مجله کنترل
ISSN (print) 2008-8345
ISSN (online) 2538-3752
جلد ۱۲، شماره ۲، تابستان ۱۳۹۷، صفحه ۲۵-۱۳

حل زیربهینه بازی های گرافی دیفرانسیلی غیر خطی با استفاده از برنامه ریزی پویای تقریبی تک-شبکه

مجید مازوچی[۱]، محمد باقر نقیبی سیستانی[۲]، سید کمال حسینی ثانی[۳]

[۱] دانشجوی دکتری مهندسی برق، گروه مهندسی برق، دانشگاه فردوسی مشهد، Mazouchi.majid@mail.um.ac.ir

[۲] دانشیار دانشکده مهندسی، گروه مهندسی برق، دانشگاه فردوسی مشهد، mb-naghibi@um.ac.ir

[۳] دانشیار دانشکده مهندسی، گروه مهندسی برق، دانشگاه فردوسی مشهد، K.hosseini@um.ac.ir

دریافت: ۱۳۹۵/۳/۲۷ ویرایش اول: ۱۳۹۶/۴/۱۰ پذیرش: ۱۳۹۶/۹/۱۹

چکیده: در این مقاله یک الگوریتم یادگیری برخط برمبنای برنامه ریزی پویای تقریبی تک-شبکه برای حل تقریبی بازی های گرافی دیفرانسیلی زمان پیوسته غیرخطی با تابع هزینه زمان نامحدود و دینامیک معین پیشنهاد شده است. در بازی های گرافی دیفرانسیلی، هدف عامل ها ردیابی حالت رهبر به صورت بهینه می باشد، به طوری که دینامیک خطا و اندیس عملکرد هر عامل بستگی به توپولوژی گراف تعاملی بازی دارد. در الگوریتم پیشنهادی، هر عامل تنها از یک شبکه عصبی نقاد برای تقریب ارزش و سیاست کنترلی بهینه خود استفاده می کند و از قوانین تنظیم وزن پیشنهاد شده برای به روزرسانی برخط وزن های شبکه عصبی نقاد خود بهره می جوید. در این مقاله، با معرفی سوئیچ های پایدار ساز محلی در قوانین تنظیم وزن های شبکه عصبی که پایداری سیستم حلقه بسته و همگرایی به سیاست های تعادل نش را تضمین می کنند، دیگر نیازی به مجموعه سیاست های کنترلی پایدار ساز اولیه وجود ندارد. بعلاوه در این مقاله از تئوری لیاپانوف برای اثبات پایداری سیستم حلقه بسته استفاده می شود. در پایان، مثال شبیه سازی، موثر بودن الگوریتم پیشنهادی را نشان می دهد.

کلمات کلیدی: برنامه ریزی پویای تقریبی، شبکه های عصبی، کنترل بهینه، یادگیری تقویتی.

Suboptimal Solution of Nonlinear Graphical Games Using Single Network Approximate Dynamic Programming

Majid Mazouchi, Mohammad B. Naghibi Sistani, Seyed Kamal Hosseini Sani

Abstract: In this paper, an online learning algorithm based on approximate dynamic programming is proposed to approximately solve the nonlinear continuous time differential graphical games with infinite horizon cost functions and known dynamics. In the proposed algorithm, every agent employs a critic neural network (NN) to approximate its optimal value and control policy and utilizes the proposed weight tuning laws to learn its critic NN optimal weights in an online fashion. Critic NN weight tuning laws containing a stabilizer switch guarantees the closed-loop system stability and the control policies convergence to the Nash equilibrium. In this algorithm, there is no requirement for any set of initial stabilizing control policies anymore. Furthermore, Lyapunov theory is employed to show uniform ultimate boundedness of the closed-loop system. Finally, a simulation example is presented to illustrate the efficiency of the proposed algorithm.

Keywords: Approximate Dynamic Programming; Neural Networks; Optimal Control; Reinforcement learning.

۱- مقدمه

تحقیق بر روی کنترل توزیع شده سیستم های چندعاملی در [۵-۱] مورد مطالعه قرار گرفته است. این زمینه ی در حال رشد، در زمینه های متفاوتی از سیستم های مهندسی، مانند آرایش گروهی از ربات های متحرک[۱] [۶]، هواپیماهای بدون سرنشین[۲] [۷]، کنترل آرایش وسایل نقلیه[۳] [۸]، تیم های خودمختار تحت شبکه[۴] [۹]، کنترل سیستم های الکترونیک قدرت [۱۰] و همزمان سازی فرآیندهای دینامیکی کاربرد دارد. کنترل توزیع شده در مقایسه با کنترل متمرکز دارای مزایای بسیاری از جمله پیچیدگی محاسباتی پایین تر، عدم نیاز به یک مرکز تصمیم گیری بصورت متمرکز و قابلیت اطمینان بالاتر بوده که باعث مورد توجه قرار گرفتن این زمینه شده است.

مسایل کنترل توزیع شده به دو گروه عمده، اجماع بدون رهبر[۵] (تنظیم توزیع شده) و همزمان سازی به رهبر[۶] (ردیابی توزیع شده)، تقسیم می شوند. در اجماع بدون رهبر [۱۴-۱۱] همه عامل ها به یک مقدار مشترک کنترل نشده که وابسته به حالت های اولیه آنها در شبکه ارتباطی می باشد، همگرا می شوند. از طرف دیگر، در مسئله همزمان سازی به رهبر که به آن اجماع پیرو-رهبر[۷] [۱۵] نیز گفته می شود، همه عامل ها به رهبر یا عامل کنترلی که مسیر مرجع مطلوب را تولید می کند، همزمان سازی می شوند [۲۱-۱۶].

تئوری بازی یک چارچوب حل مناسب برای مدل سازی و فرمول بندی مسایل کنترلی و تصمیم گیری چندنفره را فراهم می آورد، که در آن سیاست هر بازیکن بستگی به عملکرد خود بازیکن و دیگر عامل های بازی دارد [۲۲]. بازی های دیفرانسیلی شاخه ای از تئوری بازی هستند که به مسئله کنترل سیستم های چند عامله با تعاملات دینامیکی می پردازند. کلاس جدیدی از بازی های دیفرانسیلی با نام بازی های گرافی دیفرانسیلی در [۲۳] معرفی شده اند که شاخص عملکرد و دینامیک خطای هر بازیکن وابسته به توپولوژی گراف تعاملی بازی است. در بازی های گرافی دیفرانسیلی، بطور کلی بازیکنان به دنبال یافتن مجموعه ای از سیاست های کنترلی قابل قبولی هستند، که علاوه بر تضمین پایداری سیستم و همزمان سازی، با حداقل سازی توابع هزینه، حل نقطه تعادل نش نیز حاصل گردد. در این دسته از مسایل به منظور یافتن نقطه تعادل نش، باید به حل معادلات هامیلتون- جاکوبی کوپل شده[۸] پرداخت. حل این معادلات که کاملا وابسته به توپولوژی گراف تعاملی می باشد، بسیار دشوار بوده و در اکثر موارد حل تحلیلی آنها غیر ممکن است. بنابراین به منظور حل تقریبی برخط معادلات هامیلتون-جاکوبی کوپل شده، از روش های یادگیری تقویتی[۹] [۲۴،۲۵] که روش های غیر تحلیلی و عددی هستند، استفاده می شود. برنامه ریزی پویای تقریبی[۱۰] یک روش یادگیری تقویتی پیشرو در زمان[۱۱] می باشد، که می تواند برای یافتن سیاست های بهینه تقریبی برخط مورد استفاده قرار گیرد[۲۶].

مفاهیم برنامه ریزی پویای تقریبی و بازی های گرافی دیفرانسیلی در [۲۷،۲۳-۳۰]، به منظور پیدا نمودن حل بهینه کنترل ردیابی توزیع شده سیستم های خطی زمان پیوسته بصورت بر خط مورد استفاده قرار گرفته اند. در [۲۳]، یک الگوریتم تکرار سیاست همکارانه برخط برای حل بازی های گرافی دیفرانسیلی توسعه داده شد که از ساختار عملگر-نقاد[۱۲] [۳۱] با دو شبکه عصبی پیوند تابعی[۱۳] [۳۲،۳۳] استفاده کرده است. یک الگوریتم تکرار سیاست بر اساس تکنیک انتگرال یادگیری تقویتی [۳۴] به منظور یادگیری حل نش بازی های گرافی دیفرانسیلی خطی بصورت بر خط در [۲۹] پیشنهاد شده است. در [۳۰]، یک الگوریتم تکرار سیاست خطی برای حل بازی های گرافی دیفرانسیلی خطی بصورت بر خط توسعه داده شد و یک الگوریتم تکرار سیاست همکارانه نیز در [۲۷] به منظور حل بازی های گرافی دیفرانسیلی خطی با بازیکنانی با دینامیک های متفاوت پیشنهاد داده شده است. در [۲۸]، یک الگوریتم تکرار سیاست برخط برای پیدا نمودن حل معادلات همیلتون-جاکوبی-ایساک کوپل شده[۱۴] در بازی های گرافی دیفرانسیلی مجموع-صفر خطی که بازیکنان در آن تحت تاثیر اغتشاش هستند، پیشنهاد شده است. محققان در مطالعه [۳۵]، یک الگوریتم برنامه ریزی پویای تقریبی به منظور حل بازی های گرافی دیفرانسیلی سیستم های غیر خطی زمان پیوسته توسعه داده اند، که از ساختار عملگر-نقاد استفاده کرده است. در [۲۷-۳۰،۳۵، ۲۳]، سیاست های کنترلی پایدار ساز اولیه برای تضمین پایداری بازی گرافی دیفرانسیلی مورد نیاز است. شایان ذکر است که یافتن مجموعه ای از سیاست های کنترلی پایدارساز اولیه در بازی گرافی دیفرانسیلی کار راحت و سر راستی نیست.

