial}{\partial x}V_2(t,x)) = 2x^2(t)+u_2^2(t)+\frac{\partial}{\partial x}V_2(t,x)(x(t)+u_1(t)+u_2(t)),$$

we have

$$\begin{cases} \frac{\partial}{\partial t}V_1(t,x)+\frac{1}{2}\frac{\partial^2}{\partial x^2}V_1(t,x)+\min_{u_1}\{H^1(t,x,u_1,u_2,\frac{\partial}{\partial x}V_1(t,x))\}=0,\\ \frac{\partial}{\partial t}V_2(t,x)+\frac{1}{2}\frac{\partial^2}{\partial x^2}V_2(t,x)+\min_{u_2}\{H^2(t,x,u_1,u_2,\frac{\partial}{\partial x}V_2(t,x))\}=0,\\ V_1(1,x)=0, \quad V_2(1,x)=x^2. \end{cases} \tag{49}$$

Minimization of each $H^i(t,x,u_1,u_2,\frac{\partial}{\partial x}V_i(t,x)),\ i=1,2$, leads to the expression for control law $u_i,\ i=1,2$, as follows:

$$u_1^*(t,x) = -\frac{1}{2}\frac{\partial}{\partial x}V_1(t,x), \quad u_2^*(t,x) = -\frac{1}{2}\frac{\partial}{\partial x}V_2(t,x). \tag{50}$$

Substituting (50) into (49), one obtains the two coupled HJ equations:

$$\begin{aligned} &\frac{\partial}{\partial t}V_1(t,x)+\frac{1}{2}\frac{\partial^2}{\partial x^2}V_1(t,x)+x^2-\frac{1}{4}(\frac{\partial}{\partial x}V_1(t,x))^2\\ &\qquad +x\frac{\partial}{\partial x}V_1(t,x)-\frac{1}{2}\frac{\partial}{\partial x}V_1(t,x)\frac{\partial}{\partial x}V_2(t,x)=0,\\ &\frac{\partial}{\partial t}V_2(t,x)+\frac{1}{2}\frac{\partial^2}{\partial x^2}V_2(t,x)+2x^2-\frac{1}{4}(\frac{\partial}{\partial x}V_2(t,x))^2\\ &\qquad +x\frac{\partial}{\partial x}V_2(t,x)-\frac{1}{2}\frac{\partial}{\partial x}V_1(t,x)\frac{\partial}{\partial x}V_2(t,x)=0,\\ &V_1(1,x)=0, \qquad V_2(1,x)=x^2. \end{aligned} \tag{51}$$

The CSCM and the CSCM+PI methods are applied to solve Eqs. (51). The computational results of J_1 and J_2 using the present method for different values of M, N, and k are listed in Table 3. It can be seen that by increasing the value of M, N and k, we can obtain a satisfactory convergence. Notice that the CSCM+PI for $x \in [-5,5]$ works, while the CSCM for x

Table 3
The optimal cost functionals J_1 and J_2 obtained using the CSCM+PI at different values of M, N and k for Example 5.2.

M, N	k	J_1	J_2
10,4	5	0.6587542261	3.9519211051
14,4	5	0.6587256967	3.9511521297
16,4	5	0.6587213865	3.9511447964
14,6	7	0.6587212006	3.9511443521
16,6	7	0.6587212000	3.9511443516
18,6	7	0.6587211608	3.9511442815

Table 4
The comparisons between CPU time(s) obtained by the CSCM and the CSCM+PI methods on $(t, x) \in [0, 1] \times [0, 1]$ for Example 5.2.

M, N	k	CPU time(s) for the CSCM+PI	CPU time(s) for CSCM
8,2	5	1.622	9.828
12,2	5	4.539	28.15
14,4	7	30.15	311.4
12,3	7	46.44	–
14,3	7	66.28	–

in this range fails. Therefore, in Table 4, the comparisons between CPU time(s) obtained by the CSCM and the CSCM+PI are given on $(t, x) \in [0, 1] \times [0, 1]$.