[۱] Formation of a group of mobile robots
[۲] Unmanned air vehicle
[۳] vehicle formation control
[۴] Networked autonomous team
[۵] Leaderless consensus
[۶] Leader synchronization
[۷] Leader–follower consensus
[۸] Coupled Hamilton–Jacobi
[۹] Reinforcement learning
[۱۰] Approximate dynamic programming
[۱۱] Forwarded in time
[۱۲] Actor-Critic
[۱۳] Functional link
[۱۴] Coupled Hamilton–Jacobi-Issac

در این مقاله، یک طرح کنترلی زیر-بهینه جدید با استفاده از برنامه ریزی پویای تقریبی تک-شبکه، برای حل تقریبی بازی های گرافی دیفرانسیلی سیستم های غیرخطی زمان پیوسته، بدون نیاز به سیاست های کنترل اولیه پایدارساز پیشنهاد شده است. برای شبکه عصبی نقاد هر یک از عامل ها، الگوریتم تنظیم وزن جدیدی که ضامن پایداری دینامیکی حلقه بسته با مفهوم کراندار نهایی یکنواخت[۱] می باشد، ارایه شده است و در ادامه همگرایی پاسخ بهینه تقریبی به تعادل نش نیز اثبات شده است.

نوآوری های این مقاله بقرار زیر است:

۱- این مقاله، نتایج [۳۶] و [۳۷] را به بازی های گرافی دیفرانسیلی زمان پیوسته غیر خطی با N -بازیکن بسط می دهد که به دلیل فرمول نویسی بازی بصورت توزیع شده و مبتنی بر توپولوژی گراف و همچنین تعداد بازیکنان، نسبت به بازی های دیفرانسیلی مجموع صفر [۳۷] و مجموع غیر صفر [۳۶] با دو بازیکن، پیچیده تر می باشد. همچنین، پایداری سیستم حلقه بسته کل سیستم نیز ضمانت می شود.

۲- الگوریتم یادگیری توزیع شده پیشنهاد شده در این مقاله تنها از یک شبکه عصبی برای هر بازیکن استفاده می نماید. در نتیجه، این الگوریتم دارای بار محاسباتی کمتر و ساختار ساده تری برای پیاده سازی در مقایسه با [۲۳،۲۷-۳۰،۳۵]، که برای هر بازیکن از ساختار عملگر-نقاد با دو شبکه عصبی استفاده کرده است، می باشد.

۳- با معرفی اپراتورهای محلی توزیع شده جدید در قوانین تنظیم وزن ها، در مقایسه با [۲۳،۲۷-۳۰،۳۵]، دیگر هیچ نیازی به سیاست های کنترلی پایدارساز اولیه وجود ندارد.

۲- مقدمات و فرمول بندی مسئله

برخی مفاهیم پایه ای که در طول مقاله از آنها استفاده می شود در ادامه ذکر شده است. بعلاوه، فرمول بندی مسئله بازی های گرافی دیفرانسیلی N نفره برای سیستم های غیرخطی نیز در این بخش ذکر می شود.

۲-۱- گرافها و نمادها

نمادهای زیر در طول این مقاله مورد استفاده قرار می گیرند. $\Re$ اعداد حقیقی، $\Re^n$ بردارهای حقیقی n تایی و $\Re^{m\times n}$ ماتریس های حقیقی $m\times n$ را نشان می دهند. I_n ماتریس همانی با ابعاد $n\times n$ را نشان می دهد. $\|X\|$ نشان دهنده نرم اقلیدسی بردار X می باشد. $\|M\|$ نشان دهنده نرم-۲ القایی برای ماتریس M می باشد و $\zeta_i(M)$ نشان دهنده i امین مقدار منفرد ماتریس M و $\underline{\zeta}(M)$ کمترین مقدار منفرد ماتریس M را نشان می دهد. ضرب کرونکر دو ماتریس A و B به صورت $A\otimes B$ نشان داده می شود.

نحوه تعامل بین N عامل، توسط گراف $Gr(V,\Sigma)$ توصیف می شود که در آن $V=\{1,2,...,N\}$ مجموعه گره های گراف است که نماینده N عامل بوده و $\Sigma\subseteq V\times V$ مجموعه شاخه های گراف است که $(i,j)\in\Sigma$ به معنی وجود یک شاخه از گره i به گره j می باشد. در این مقاله گراف ساده فرض می شود، یعنی بین هر دو گره تنها یک شاخه وجود دارد و خود حلقه $(i,i)\notin\Sigma,\ \forall i$ در گراف وجود ندارد. توپولوژی یک گراف معمولا توسط ماتریس همسایگی آن $E=\left[e_{ij}\right]\in\Re^{N\times N}$ نمایش داده می شود به طوری که اگر $(j,i)\in\Sigma$ آنگاه $e_{ij}=1$ و در غیر این صورت $e_{ij}=0$ می باشد. $N_i^I=\{j:(j,i)\in\Sigma\}$ مجموعه همسایگان گره i است، به عبارت دیگر مجموعه گره ها با شاخه هایی است که به گره i وارد می شوند. $N_i^O=\{j:(i,j)\in\Sigma\}$ نیز نشان دهنده مجموعه ای از عامل ها هستند که عامل i در همسایگی آنها می باشد. ماتریس درجه-واردشونده[۲] $D=diag(d_i)\in\Re^{N\times N}$ ، یک ماتریس قطری است، با $d_i=\sum_{j\in N_i^I}e_{ij}$ که درجه-واردشونده گره i می باشد (یعنی مجموع عناصر سطر i ام E). ماتریس لاپلاسین گراف به صورت $L=D-E$ نمایش داده می شود و مجموع عناصر هر سطر آن صفر می باشد. مسیر، دنباله ای از گره های به هم متصل در یک گراف است و یک گراف را قویا متصل گویند اگر مسیری بین هر دو گره دلخواه آن وجود داشته باشد. معمولا گره رهبر توسط اندیس صفر نشان داده می شود.

در این مقاله گراف های ساده، قویا متصل و جهتدار با توپولوژی تغییرناپذیر با زمان برای نمایش توپولوژی تعاملی بین عامل ها در نظر گرفته می شوند.

۲-۲- فرمول بندی مسئله

یک گروه شامل N بازیکن ناهمگن[۳] را بر روی یک گراف ارتباطی قویا متصل و جهتدار در نظر بگیرید که دینامیک آنها به صورت زیر بیان می شود

$$\dot{x}_i=f_i(x_i)+g_i(x_i)u_i,\ i=1,...,N \qquad (۱)$$

که در آن $x_i\in\Re^n$ بردار حالت و $u_i\in\Re^m$ بردار ورودی کنترلی برای بازیکن i می باشد. همچنین دینامیک عامل رهبر $x_0\in\Re^n$ به صورت زیر داده می شود

$$\dot{x}_0=f_0(x_0) \qquad (۲)$$

[۱] Uniformly ultimately bounded

[۲] In-degree

[۳] Heterogeneous

که حداقل به یکی از بازیکن ها در گراف تعاملی متصل می باشد.

در این مقاله فرض می شود که $f_i(x_i)$، $f_0(x_0)$ و $g_i(x_i)$ برای $i=1,...,N$ لیپشیتز محلی هستند. خطای ردیابی همسایگان محلی برای هر بازیکن i به صورت زیر تعریف می شود

$$\delta_i = \sum_{j\in N_i^I} e_{ij}(x_i - x_j) + e_{i0}(x_i - x_0) \qquad (۳)$$

که در آن $e_{i0}\geq 0$ بهره اتصال می باشد که برای حداقل یک بازیکن، غیر صفر می باشد. توجه شود که اگر بازیکن i به صورت مستقیم با رهبر تعامل نماید، آنگاه $e_{i0}=1$ در غیر این صورت $e_{i0}=0$. دینامیک خطای ردیابی همسایگان محلی برای بازیکن i، $i=1,...,N$، به صورت زیر داده می شود [۳۵]

$$\begin{aligned}\dot{\delta}_i &= \sum_{j\in N_i^I} e_{ij}(f_i(x_i)-f_j(x_j)) + \\ &e_{i0}(f_i(x_i)-f_0(x_0)) + \\ &(d_i+e_{i0})g_i(x_i)u_i - \sum_{j\in N_i^I} e_{ij}g_j(x_j)u_j\end{aligned} \qquad (۴)$$

دینامیک خطای محلی هر عامل تحت تاثیر ورودی کنترلی خود عامل i و ورودی های کنترلی همسایگان این عامل می باشد. تابع هزینه محلی توزیع شده برای هر بازیکن به صورت زیر تعریف می شود [۲۳]

$$V_i(\delta_i,u_i,u_{N_i^I}) = \int_t^\infty r_i(\delta_i(\tau),u_i(\tau),u_{N_i^I}(\tau))d\tau \qquad (۵)$$

که در آن $u_{N_i^I}=\{u_j \mid j\in N_i^I\}$ و $r_i(\delta_i,u_i,u_{N_i^I})$ برابر با $1/2(Q_i(\delta_i)+u_i^T R_{ii}u_i+\sum_{j\in N_i^I}u_j^T R_{ij}u_j)$ و همچنین ماتریس های وزن $Q_i(\delta_i)>0, R_{ii}>0, R_{ij}>0$ متقارن و ثابت هستند.

توجه کنید که تابع هزینه محلی توزیع شده در بازی های گرافی دیفرانسیلی برای بازیکن i متاثر از ورودی بازیکن i و همسایگان آن می باشد. بنابراین، هدف کنترلی بازیکن i انتخاب سیاست کنترلی فیدبکی برای حداقل کردن تابع هزینه محلی و یافتن ارزش بهینه زیر می باشد

$$V_i^*(\delta_i) = \min_{u_i,u_{N_i^I}} \int_t^\infty r_i(\delta_i,u_i,u_{N_i^I})d\tau \qquad (۶)$$

فرمول معادل دیفرانسیلی رابطه (۵) به صورت معادله لیاپانوف غیرخطی زیر بیان می شود

$$\begin{aligned}&\nabla V_i^T(\sum_{j\in N_i^I} e_{ij}(f_i(x_i)-f_j(x_j)) + e_{i0}(f_i(x_i)-f_0(x_0)) \\ &+(d_i+e_{i0})g_i(x_i)u_i - \sum_{j\in N_i^I} e_{ij}g_j(x_j)u_j) + \frac{1}{2}Q_i(\delta_i) \\ &+\frac{1}{2}u_i^T R_{ii}u_i + \frac{1}{2}\sum_{j\in N_i^I} u_j^T R_{ij}u_j = 0, V_i(0)=0\end{aligned} \qquad (۷)$$

تابع هامیلتونین برای تابع هزینه محلی توزیع شده برای بازیکن i به صورت زیر تعریف می شود

$$\begin{aligned}H_i(\delta_i,u_i,u_{N_i^I}) \equiv &\frac{1}{2}Q_i(\delta_i) + \frac{1}{2}u_i^T R_{ii}u_i + \frac{1}{2}\times \\ &\sum_{j\in N_i^I} u_j^T R_{ij}u_j + \nabla V_i^T(\sum_{j\in N_i^I} e_{ij}(f_i(x_i)-f_j(x_j)) \\ &+e_{i0}(f_i(x_i)-f_0(x_0)) + (d_i+e_{i0})g_i(x_i)u_i \\ &-\sum_{j\in N_i^I} e_{ij}g_j(x_j)u_j)\end{aligned} \qquad (۸)$$

که در آن $\nabla V_i = \partial V_i/\partial\delta_i \in \Re^n$ مشتق جزیی $V_i(\delta_i)$ نسبت به δ_i می باشد.