Example 5.3 ([10,13]). Consider the stochastic dynamic advertising game. We suppose that the expected profit of firm 1 and firm 2 is respectively:

$$\max_{u_1} \left\{ E\left[\int_0^{t_f} \left(q_1 x(t) - \frac{c_1}{2} u_1^2(t) \right) \exp(-rt)dt + \exp(-rt_f) q_{1t_f} x(t_f) \right] \right\},$$

and

$$\max_{u_2} \left\{ E\left[\int_0^{t_f} \left(q_2 (1 - x(t)) - \frac{c_2}{2} u_2^2(t) \right) \exp(-rt)dt + \exp(-rt_f) q_{2t_f} (1 - x(t_f)) \right] \right\},$$

for $i = 1, 2$, q_i, c_i, q_{it_f} and r are real positive constants. $x(t)$ and $(1-x(t))$ are the market shares of firm 1 and firm 2 at time t, respectively, and advertising rate $u_i(t)$ is the control variable of firm i for $i = 1, 2$. The dynamic of game is governed by:

$$dx(t) = (u_1(t)\sqrt{1 - x(t)} - u_2(t)\sqrt{x(t)})dt + \sigma x(t)dw(t), \;\; x(0) = 1, \tag{52}$$

where σ is a real positive constant.

By introducing the Hamiltonian functions

$$\begin{aligned} H^1(t, x, u_1, u_2, \frac{\partial}{\partial x} V_1(t, x)) &= \frac{\partial}{\partial x} V_1(t, x)(u_1(t)\sqrt{1 - x(t)} - u_2(t)\sqrt{x(t)}) \\ &\quad + (q_1 x(t) - \frac{c_1}{2} u_1^2(t)) \exp(-rt), \\ H^2(t, x, u_1, u_2, \frac{\partial}{\partial x} V_2(t, x)) &= \frac{\partial}{\partial x} V_2(t, x)(u_1(t)\sqrt{1 - x(t)} - u_2(t)\sqrt{x(t)}) \\ &\quad + (q_2(1 - x(t)) - \frac{c_2}{2} u_2^2(t)) \exp(-rt), \end{aligned}$$

we have

$$\begin{cases} \frac{\partial}{\partial t} V_1(t, x) + \frac{1}{2}\sigma^2 x^2 \frac{\partial^2}{\partial x^2} V_1(t, x) + \min_{u_1}\{H^1(t, x, u_1, u_2, \frac{\partial}{\partial x} V_1(t, x))\} = 0, \\ \frac{\partial}{\partial t} V_2(t, x) + \frac{1}{2}\sigma^2 x^2 \frac{\partial^2}{\partial x^2} V_2(t, x) + \min_{u_2}\{H^2(t, x, u_1, u_2, \frac{\partial}{\partial x} V_2(t, x))\} = 0, \\ V_1(t_f, x) = q_{1t_f} \exp(-rt_f)x, \quad V_2(t_f, x) = q_{2t_f} \exp(-rt_f)(1 - x). \end{cases} \tag{53}$$

Minimization of each $H^i(t, x, u_1, u_2, \frac{\partial}{\partial x} V_i(t, x))$, $i = 1, 2$, leads to the expression for control law u_i, $i = 1, 2$, as follows:

$$u_1^*(t, x) = \frac{\partial}{\partial x} V_1(t, x) \frac{\sqrt{1 - x}}{\exp(-rt)c_1}, \quad u_2^*(t, x) = -\frac{\partial}{\partial x} V_2(t, x) \frac{\sqrt{x}}{\exp(-rt)c_2}. \tag{54}$$

Table 5
The optimal cost functionals J_1 and J_2 obtained using the CSCM+PI at different values of M, N and k for Example 5.3.

M, N	k	J_1	J_2
8,2	5	0.7163094833	0.1418452583
9,3	5	0.7163060613	0.1418469693
10,2	7	0.7163068541	0.1418465729
12,3	7	0.7163067119	0.1418466440
16,4	7	0.7163067036	0.1418466481

Table 6
The comparisons between CPU time(s) obtained by the CSCM and the CSCM+PI on $(t, x) \in [0, 1] \times [0, 1]$ for Example 5.3.