تعریف ۱- [۲۳] سیاست های کنترلی u_i به ازای هر i، قابل قبول گفته می شوند، اگر u_i پیوسته و $u_i(0)=0$ باشد، همچنین u_i سیستم (۴) را به صورت محلی پایدار سازد و مقدار رابطه (۵) را محدود گرداند.

تعریف ۲- [۲۲] مجموعه سیاست های $\{u_1^*,u_2^*,...,u_N^*\}$ حل تعادل نش همه جایی برای بازی N نفره می باشد، اگر نامساوی های زیر $\forall u_i, u_{Gr-i}$ برقرار باشد

$$V_i(u_i^*,u_{Gr-i}^*) \leq V_i(u_i,u_{Gr-i}^*) \qquad (۹)$$

که در آن $u_{Gr-i}=\{u_j \mid j\neq i\}$.

لم ۱- سیستم (۴) و تابع هزینه محلی توزیع شده (۵) را در نظر بگیرید. بر اساس همیلتونین (۸)، سیاست های کنترلی فیدبک بهینه با استفاده از شرط ایستایی $\partial H_i/\partial u_i = 0$ [۳۸]، به صورت زیر به دست می آید

$$u_i^* = -(d_i+e_{i0})R_{ii}^{-1}g_i^T(x_i)\nabla V_i \qquad (۱۰)$$

که ∇V_i حل معادلات هامیلتون-جاکوبی کوپل شده (۱۱) است، که این معادلات با جایگذاری سیاست کنترلی فیدبک بهینه (۱۰) در (۸)، بصورت زیر به دست آمده اند.

$$\begin{aligned}&\frac{1}{2}Q_i(\delta_i) + \frac{1}{2}(d_i+e_{i0})^2\nabla V_i^T g_i(x_i)R_{ii}^{-1}\times \\ &g_i^T(x_i)\nabla V_i + \frac{1}{2}\sum_{j\in N_i^I}(d_j+e_{j0})^2\nabla V_j^T \times \\ &g_j(x_j)R_{jj}^{-1}R_{ij}R_{jj}^{-1}g_j^T(x_j)\nabla V_j + \nabla V_i^T \times \\ &(\sum_{j\in N_i^I} e_{ij}(f_i(x_i)-f_j(x_j)) + e_{i0}(f_i(x_i)-f_0(x_0)) \\ &-(d_i+e_{i0})^2 g_i(x_i)R_{ii}^{-1}g_i^T(x_i)\nabla V_i \\ &+\sum_{j\in N_i^I} e_{ij}(d_j+e_{j0})g_j(x_j)R_{jj}^{-1}g_j^T(x_j)\nabla V_j) = 0\end{aligned} \qquad (۱۱)$$

به طور کلی، یافتن راه حل های تحلیلی برای معادلات هامیلتون-جاکوبی کوپل شده دشوار و حتی غیر ممکن می باشد. در این مقاله، برای حل تقریبی این معادلات به صورت برخط، یک طرح کنترلی زیر-بهینه جدید که از برنامه ریزی پویای تقریبی تک-شبکه برای هر بازیکن استفاده می کند، پیشنهاد می شود.

فرض زیر در ادامه مقاله، مورد نیاز می باشد:

فرض ۱- برای هر بازیکن، یک کاندید لیاپانوف شعاعی بی کران دیفرانسیل پذیر پیوسته[۱] $J_i(\delta_i)$ وجود دارد به طوری که

$$\dot{J}_i = \nabla J_i^T \dot{\delta}_i = \nabla J_i^T \Big(\sum_{j\in N_i^I} e_{ij}(f_i(x_i)-f_j(x_j)) + e_{i0}(f_i(x_i)-f_0(x_0)) + (d_i+e_{i0})g_i(x_i)u_i^* - \sum_{j\in N_i^I} e_{ij} g_j(x_j)u_j^* \Big) < 0 \quad (۱۲)$$

که در آن $\nabla J_i = \partial J_i / \partial \delta_i \in \Re^n$ مشتق جزیی $J_i(\delta_i)$ نسبت به δ_i می باشد.

لم ۲ – سیستم غیرخطی (۴) با توابع هزینه محلی توزیع شده (۵) و سیاست های بهینه (۱۰) را در نظر بگیرید. ضمن برقراری فرض ۱، فرض کنید که ثابت مثبت $\bar{Q}_i$ وجود داشته باشد که در نامساوی زیر صدق نماید

$$\nabla V_i^{*T} \bar{Q}_i \nabla J_i \le r_i(\delta_i, u_i^*, u_{N_i^I}^*) \quad (۱۳)$$

آنگاه رابطه زیر برقرار می باشد

$$\nabla J_i^T \Big(\sum_{j\in N_i^I} e_{ij}(f_i(x_i)-f_j(x_j)) + e_{i0} \times (f_i(x_i)-f_0(x_0)) + (d_i+e_{i0})g_i(x_i)u_i^* - \sum_{j\in N_i^I} e_{ij} g_j(x_j)u_j^* \Big) \le -\nabla J_i^T \bar{Q}_i \nabla J_i \quad (۱۴)$$

اثبات لم ۲- با به کارگیری سیاستهای کنترلی فیدبک بهینه (۱۰) در سیستم غیرخطی (۴)، تابع هزینه محلی توزیع شده $V_i(\delta_i, u_i^*, u_{N_i^I}^*)$ در (۵)، تشکیل یک تابع لیاپانوف می دهد. آنگاه با استفاده از تابع هامیلتونین (۸) و مشتق گرفتن از تابع هزینه محلی توزیع شده $V_i(\delta_i, u_i^*, u_{N_i^I}^*)$ نسبت به t، برای $i = 1,...,N$ ، داریم

$$\dot{V}_i^* = \nabla V_i^{*T} \Big(\sum_{j\in N_i^I} e_{ij}(f_i(x_i)-f_j(x_j)) + e_{i0}(f_i(x_i)-f_0(x_0)) + (d_i+e_{i0})g_i(x_i)u_i^* - \sum_{j\in N_i^I} e_{ij} g_j(x_j)u_j^* \Big) = -r_i(\delta_i, u_i^*, u_{N_i^I}^*) \quad (۱۵)$$

با استفاده از (۱۳)، می توان (۱۵) را به صورت زیر بازنویسی کرد

$$\begin{aligned} &\sum_{j\in N_i^I} e_{ij}(f_i(x_i)-f_j(x_j)) + e_{i0}(f_i(x_i)-f_0(x_0)) \\ &+(d_i+e_{i0})g_i(x_i)u_i^* - \sum_{j\in N_i^I} e_{ij} g_j(x_j)u_j^* = \\ &-(\nabla V_i^* \nabla V_i^{*T})^{-1} \nabla V_i^* r_i(\delta_i, u_i^*, u_{N_i^I}^*) \\ &\le -(\nabla V_i^* \nabla V_i^{*T})^{-1} \nabla V_i^* \nabla V_i^{*T} \bar{Q}_i \nabla J_i \\ &\le -\bar{Q}_i \nabla J_i \end{aligned} \quad (۱۶)$$

در نهایت، با ضرب ∇J_i^T به هر دو طرف (۱۶)، (۱۴) به دست می آید که اثبات را کامل می نماید.

[۱] Continuously differentiable radially unbounded

۲- حل برخط بازی های گرافی دیفرانسیلی سیستم های چندعاملی غیرخطی با استفاده از برنامه ریزی پویای تقریبی تک-شبکه

بر اساس قضیه تقریب مرتبه بالای وایرشتراس [۴۰-۳۹] و نتیجه [۴۱]، مجموعه های پایه مستقل کامل $\sigma_i(\delta_i): \Re^n \to \Re^{K_i}$ برای $i = 1,...,N$ وجود دارند به طوری که $\sigma_i(0) = 0$، $\nabla\sigma_i(0) = 0$ ، آنگاه حل $V_i(\delta_i)$ و گرادیان آن به طور یکنواخت تخمین زده می شوند. بنابراین می توان در نظر گرفت که وزن $W_i \in \Re^{K_i}$ شبکه های عصبی نقاد وجود دارند به طوری که توابع هزینه $V_i(\delta_i)$ برای $i = 1,...,N$ به صورت زیر تقریب زده می شوند

$$V_i = V_i(\delta_i) = W_i^T \sigma_i(\delta_i) + \varepsilon_i(\delta_i) \quad (۱۷)$$

که $\sigma_i(\delta_i): \Re^n \to \Re^{K_i}$ توابع فعالیت شبکه های عصبی نقاد، K_i تعداد نرون ها در لایه مخفی عامل i و $\varepsilon_i(\delta_i)$ خطای تقریب شبکه های عصبی برای $i = 1,...,N$ می باشند. همانگونه که پیش تر گفته شد، توابع فعالیت شبکه عصبی نقاد $\sigma_i(\delta_i)$ طوری انتخاب می شوند که مجموعه های پایه مستقل کامل را طوری فراهم آورند که $V_i(\delta_i, u_i, u_{N_i^I})$ و گرادیان آنها برای $i = 1,...,N$

$$\nabla V_i = \nabla \sigma_i^T W_i + \nabla \varepsilon_i \quad (۱۸)$$

که $\nabla\varepsilon_i \to \partial\varepsilon_i / \partial\delta_i$ و $\nabla\sigma_i \to \partial\sigma_i / \partial\delta_i$، به طور یکنواخت تخمین زده می شوند. بنابراین هنگامی که تعداد نرون های لایه مخفی $K_i \to \infty$، خطای تقریب برای $i = 1,...,N$ به صورت یکنواخت $\varepsilon_i(\delta_i) \to 0$ ، $\nabla\varepsilon_i(\delta_i) \to 0$ [۴۰].

با جاگذاری (۱۸) در (۱۰) و (۱۱)، می توانیم سیاست های بهینه (۱۰) و معادلات هامیلتون-جاکوبی کوپل شده (۱۱) را به ترتیب زیر بازنویسی کنیم

$$u_i^* = -(d_i+e_{i0})R_{ii}^{-1} g_i^T(x_i)\nabla\sigma_i^T W_i - (d_i+e_{i0})R_{ii}^{-1} g_i^T(x_i)\nabla\varepsilon_i \quad (۱۹)$$

$$\frac{1}{2}Q_i(\delta_i) - \frac{1}{2}(d_i+e_{i0})^2 W_i^T \nabla\sigma_i D_i \nabla\sigma_i^T W_i + \frac{1}{2}\sum_{j\in N_i^I}(d_j+e_{j0})^2 W_j^T \nabla\sigma_j S_{ij} \nabla\sigma_j^T W_j + W_i^T \times \nabla\sigma_i \Big(\sum_{j\in N_i^I} e_{ij}(f_i(x_i)-f_j(x_j)) + e_{i0}(f_i(x_i)-f_0(x_0)) \Big) + \sum_{j\in N_i^I} e_{ij}(d_j+e_{j0}) D_j \nabla\sigma_j^T W_j - \varepsilon_{HJ_i} = 0 \quad (۲۰)$$

که در آن

$$D_i = g_i(x_i) R_{ii}^{-1} g_i^T(x_i) \quad (۲۱)$$

$$S_{ij} = g_j(x_j)R_{jj}^{-1}R_{ij}R_{jj}^{-1}g_j^T(x_j) \quad (۲۲)$$

و خطای باقی مانده با ε_{HJ_i} نشان داده شده است.

وزن های شبکه های عصبی نقاد W_i برای $i=1,...,N$ نامعلوم هستند و باید به صورت برخط تخمین زده شوند. $\hat{W}_i$ را مقدار تخمین زده شده W_i برای هر بازیکن i ، برای $i=1,...,N$ در نظر می گیریم.

بنابراین خروجی شبکه عصبی نقاد به صورت زیر می باشد

$$\hat{V}_i = \hat{W}_i^T\sigma_i(\delta_i) \quad (۲۳)$$

با جاگذاری (۲۳) در (۱۰)، تخمین سیاست های کنترلی بهینه به صورت زیر به دست می آید

$$\hat{u}_i = -(d_i+e_{i0})R_{ii}^{-1}g_i^T(x_i)\nabla\sigma_i^T\hat{W}_i \quad (۲۴)$$

با اعمال کردن (۲۴) در سیستم (۴)، دینامیک حلقه بسته سیستم به صورت زیر به دست می آید

$$\begin{aligned}\dot{\delta}_i(\hat{W}_i,\hat{W}_j) = \dot{\delta}_i &= \sum_{j\in N_i^I} e_{ij}(f_i(x_i)-f_j(x_j))\\ &+e_{i0}(f_i(x_i)-f_0(x_0))-(d_i+e_{i0})^2D_i\nabla\sigma_i^T\\ &\times\hat{W}_i+\sum_{j\in N_i^I}e_{ij}(d_j+e_{j0})D_j\nabla\sigma_j^T\hat{W}_j\end{aligned} \quad (۲۵)$$

با جایگذاری (۲۳) و (۲۴) در (۸)، توابع هامیلتونین تقریبی به صورت زیر به دست می آید

$$\begin{aligned}H_i(\delta_i,\hat{W}_i,\hat{W}_j) &= \frac{1}{2}Q_i(\delta_i)+\frac{1}{2}(d_i+e_{i0})^2\hat{W}_i^T\times\\ &\nabla\sigma_iD_i\nabla\sigma_i^T\hat{W}_i+\frac{1}{2}\sum_{j\in N_i^I}(d_j+e_{j0})^2\hat{W}_j^T\nabla\sigma_jS_{ij}\times\\ &\nabla\sigma_j^T\hat{W}_j+\hat{W}_i^T\nabla\sigma_i(\sum_{j\in N_i^I}e_{ij}(f_i(x_i)-f_j(x_j))+\\ &e_{i0}(f_i(x_i)-f_0(x_0))-(d_i+e_{i0})^2D_i\nabla\sigma_i^T\hat{W}_i+\\ &\sum_{j\in N_i^I}e_{ij}(d_j+e_{j0})D_j\nabla\sigma_j^T\hat{W}_j)=e_i\end{aligned} \quad (۲۶)$$

برای به دست آوردن مقادیر ایده آل وزن های شبکه های عصبی نقاد، مطلوب است تا مربع خطای باقی مانده e_i در (۲۶) حداقل شود.

برای این منظور، از گرادیان نزولی نرمالایز شده برای انتخاب $\hat{W}_i$ برای $i=1,...,N$ استفاده می شود. در اینجا قوانین تنظیم وزن جدید برای شبکه های عصبی نقاد برای N بازیکن که علاوه بر حداقل ساختن مربع خطای باقی مانده (۲۷) می تواند پایداری سیستم را برآورده کند، ارایه می گردد.

$$E=\sum_{i=1}^{N}E_i=\frac{1}{2}\sum_{i=1}^{N}e_i^Te_i \quad (۲۷)$$

قانون تنظیم پیشنهادی برای شبکه عصبی نقاد هر عامل i برای $i=1,...,N$ به صورت زیر به دست می آید

$$\begin{aligned}\dot{\hat{W}}_i &= -\alpha_i\frac{\bar{B}_i}{m_{s_i}}(\frac{1}{2}Q_i(\delta_i)+\frac{1}{2}(d_i+e_{i0})^2\times\\ &\hat{W}_i^T\nabla\sigma_iD_i\nabla\sigma_i^T\hat{W}_i+\frac{1}{2}\sum_{j\in N_i^I}(d_j+e_{j0})^2\times\\ &\hat{W}_j^T\nabla\sigma_jS_{ij}\nabla\sigma_j^T\hat{W}_j+\hat{W}_i^T\nabla\sigma_i(\sum_{j\in N_i^I}e_{ij}\times\\ &(f_i(x_i)-f_j(x_j))+e_{i0}(f_i(x_i)-f_0(x_0))\\ &-(d_i+e_{i0})^2D_i\nabla\sigma_i^T\hat{W}_i+\sum_{j\in N_i^I}e_{ij}(d_j+e_{j0})\\ &\times D_j\nabla\sigma_j^T\hat{W}_j))+\frac{1}{2}\alpha_i(d_i+e_{i0})^2\nabla\sigma_iD_i\times\\ &\nabla\sigma_i^T\hat{W}_i\frac{\bar{B}_i^T}{m_{s_i}}\hat{W}_i+\frac{1}{2}\alpha_i\lambda_i^{-1}(d_i+e_{i0})^2\nabla\sigma_i\times\\ &\sum_{j\in N_i^O}\lambda_j\hat{W}_j^T\frac{\bar{B}_j}{m_{s_j}}S_{ji}\nabla\sigma_i^T\hat{W}_i-\bar{\Sigma}_i(\lambda_i^{-1}\alpha_i\times\\ &(d_i+e_{i0})\nabla\sigma_iD_i(\sum_{j\in N_i^O}e_{ji}\nabla J_j\\ &-(d_i+e_{i0})\nabla J_i))+\lambda_i^{-1}\alpha_i(d_i+e_{i0})\nabla\sigma_i\times\\ &D_i(\bar{\Sigma}_i\sum_{j\in N_i^O}e_{ji}\Sigma_j\nabla J_j-\Sigma_i\sum_{j\in N_i^O}e_{ji}\bar{\Sigma}_j\nabla J_j)\\ &-\alpha_iF_{1i}\frac{\nabla\sigma_i\nabla\sigma_i^T}{1+\left\|\nabla\sigma_i\nabla\sigma_i^T\right\|}\hat{W}_i\\ &-\alpha_iF_{2i}\begin{bmatrix}e_{i1}\dfrac{\nabla\sigma_1\nabla\sigma_1^T}{1+\left\|\nabla\sigma_1\nabla\sigma_1^T\right\|}\hat{W}_1\\ \vdots\\ e_{iN}\dfrac{\nabla\sigma_N\nabla\sigma_N^T}{1+\left\|\nabla\sigma_N\nabla\sigma_N^T\right\|}\hat{W}_N\end{bmatrix}\end{aligned} \quad (۲۸)$$

که در آن $\bar{B}_i=B_i/m_{s_i}$ ، $m_{s_i}=1+B_i^TB_i$ ، $\alpha_i>0$ بهره تطبیقی، F_{1i} و F_{2i} ماتریس های طراحی برای $i=1,...,N$ می باشند، همچنین ∇J_i برای $i=1,...,N$ در لم ۲ ذکر شده است. بعلاوه، B_i که با مشتق جزیی گرفتن از e_i نسبت به $\hat{W}_i$ به دست می آید به صورت زیر می باشد

$$\begin{aligned}B_i &= \nabla\sigma_i(\sum_{j\in N_i^I}e_{ij}(f_i(x_i)-f_j(x_j))\\ &+e_{i0}(f_i(x_i)-f_0(x_0))+\sum_{j\in N_i^I}e_{ij}(d_j+e_{j0})\\ &\times D_j\nabla\sigma_j^T\hat{W}_j)-\nabla\sigma_i(d_i+e_{i0})^2D_i\nabla\sigma_i^T\hat{W}_i\end{aligned} \quad (۲۹)$$

اپراتورهای سوئیچ $\bar{\Sigma}_i$ و Σ_i برای $i=1,...,N$ به صورت زیر تعریف می شود

$$\bar{\Sigma}_i = \begin{cases} 0 & i \in S \\ 1 & i \in \bar{S} \end{cases} \qquad (۳۰)$$

$$\Sigma_i = \begin{cases} 1 & i \in S \\ 0 & i \in \bar{S} \end{cases} \qquad (۳۱)$$

که $S = \{i : \nabla J_i \dot{\delta}_i < 0 \,\&\, \nabla J_j \dot{\delta}_j\}_{j \in N_i^O}$ و $\bar{S} = \{i : i \notin S\}$.