M, N	k	CPU time(s) for CSCM with PI	CPU time(s) for CSCM
8,2	5	4.196	21.84
9,3	5	8.798	29.53
10,2	7	9.130	13.72
12,3	7	21.79	67.14
16,4	7	81.97	–

Substituting (54) into (53), one obtains the coupled HJ equations

$$\begin{aligned}
&\frac{\partial}{\partial t}V_1(t,x) + \frac{1}{2}\sigma^2x^2\frac{\partial^2}{\partial x^2}V_1(t,x) + q_1\exp(-rt)x - \frac{1}{2c_1}\exp(t)(x-1)(\frac{\partial}{\partial x}V_1(t,x))^2\\
&\qquad + c_2^{-1}x\exp(rt)\frac{\partial}{\partial x}V_1(t,x)\frac{\partial}{\partial x}V_2(t,x) = 0,\\
&\frac{\partial}{\partial t}V_2(t,x) + \frac{1}{2}\sigma^2x^2\frac{\partial^2}{\partial x^2}V_2(t,x) + q_2\exp(-rt)(1-x) + \frac{1}{2c_2}\exp(rt)(x)V_2(t,x)^2\\
&\qquad + c_1^{-1}\exp(rt)(1-x)\frac{\partial}{\partial x}V_1(t,x)\frac{\partial}{\partial x}V_2(t,x) = 0,\\
&V_1(t_f,x) = q_{1t_f}\exp(-rt_f)x, \quad V_2(t_f,x) = q_{2t_f}\exp(-rt_f)(1-x).
\end{aligned} \tag{55}$$

The CSCM and the CSCM+PI methods are applied to solve Eqs. (55). In Table 5, the computational results of J_1 and J_2 using the present method for different values of M, N and k are listed. The comparisons between CPU time(s) obtained by the CSCM and the CSCM+PI on $(t, x) \in [0, 1] \times [0, 1]$ are given in Table 6.

6. Conclusions

In this article, we have presented a combined iterative method for solving the second order coupled HJ equations arising in nonzero-sum stochastic differential games. In our approach, the CSCM+PI algorithm are applied to approximate the value functions. Convergence results for the method have also been provided. In several examples, the CSCM+PI algorithm was applied to obtain the feedback Nash equilibrium solutions. The accuracy and CPU time(s) of the obtained solutions by the CSCM+PI algorithm were compared with the obtained solutions by the CSCM. The results showed that the CSCM+PI algorithm is more effective than the CSCM in which the system of nonlinear algebraic equations obtained by the CSCM is not easily solvable. Although, there exist other computational methods for solving the coupled HJ equations, but to the best of our knowledge, none of those methods have not been applied to the coupled HJ equations arising in differential games with finite-horizon and nonlinear boundary conditions.

References

[1] R. Bafico, On the definition of stochastic differential games an the existence of value ond of saddle points, Ann. Math. Pura Appl. 96 (1972) 41–67.
[2] E. Roxin, C.P. Tsokos, On the definition of a stochastic differential game, Math. Syst. 4 (1) (1970) 60–64.
[3] R.J. Elliott, Stochastic differential games and alternate play, in: Proceedings of the International Symposium on Control Theory at I.N.R.I.A., in: Lecture Notes in Econ. Math. Syst., vol. 107, Springer-Verlag, 1974, pp. 97–106.
[4] R.J. Elliott, Introduction to differential games II. Stochastic Games and Parabolic Equations, Theory and Appl. of Diff Games, Dordrecht, Holland, 1975, pp. 34–43.
[5] R.J. Elliott, The existence of value in stochastic differential games, SIAM J. Control Optim. 14 (1976) 85–94.
[6] A. Bensoussan, J.L. Lions, Stochastic differential games with stopping times, Diff. Games and Control Theory II, in: Proceedings of second Kingston conference, 1976, pp. 377–399.
[7] A. Bensoussan, A. Friedman, Nonzero-sum stochastic differential games with stopping times and free boundary problems, Trans. Amer. Math. Soc. 231 (1977) 275–327.