با به کار گیری الگوریتم گرادیان نزولی نرمالایز شده، از جملات ابتدایی (۲۸) برای حداقل کردن خطای باقی مانده مربعی (۲۷) استفاده می شود و از باقی جملات، برای تضمین پایداری سیستم حلقه بسته در زمان یادگیری وزن های بهینه شبکه های عصبی نقاد استفاده می شود.

ملاحظه ۱- توجه شود که $\bar{\Sigma}_i = 0$ و $\Sigma_i = 1$ به این معناست که دینامیک های خطای ردیابی محلی بازیکن i و بازیکنانی که i در همسایگی آنهاست پایدار است. از طرف دیگر، $\bar{\Sigma}_i = 1$ و $\Sigma_i = 0$ به این معناست که حداقل دینامیک خطای ردیابی محلی یکی از این بازیکنان ناپایدار شده است. به این ترتیب با استفاده از اپراتورهای سوئیچ (۳۰) و (۳۱)، نیاز به سیاست کنترلی اولیه قابل قبول برای هر عامل که در [۲۳] و [۳۵] احتیاج است، برطرف می شود. توجه شود که جملاتی که به وسیله اپراتورهای سوئیچ (۳۰) و (۳۱) فعال می شوند، طبق شرایط کافی لیاپانوف برای پایداری انتخاب شده اند.

ملاحظه ۲- به منظور حداقل کردن خطاهای باقی مانده مربعی (۲۷)، نیاز است تا حالتهای سیستم (۴) به اندازه کافی تحریک پایا[۱] باشند. در نتیجه برای برآورده کردن شرط تحریک پایا، یک نویز تحریک به ورودی های کنترلی اضافه می شود. خطای تخمین وزن شبکه های عصبی نقاد برای هر عامل i، $i = 1,...,N$ بصورت $\tilde{W}_i = W_i - \hat{W}_i$ تعریف می شود.

۳- تحلیل پایداری

در این قسمت، پایداری بازی گرافی دیفرانسیلی با قوانین تنظیم وزن پیشنهادی شبکه های عصبی نقاد (۲۸) مورد تحلیل قرار می گیرد. ابتدا فرضیات زیر را در نظر می گیریم.

فرض ۲-

۱) $g_i(.)$ ها کراندار هستند، یعنی $\|g_i(.)\| \le g_{iM}$ که g_{iM} ثابت هایی مثبت می باشند برای $i = 1,...,N$.

۲) خطاهای تقریب شبکه عصبی نقاد و گرادیان آن کراندار هستند، به طوری که $\|\varepsilon_i\| \le \varepsilon_{iM}$ و $\|\nabla \varepsilon_i\| \le \varepsilon_{idM}$ با $\varepsilon_{iM} > 0$ و $\varepsilon_{idM} > 0$ برای $i = 1,...,N$.

۳) توابع فعالیت شبکه های عصبی نقاد کراندار هستند، به طوری که $\|\sigma_i\| \le \sigma_{iM}$ و $\|\nabla \sigma_i\| \le \sigma_{idM}$ با $\sigma_{iM} > 0$ و $\sigma_{idM} > 0$ برای $i = 1,...,N$.

۴) وزن های شبکه های عصبی نقاد کرندار هستند، به طوری که $\|W_i\| \le W_{iM}$ ، با $W_{iM} > 0$ برای $i = 1,...,N$.

۵) خطاهای باقی مانده ε_{HJ_i} کراندار هستند، به طوری که $\|\varepsilon_{HJ_i}\| \le \varepsilon_{HJ_{iM}}$ با $\varepsilon_{HJ_{iM}} > 0$ برای $i = 1,...,N$.

قضیه ۱- سیستم (۴) را در نظر بگیرید، سیاست های کنترلی توسط رابطه (۲۴) داده شده اند و قانون تنظیم وزن شبکه عصبی نقاد برای هر عامل توسط (۲۸) فراهم شده است. در نظر می گیریم که فرضیات ۱ و ۲ برقرار هستند. آنگاه حالتهای سیستم δ_i و خطای تخمین وزن شبکه های عصبی نقاد $\tilde{W}_i$ برای $i = 1,...,N$، به ازای تعداد کافی نرون، کراندار نهایی یکنواخت هستند.

اثبات قضیه ۱- تابع لیاپانوف زیر را در نظر بگیرید

$$J = \sum_{i=1}^{N} \{J_i(\delta_i) + \frac{1}{2} \lambda_i \tilde{W}_i^T \alpha_i^{-1} \tilde{W}_i\} \qquad (۳۲)$$

که $J_i(\delta_i)$ برای $i = 1,...,N$ از لم ۲ به دست آمده اند.

مشتق زمانی تابع لیاپانوف (۳۲) با توجه به سیستم (۲۵) به صورت زیر به دست می آید

[۱] Persistent Excitation

$$\dot{J}=\sum_{i=1}^{N}\nabla J_i^T\Big(\sum_{j\in N_i^I}e_{ij}(f_i(x_i)-f_j(x_j))$$
$$+e_{i0}(f_i(x_i)-f_0(x_0))-(d_i+e_{i0})^2D_i\nabla\sigma_i^T\hat{W}_i$$
$$+\sum_{j\in N_i^I}e_{ij}(d_j+e_{j0})D_j\nabla\sigma_j^T\hat{W}_j-\tilde{W}_i^T\bar{B}_i\lambda_i\bar{B}_i^T\tilde{W}_i$$
$$+\lambda_i\tilde{W}_i^T\bar{B}_i\frac{\varepsilon_{HJ_i}}{m_{s_i}}-\frac{1}{2}\tilde{W}_i^T\lambda_i(d_i+e_{i0})^2\nabla\sigma_iD_i\times$$
$$\nabla\sigma_i^TW_i\frac{\bar{B}_i^T}{m_{s_i}}W_i+\frac{1}{2}\lambda_i\tilde{W}_i^T(d_i+e_{i0})^2\nabla\sigma_iD_i\times$$
$$\nabla\sigma_i^TW_i\frac{\bar{B}_i^T}{m_{s_i}}\tilde{W}_i+\frac{1}{2}\lambda_i\tilde{W}_i^T(d_i+e_{i0})^2\frac{\bar{B}_i^T}{m_{s_i}}W_i\times$$
$$\nabla\sigma_iD_i\nabla\sigma_i^T\tilde{W}_i-\frac{1}{2}\tilde{W}_i^T(d_i+e_{i0})^2\nabla\sigma_i\times$$
$$\sum_{j\in N_i^O}\lambda_jS_{ji}\nabla\sigma_i^TW_i\frac{\bar{B}_j^T}{m_{s_j}}W_j+\frac{1}{2}\tilde{W}_i^T(d_i+e_{i0})^2\times$$
$$\nabla\sigma_i\sum_{j\in N_i^O}\frac{\bar{B}_j^T}{m_{s_j}}W_j\lambda_jS_{ji}\nabla\sigma_i^T\tilde{W}_i\quad-\frac{1}{2}\lambda_i\tilde{W}_i^T\frac{\bar{B}_i}{m_{s_i}}$$
$$\times\begin{bmatrix}\nabla\sigma_1S_{i1}\nabla\sigma_1^TW_1\\ \vdots\\ \nabla\sigma_NS_{iN}\nabla\sigma_N^TW_N\end{bmatrix}^T\begin{bmatrix}e_{i1}(d_1+e_{10})^2\tilde{W}_1\\ \vdots\\ e_{iN}(d_N+e_{N0})^2\tilde{W}_N\end{bmatrix}$$
$$+\lambda_i\tilde{W}_i^TF_{1i}\frac{\nabla\sigma_i\nabla\sigma_i^T}{1+\left\|\nabla\sigma_i\nabla\sigma_i^T\right\|}\hat{W}_i-\lambda_i\tilde{W}_i^T\frac{\bar{B}_i}{m_{s_i}}\times$$
$$W_i^T\nabla\sigma_i\begin{bmatrix}\nabla\sigma_1D_1\\ \vdots\\ \nabla\sigma_ND_N\end{bmatrix}^T\begin{bmatrix}e_{i1}(d_1+e_{10})\tilde{W}_1\\ \vdots\\ e_{iN}(d_N+e_{N0})\tilde{W}_N\end{bmatrix}\qquad(۳۳)$$
$$+\lambda_i\tilde{W}_i^TF_{2i}\begin{bmatrix}e_{i1}\frac{\nabla\sigma_1\nabla\sigma_1^T}{1+\left\|\nabla\sigma_1\nabla\sigma_1^T\right\|}\hat{W}_1\\ \vdots\\ e_{iN}\frac{\nabla\sigma_N\nabla\sigma_N^T}{1+\left\|\nabla\sigma_N\nabla\sigma_N^T\right\|}\hat{W}_N\end{bmatrix}\Big\}$$
$$+\sum_{i\in\bar{S}}\{\nabla J_i^T\sum_{j\in N_i^I}e_{ij}(d_j+e_{j0})D_j\nabla\sigma_j^T\tilde{W}_j$$
$$-\tilde{W}_i^T(d_i+e_{i0})^2\nabla\sigma_iD_i\nabla J_i\}$$

برای نوشتن (۳۳) به صورت فشرده، $Z^T=\begin{bmatrix}\tilde{W}_1^T & \cdots & \tilde{W}_N^T\end{bmatrix}$ را تعریف می کنیم، اکنون می توان (۳۳) را به صورت زیر بازنویسی کرد

$$\dot{J}=-Z^T\overbrace{\begin{bmatrix}m_{11} & \cdots & \frac{m_{1N}+m_{N1}^T}{2}\\ \vdots & \ddots & \vdots\\ \frac{m_{N1}+m_{1N}^T}{2} & \cdots & m_{NN}\end{bmatrix}}^{M}Z$$
$$+Z^Td+\sum_{i=1}^{N}\{\nabla J_i^T\Big(\sum_{j\in N_i^I}e_{ij}(f_i(x_i)-f_j(x_j))\qquad(۳۴)$$
$$+e_{i0}(f_i(x_i)-f_0(x_0))-(d_i+e_{i0})^2D_i\nabla\sigma_i^T\hat{W}_i$$
$$+\sum_{j\in N_i^I}e_{ij}(d_j+e_{j0})D_j\nabla\sigma_j^T\hat{W}_j+\sum_{i\in\bar{S}}\{\nabla J_i^T\times$$
$$\sum_{j\in N_i^I}e_{ij}(d_j+e_{j0})D_j\nabla\sigma_j^T\tilde{W}_j-\tilde{W}_i^T(d_i+e_{i0})^2$$
$$\times\nabla\sigma_iD_i\nabla J_i\}$$

که در آن اجزای ماتریس M به صورت زیر داده شده اند.

$$m_{ii}=-\frac{1}{2}(d_i+e_{i0})^2\nabla\sigma_i\sum_{j\in N_i^O}\frac{\bar{B}_j^T}{m_{s_j}}W_j\lambda_jS_{ji}\times$$
$$\nabla\sigma_i^T-\frac{1}{2}\lambda_i(d_i+e_{i0})^2\nabla\sigma_iD_i\nabla\sigma_i^TW_i\frac{\bar{B}_i^T}{m_{s_i}}+\qquad(۳۵)$$
$$\bar{B}_i\lambda_i\bar{B}_i^T-\frac{1}{2}\lambda_i(d_i+e_{i0})^2\frac{\bar{B}_i^T}{m_{s_i}}W_i\nabla\sigma_iD_i\nabla\sigma_i^T$$
$$+\lambda_iF_{1i}\frac{\nabla\sigma_i\nabla\sigma_i^T}{1+\left\|\nabla\sigma_i\nabla\sigma_i^T\right\|}$$
$$m_{ij}=\lambda_ie_{ij}(d_j+e_{j0})\frac{\bar{B}_i}{m_{s_i}}W_i^T\nabla\sigma_iD_j\nabla\sigma_j^T$$
$$+\frac{1}{2}\lambda_ie_{ij}(d_j+e_{j0})^2\frac{\bar{B}_i}{m_{s_i}}W_j^T\nabla\sigma_jS_{ij}\nabla\sigma_j^T$$
$$+\lambda_iF_{2i}\left(\begin{bmatrix}0&0&0&0&\cdots&0\\0&0&e_{ij}&0&\cdots&0\\0&0&0&0&\cdots&0\\0&0&0&0&\cdots&0\\\vdots&\vdots&\vdots&\vdots&\ddots&0\\0&0&0&0&0&0\end{bmatrix}\otimes I_K\right)$$
$$\times\begin{bmatrix}\frac{\nabla\sigma_1\nabla\sigma_1^T}{1+\left\|\nabla\sigma_1\nabla\sigma_1^T\right\|}\hat{W}_1\\ \vdots\\ \frac{\nabla\sigma_N\nabla\sigma_N^T}{1+\left\|\nabla\sigma_N\nabla\sigma_N^T\right\|}\hat{W}_N\end{bmatrix}\qquad(۳۶)$$

و اجزای بردار $d^T=\begin{bmatrix}d_1^T & \cdots & d_N^T\end{bmatrix}$ به صورت زیر به دست می آیند

$$d_i=-\frac{1}{2}\lambda_i(d_i+e_{i0})^2\nabla\sigma_iD_i\nabla\sigma_i^TW_i\frac{\bar{B}_i^T}{m_{s_i}}W_i$$

$$-\frac{1}{2}(d_i+e_{i0})^2\nabla\sigma_i\sum_{j\in N_i^O}\lambda_j S_{ji}\nabla\sigma_i^T W_i\frac{\bar{B}_j^T}{m_{s_j}}W_j \quad (۳۷)$$
$$+\lambda_i\bar{B}_i\frac{\varepsilon_{HJ_i}}{m_{s_i}}+\lambda_i F_{1i}\frac{\nabla\sigma_i\nabla\sigma_i^T}{1+\left\|\nabla\sigma_i\nabla\sigma_i^T\right\|}W_i+$$
$$\lambda_i F_{2i}\begin{bmatrix} e_{i1}\dfrac{\nabla\sigma_1\nabla\sigma_1^T}{1+\left\|\nabla\sigma_1\nabla\sigma_1^T\right\|}W_1 \\ \vdots \\ e_{iN}\dfrac{\nabla\sigma_N\nabla\sigma_N^T}{1+\left\|\nabla\sigma_N\nabla\sigma_N^T\right\|}W_N\end{bmatrix}$$

پارامترهای طراحی λ_i و ماتریس های طراحی F_{1i} و F_{2i} برای $i=1,...,N$ باید به گونه ای انتخاب شوند که $M>0$ باشد.

تحت شرط تحریک پایا داریم $\|\delta_i\|>0$ ، که وجود ثابت های δ_{idmin} را که در $\|\dot{\delta}_i\|<\delta_{idmin}<0$ صدق می کند را تضمین می نماید. بنابراین داریم

$$\sum_{i\in S}\{\nabla J_i^T(\sum_{j\in N_i^I}e_{ij}(f_i(x_i)-f_j(x_j))+$$
$$e_{i0}(f_i(x_i)-f_0(x_0))-(d_i+e_{i0})^2D_i\times \quad (۳۸)$$
$$\nabla\sigma_i^T\hat{W}_i+\sum_{j\in N_i^I}e_{ij}(d_j+e_{j0})D_j\nabla\sigma_j^T\hat{W}_j)\}$$
$$<-\sum_{i\in S}\{\delta_{id\min}\|\nabla J_i\|\}<0$$

طبق فرض ۲ و این حقیقت که $\bar{B}_i<1$ ، برای $i=1,...,N$ ، می توان نشان داد $\|D_i\|\le D_{iM}$ و $\|d\|\le d_M$ ، که D_{iM} و d_M هر دو یک ثابت مثبت مشخص می باشند. اکنون (۳۴) به صورت زیر در می آید.

$$\dot{J}\le-\|Z\|^2\underline{\zeta}(M)+\|Z\|d_M-\sum_{i\in S}\{\delta_{id\min}\|\nabla J_i\|\}$$
$$+\sum_{i\in\bar{S}}\{\nabla J_i^T(\sum_{j\in N_i^I}e_{ij}(f_i(x_i)-f_j(x_j))+$$
$$e_{i0}(f_i(x_i)-f_0(x_0))-(d_i+e_{i0})^2D_i\nabla\sigma_i^T\hat{W}_i \quad (۳۹)$$
$$+\sum_{j\in N_i^I}e_{ij}(d_j+e_{j0})D_j\nabla\sigma_j^T\hat{W}_j)\}+\sum_{i\in\bar{S}}\{\nabla J_i^T\times$$
$$\sum_{j\in N_i^I}e_{ij}(d_j+e_{j0})D_j\nabla\sigma_j^T\tilde{W}_j-\tilde{W}_i^T\times$$
$$(d_i+e_{i0})^2\nabla\sigma_iD_i\nabla J_i\}$$

که در آن کوچکترین مقدار منفرد ماتریس M با $\underline{\zeta}(M)$ نشان داده شده است.

ملاحظه ۳- توجه داشته باشید که با انتخاب مناسب پارامترهای طراحی λ_i و ماتریس های طراحی F_{1i} و F_{2i} برای $i=1,...,N$ می توانیم $\underline{\zeta}(M)$ را افزایش دهیم.

با توجه به (۱۰) و (۲۰) و اضافه و کم نمودن جملات زیر

$$\sum_{i\in\bar{S}}\{\nabla J_i^T(\sum_{j\in N_i^I}e_{ij}(d_j+e_{j0})D_j\nabla\varepsilon_j \quad (۴۰)$$
$$-(d_i+e_{i0})^2D_i\nabla\varepsilon_i)\}$$

به سمت راست (۳۹) داریم

$$\dot{J}\le-\|Z\|^2\underline{\zeta}(M)+\|Z\|d_M-\sum_{i\in S}\{\delta_{id\min}\|\nabla J_i\|\}$$
$$+\sum_{i\in\bar{S}}\{\nabla J_i^T(\sum_{j\in N_i^I}e_{ij}(f_i(x_i)-f_j(x_j))+ \quad (۴۱)$$
$$e_{i0}(f_i(x_i)-f_0(x_0))-(d_i+e_{i0})g_i(x_i)u_i^*$$
$$+\sum_{j\in N_i^I}e_{ij}g_j(x_j)u_j^*+(d_i+e_{i0})^2D_i\nabla\varepsilon_i$$
$$-\sum_{j\in N_i^I}e_{ij}(d_j+e_{j0})D_j\nabla\varepsilon_j)\}$$

با استفاده از لم ۲، (۴۱) به صورت زیر بازنویسی می شود

$$\dot{J}\le-\sum_{i\in S}\{\delta_{id\min}\|\nabla J_i\|\}+\sum_{i=1}^{N}\{\frac{\eta_i^2}{4\bar{Q}_{i\min}}\}$$
$$-\sum_{i\in\bar{S}}\{\bar{Q}_{i\min}(\|\nabla J_i\|-\frac{\eta_i}{2\bar{Q}_{i\min}})^2\}+\frac{d_M^2}{4\underline{\zeta}(M)} \quad (۴۲)$$
$$-\underline{\zeta}(M)(\|Z\|-\frac{d_M}{2\underline{\zeta}(M)})^2$$

که در آن

$$\eta_i=(d_i+e_{i0})^2D_{iM}\varepsilon_{idM}+$$
$$\sum_{j\in N_i^I}e_{ij}(d_j+e_{j0})D_{jM}\varepsilon_{jdM} \quad (۴۳)$$

توجه شود که $\nabla\varepsilon_i$ برای $i=1,...,N$ و d از بالا کراندار هستند.