[8] T. Basar, G.J. Olsder, Dynamic Noncooperative Game Theory, Reprint of the second ed., SIAM, Philadelphia, 1999.
[9] J.C. Engwerda, LQ Dynamic Optimization and Differential Games, John Wiley and Sons, 2005.
[10] D.W.K. Yeung, L.A. Petrosyan, Cooperative Stochastic Differential Games, Springer, 2005.
[11] Z. Nikooeinejad, A. Delavarkhalafi, M. Heydari, A numerical solution of open-loop Nash equilibrium in nonlinear differential games based on Chebyshev pseudospectral method, J. Comput. Appl. Math. 300 (2016) 369–384.
[12] H.J. Kushner, P.G. Dupuis, Numerical Methods for Stochastic Control Problems in Continuous Time, second ed., Springer-Verlag, 2001.
[13] K.M. Ramachandran, C.P. Tsokos, Stochastic Differential Games: Theory and Applications, Atlantis Press, USA, 2012.
[14] H.J. Kushner, S.G. Chamberlain, Finite state stochastic games: existence theorems and computational procedures, IEEE Trans. Automat. Control. 14 (1969) 248–255.
[15] H.J. Kushner, Numerical approximations for stochastic differential games: the ergodic case, SIAM J. Control Optim. 42 (2004) 1911–1933.
[16] H.J. Kushner, Numerical methods for stochastic differential games, SIAM J. Control Optim. 41 (2002) 457–486.
[17] M. Akian, S. Detournay, Multigrid methods for two-player zero-sum stochastic games, Numer. Linear Algebra Appl. 19 (2012) 313–342.
[18] K.G. Vamvoudakis, F.L. Lewis, Multi-player non-zero-sum games: online adaptive learning solution of coupled Hamilton–Jacobi equations, Automatica 47 (8) (2011) 1556–1569.
[19] D. Vrabie, F.L. Lewis, Integral reinforcement learning for online computation of feedback Nash strategies of nonzero-sum differential games, in: Proceedings of the IEEE conference on Decis. Control., 2010, pp. 3066–3071.
[20] D. Liu, Q. Wei, D. Wang, X. Yang, H. Li, Adaptive Dynamic Programming with Applications in Optimal Control, in: Adv. Ind. Control., Springer, 2017.
[21] H. Li, D. Liu, D. Wang, Adaptive dynamic programming for solving nonzero-sum differential games, in: 3rd IFAC Int. Conf. Intell. Control and Autom. Sci., 2013, pp. 587–591.
[22] B. Fornberg, A Practical Guide to Pseudospectral Methods, Cambridge Monographs, 1995.
[23] C. Canuto, M.Y. Hussaini, A. Quarteroni, T.A. Zang, Spectral Methods in Fluid Dynamics, Springer, NewYork, 1988.
[24] J. Boyd, Chebyshev and Fourier Spectral Methods, second ed., Dover, 2001.
[25] J.C. Mason, D.C. Handscomb, Chebyshev Polynomials, CRC Press LLC, 2003.
[26] Z. Avazzadeh, M. Heydari, Chebyshev polynomials for solving two dimensional linear and nonlinear integral equations of the second kind, J. Comput. Appl. Math. 31 (2012) 127–142.
[27] E. Scheiber, On the Chebyshev approximation of a function with two variables, 2015, arXiv:1504.04693.
[28] E. Tohidi, Application of Chebyshev collocation method for solving two classes of non-classical parabolic PDEs, Ain Shams Engrg. J. (2015) 373–379.
[29] F.R. Chang, Stochastic Optimization in Continuous Time, Cambridge University Press, 2004.
[30] P. Mannucci, Nonzero-sum stochastic differential games with discontinuous feedback, SIAM J. Control Optim. 43 (4) (2004) 1222–1233.
[31] B. Kafash, A. Delavarkhalafi, S.M. Karbassi, A computational method for stochastic optimal control problems in financial mathematics, Asian J. Control. 18 (4) (2016) 1501–1512.