اکنون اگر یکی از نامساوی های زیر به ازای یکی از i ها، $i=1,...,N$

$$\|\nabla J_{i\in\bar{S}}\|>\sqrt{\frac{\left(\sum_{i\in\bar{S}}^{N}\left\{\frac{\eta_i^2}{4\bar{Q}_{i\min}}\right\}+\frac{d_M^2}{4\underline{\zeta}(M)}\right)}{\bar{Q}_{i\min}}}+\frac{\eta_i}{2\bar{Q}_{i\min}}\square B_{\nabla J_i}^{\bar{S}} \quad (۴۴)$$

$$\|\nabla J_{i\in S}\|>\sqrt{\frac{\left(\sum_{i\in\bar{S}}^{N}\left\{\frac{\eta_i^2}{4\bar{Q}_{i\min}}\right\}+\frac{d_M^2}{4\underline{\zeta}(M)}\right)}{\delta_{id\min}}}\square B_{\nabla J_i}^{S} \quad (۴۵)$$

$$\|Z\|>\sqrt{\frac{\sum_{i\in\bar{S}}^{N}\left\{\frac{\eta_i^2}{4\bar{Q}_{i\min}}\right\}+\frac{d_M^2}{4\underline{\zeta}(M)}}{\underline{\zeta}(M)}}+\frac{d_M}{2\underline{\zeta}(M)}\square B_Z^{'} \quad (۴۶)$$

برقرار باشد، آنگاه $\dot{J}<0$. بنابراین طبق تئوری پایداری لیاپانوف [۴۲]، نتیجه می شود که $\|Z\|>B_Z$ و $\|\nabla J_i\|>\max(B_{\nabla J_i}^{S},B_{\nabla J_i}^{\bar{S}})\square\bar{B}_{\nabla J_i}$ برای $i=1,...,N$

آنگاه $\dot{J}<0$ و $\|\nabla J_i\|$ و $\|Z\|$ کراندار نهایی یکنواخت هستند، یعنی $\|\nabla J_i\|<\bar{B}_{\nabla J_i}$ برای $i=1,...,N$ و $\|Z\|<\bar{B}_Z$. توجه شود که اگر یکی از اجزای Z از باند $\bar{B}_Z$ فراتر رود، یعنی $\|\tilde{W}_i\|>\bar{B}_Z$ برای یک i ، آنگاه با توجه به این که Z توسط باند $\bar{B}_Z$ کراندار نهایی یکنواخت است، می توان مشاهده کرد که خطاهای تخمین وزن شبکه عصبی نقاد، یعنی $\|\tilde{W}_i\|$ برای $i=1,...,N$ نیز توسط باند $\bar{B}_Z$ کراندار نهایی یکنواخت هستند. طبق فرض ۱، کراندار $\|\nabla J_i\|$ ها، کراندار بودن $\|\delta_i\|$ ها را برای $i=1,...,N$ نشان می دهد. بنابراین $\|\delta_i\|\le\bar{B}_{\delta_i}$ است برای $i=1,...,N$ ، که $\bar{B}_{\delta_i}$ ها توسط $\bar{B}_{\nabla J_i}$ ها برای $i=1,...,N$ تعیین می شوند.

نتیجه فرعی ۱- با اثبات قضیه ۱، همگرایی سیاست های $\hat{u}_i$ برای $i=1,...,N$ به استراتژی تعادل نش تقریبی بازی گرافی دیفرانسیلی نیز بدست می آید.

اثبات -طبق فرض ۲ و کراندار $\|\tilde{W}_i\|$ برای $i=1,...,N$ ، از (۱۰) و (۲۴) داریم

$$\begin{aligned}\|\hat{u}_i-u_i^*\|&\le\|(d_i+e_{i0})R_{ii}^{-1}g_i^T(x_i)\nabla\sigma_i^T\tilde{W}_i\|\\&\le(d_i+e_{i0})\lambda_{\max}(R_{ii}^{-1})g_{iM}\sigma_{idM}\bar{B}_Z\equiv\in_{u_i}\end{aligned}\quad(۴۷)$$

که همگرایی به تعادل نش تقریبی بازی را تضمین می کند.

ملاحظه ٤- از (۴۷) مشاهده می شود که خطای همگرایی $\in_{u_i}$ می تواند با کاهش $\bar{B}_Z$ کاهش پیدا کند که آن نیز با افزایش $\underline{\zeta}(M)$ و کاهش d_M ، کاهش پیدا می کند. آنگاه با انتخاب مناسب پارامترهای λ_i و ماتریس های طراحی F_{1i} و F_{2i} برای $i=1,...,N$ خطای همگرایی $\in_{u_i}$ در (۴۷) را می توانیم کاهش دهیم.

٤- نتایج شبیه‌سازی

در این قسمت مثال شبیه سازی برای نمایش موثر بودن الگوریتم ارایه شده، برای بازی های گرافی دیفرانسیلی چندعاملی بدون نیاز به کنترل پایدارساز اولیه، ارایه می گردد.

گراف ارتباطی شامل ۵ عامل و یک رهبر که در شکل ۱ نشان داده شده است را در نظر بگیرید. بهره های اتصال و وزن یال ها برابر یک می باشند.

مشابه [۳۵]، دینامیک عامل ها برای $i=1,...,5$ ، $x_i\equiv[x_{i1}\quad x_{i2}]^T$ ، به صورت زیر در نظر گرفته شده اند

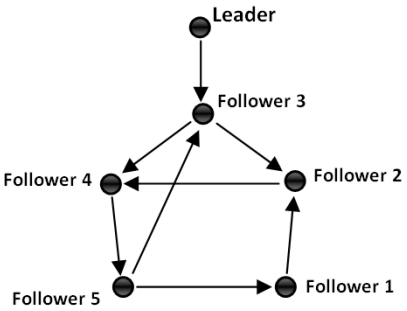
$$\dot{x}_i=f_i(x_i)+g_i(x_i)u_i\quad(۴۸)$$

$$f_i(x_i)=\begin{pmatrix}x_{i2}\\-x_{i1}+\varepsilon(1-x_{i1}^2)x_{i2}\end{pmatrix}\quad(۴۹)$$

$$\begin{aligned}&g_1(x_1)=\begin{bmatrix}0\\-0.8x_{11}x_{12}\end{bmatrix},g_2(x_2)=\begin{bmatrix}0\\x_{21}x_{22}\end{bmatrix},\\&g_3(x_3)=\begin{bmatrix}0\\0.5x_{31}x_{32}\end{bmatrix},g_4(x_4)=\begin{bmatrix}0\\-0.2x_{41}x_{42}\end{bmatrix}\\&g_5(x_5)=\begin{bmatrix}0\\1.4x_{51}x_{52}\end{bmatrix}\end{aligned}\quad(۵۰)$$

که $\varepsilon=0.5$ می باشد و دینامیک عامل رهبر به صورت زیر می باشد

$$f(x_0)=\begin{pmatrix}x_{02}\\-x_{01}+\varepsilon(1-x_{01}^2)x_{02}\end{pmatrix}\quad(۵۱)$$



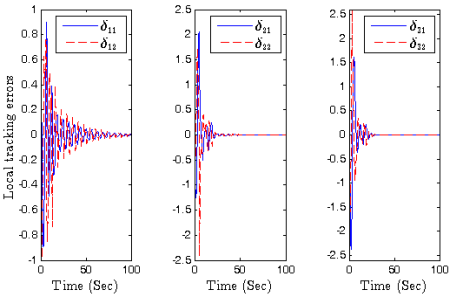
شکل ۱. گراف ارتباطی

برای توابع هزینه عامل ها داریم $Q_i(\delta_i)=\delta_i^T\delta_i$ ، $R_{ii}=10,$ $R_{ij}=1,(i\ne j,j\in N_i^I)$ برای $i=1,...,5$. برای شبکه های عصبی نقاد عامل ها داریم $\alpha_i=1$ برای $i=1,...,5$ و ماتریس های طراحی برابر با $F_{1i}=0.1I_5,\quad F_{2i}=[F_{11},F_{12},F_{13},F_{14},F_{15}]$ برای $i=1,...,5$ و پارامتر های طراحی $\lambda_1=0.7,\lambda_2=10,$ $\lambda_3=10,\lambda_4=0.46,\lambda_5=12$ انتخاب شده اند. توابع فعالیت هر یک از عامل ها به صورت زیر در نظر گرفته شده اند:

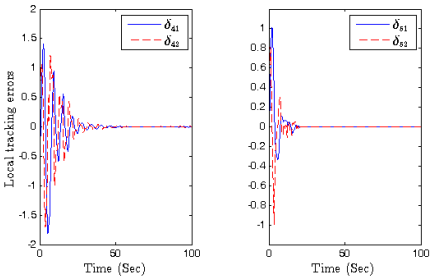
$$\begin{aligned}\sigma_i=[&\delta_{i1}^2,\delta_{i1}\delta_{i2},\delta_{i2}^2,\delta_{i1}^3,\delta_{i1}^2\delta_{i2},\delta_{i1}\delta_{i2}^2\\&,\delta_{i2}^3,\delta_{i1}^4,\delta_{i1}^3\delta_{i2},\delta_{i1}^2\delta_{i2}^2,\delta_{i1}\delta_{i2}^3,\delta_{i2}^4]\end{aligned}\quad(۵۲)$$

برای نشان دادن عدم نیاز به سیاست های کنترلی اولیه پایدارساز در الگوریتم پیشنهادی، همه وزن های اولیه شبکه های عصبی نقاد برابر با صفر انتخاب می شوند. در ۵۰ ثانیه ابتدایی شبیه سازی، یک نویز نمایی کاهشی برای تضمین شرط تحریک پایا به ورودی های کنترلی اضافه شده است. شکل های ۲ و ۳ همگرایی خطای ردیابی محلی (۳) هر عامل

به صفر را نشان می دهد. همگرایی کلیه خطاهای ردیابی به صفر، همگرایی حالت های عامل ها به رهبر را نشان می دهد.

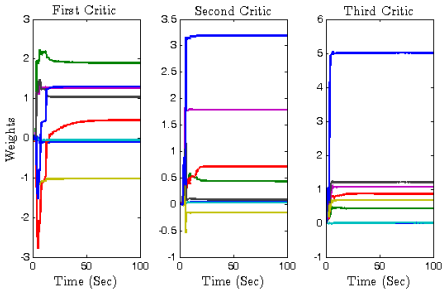


شکل ۲. همگرایی خطای ردیابی محلی عامل های ۱،۲، ۳

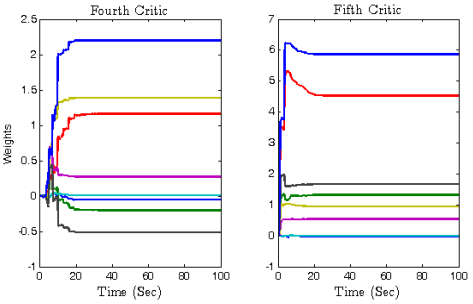


شکل ۳. همگرایی خطای ردیابی محلی عامل های ۴ و ۵

شکل های ۴ و ۵ همگرایی وزن های شبکه های عصبی نقاد عامل ها را به وزن های زیر بهینه نشان می دهد. شایان ذکر است، همانطور که از شکل های ۴ و ۵ دیده می شود، همه وزن های اولیه شبکه های عصبی نقاد برابر با صفر انتخاب شده اند، که عدم نیاز به سیاست های کنترلی اولیه پایدارساز در الگوریتم پیشنهادی را نشان می دهد.

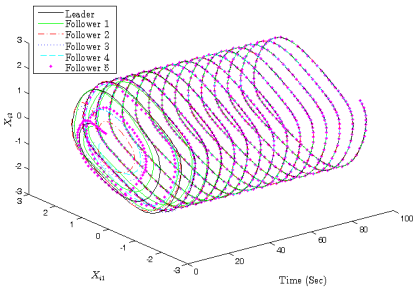


شکل ۴. همگرایی وزن های شبکه های عصبی نقاد برای عامل های ۱، ۲، ۳



شکل ۵. همگرایی وزن های شبکه های عصبی نقاد برای عامل های ۴ و ۵

صفحه فاز سه بعدی نمایش داده شده در شکل ۶، نشان دهنده همگرایی حالت همه عامل ها به حالت رهبر است. همانطور که از شکل ۶ دیده می شود، حالت های تمامی عامل ها با شرایط اولیه متفاوت نهایتا با همگرایی وزن های شبکه عصبی نقاد به وزن های زیر بهینه، به حالت های رهبر همگرا شده و در نتیجه خطای ردیابی محلی تمامی عامل ها به صفر همگرا می شود.



شکل ۶. صفحه فاز همگرایی حالت عامل ها به حالت رهبر

نتایج شبیه سازی نشان می دهد که الگوریتم پیشنهادی ضمن حفظ پایداری حلقه بسته تمامی عامل ها، به حل نش تقریبی بازی گرافی دیفرانسیلی مذکور همگرا شده است.

همانطور که پیشتر ادعا شد، الگوریتم پیشنهادی در این مقاله بار محاسباتی کمتری در مقایسه با روش [۳۵] دارد. برای توجیه این ادعا، الگوریتم پیشنهادی در [۳۵] و الگوریتم پیشنهادی در این مقاله به سیستم چند عاملی (۴۸)-(۵۰) با گراف ارتباطی نشان داده شده در شکل ۱ اعمال می شوند. شرایط اولیه حالت ها ی بازیکنان بصورت یکسان انتخاب می شوند. توابع فعالیت شبکه عصبی نقاد هر یک از عامل ها بصورت (۵۲) انتخاب می گردند. در الگوریتم پیشنهادی در [۳۵] توابع فعالیت شبکه عصبی عملگر بصورت $\sigma_i^{Actor} = \nabla \sigma_i$ برای $i = 1,...,5$ انتخاب می شوند. در هر دو روش ها، $(i \neq j, j \in N_i^I)$ ، $R_{ij} = 1$ ، $R_{ii} = 10$ ، $Q_i(\delta_i) = \delta_i^T \delta_i$ برای $i = 1,...,5$. در الگوریتم پیشنهادی در [۳۵] بهره های تنظیم همگی برابر با یک انتخاب می شوند. در الگوریتم پیشنهادی در این مقاله، $\alpha_i = 1$ برای $i = 1,...,5$ و ماتریس های طراحی برابر با $F_{2i} = [F_{11}, F_{12}, F_{13}, F_{14}, F_{15}]$ ، $F_{1i} = 0.1 I_5$ برای $i = 1,...,5$ و پارامتر های طراحی $\lambda_1 = 0.7$ ، $\lambda_2 = 10$ ، $\lambda_3 = 10$ ، $\lambda_4 = 0.46$ ، $\lambda_5 = 12$ انتخاب شده اند.

برای مقایسه عملکرد ها، توابع ارزیابی بشکل زیر تعریف می شوند.

$$P(i) = \sum_{i=1}^{N_S} \{ \|\delta_i(K)\| + R_{ii} \|\hat{u}_i(K)\| + \sum_{j \in N_i} R_{ij} \|\hat{u}_j(K)\| \} \quad (۵۳)$$

برای $i = 1,...,5$ ، که N_S تعداد نمونه ها است.

جدول ۱- مقایسه بین روش پیشنهادی در این مقاله و روش پیشنهادی در [۳۵]

	روش پیشنهادی	روش [۳۵]
$P(1)$	۴۷۶,۱۶	۵۶۱,۵۸
$P(2)$	۷۷۱,۴۴	۸۵۲,۷۴
$P(3)$	۸۰۵,۸۴	۸۹۱,۵۴
$P(4)$	۵۰۹,۷۲	۵۱۲,۷۰
$P(5)$	۴۸۳,۸۶	۵۰۶,۴۶
زمان(ثانیه)	۱۴,۷۲	۱۶,۳۵

جدول ۱، مقایسه ای بین روش پیشنهادی در این مقاله و روش پیشنهادی در [۳۵] را با توجه به توابع ارزیابی (۵۳) و مقدار زمانی که هر یک از این روش ها طول می کشند، ارایه می نماید. همانطور که از جدول ۱ دیده می شود، در شرایط یکسان، روش پیشنهادی در این مقاله به دلیل این که فقط از ۵ شبکه عصبی در مقابل ۱۰ شبکه عصبی در روش پیشنهادی در [۳۵] استفاده می نماید، دارای بار محاسباتی کمتر بوده و سریعتر می باشد.

۵- نتیجه گیری

در این مقاله، الگوریتم یادگیری بر خط برای حل تقریبی بازی های گرافی دیفرانسیلی زمان پیوسته غیرخطی با استفاده از برنامه ریزی پویای تقریبی تک-شبکه برای هر یک از عامل ها، پیشنهاد داده شده است. قوانین تنظیم وزن جدید برای تضمین پایداری حلقه بسته و همگرایی به سیاست های نش بازی بدون نیاز به سیاست های اولیه پایدارساز برای شبکه های عصبی نقاد توسعه داده شده است. تئوری لیاپانوف برای اثبات پایداری سیستم حلقه بسته به کار گرفته شده است. در نهایت نتایج شبیه سازی موثر بودن الگوریتم پیشنهادی را نشان داده است.

مراجع

[1] Olfati-Saber R. and Murray R. M., 2004, "Consensus problems in networks of agents with switching topology and time-delays," IEEE Transactions on Automatic Control, vol. 49, no. 9, pp. 1520–1533.

[2] Ren W., Beard R. W. and Atkins E. M., 2005, "A survey of consensus problems in multi-agent coordination," in Proc. of the 2005 IEEE American Control Conference, pp. 1859–1864.

[3] Olfati-Saber R., Alex Fax J. and Murray R. M., 2007, "Consensus and cooperation in networked multi-agent systems," in Proc. of the IEEE 2007, vol. 95, no. 1, pp. 215–233.

[4] Qu Z., Cooperative Control of Dynamical Systems: Applications to Autonomous Vehicles. New York: Springer-Verlag, 2009.

[5] Lewis F. L., Zhang H., Hengster-Movric K. and Das A., Cooperative control of multi-agent systems: optimal and adaptive design approaches, Berlin: Spring-Verlag, 2014.

[6] Defoort M., Floquet T., Kokosy A., et al. 2008, "Sliding-mode formation control for cooperative autonomous mobile robots", IEEE Transactions on Industrial Electronics, vol. 55, no. 11, pp. 3944–3953.

[7] Lin W., 2014, "Distributed UAV formation control using differential game approach", Aerospace Science and Technology, vol. 35, pp. 54–62.

[8] Beard, R. W. and Stepanyan, V., 2003, "Synchronization of information in distributed multiple vehicle coordination control". In Proc. of the IEEE conference on decision and control, Maui, HI, pp. 2029–2034.

[9] Mu S., Chu T. and Wang L., 2005, "Coordinated collective motion in a motile particle group with a leader", Physica A, vol. 351, pp. 211–226.

[10] Nasirian V., Davoudi A., and Lewis F. L., 2014 "Distributed adaptive droop control for DC Microgrids," in Proc. 29th IEEE Applied Power Electronics Conference and Exposition, pp. 1147–1152.

[11] Rong L., Xu S. and Zhang B., 2012, "On the general second-order consensus protocol in multi-agent systems with input delays", Transactions of the Institute of Measurement and Control, vol. 34, no. 8, pp. 983–989.

[12] Xie D. and Chen J., 2013, "Consensus problem of data-sampled networked multi-agent systems with time-varying communication delays", Transactions of the Institute of Measurement and Control, vol. 35, no. 6, pp. 753–763.

[13] Zhang H., Lewis F. and Qu Z., 2012, "Lyapunov, adaptive, and optimal design techniques for cooperative systems on directed communication graphs", IEEE Transactions on Industrial Electronics, vol. 59, pp. 3026–3041.

[14] Ren W., Beard R. and Atkins E., 2007, "Information consensus in multi vehicle cooperative control", IEEE Control Systems, vol. 27, no.2, pp. 71–82.

[15] Zhuand W. and Cheng D., 2010, "Leader-following consensus of second-order agents with multiple time-varying delays". Automatica 46(12): 1994–1999.

[16] Ren W., Moore K. and Chen Y., 2007, "High-order and model reference consensus algorithms in cooperative control of multi vehicle systems", Journal of Dynamic Systems, Measurement, and Control, vol. 129, no. 5, pp. 678–688.

[17] Wang X. and Chen G., 2002, "Pinning control of scale-free dynamical networks", Physica A, vol. 310, no. 3–4, pp. 521–531.

[18] Hong Y., Hu J. and Gao L., 2006, "Tracking control for multi-agent consensus with an active

[31] Barto A.G., Sutton R.S. and Anderson C.W., 1983, "Neuronlike adaptive elements that can solve difficult learning control problems", IEEE Transactions on Systems, Man, and Cybernetics, vol. 13, pp. 834–846.

[32] Pao Y.H. and Philips S.M., 1995, "The functional link net learning optimal control", Neurocomputing vol. 9, pp. 149–164.

[33] Abu-Khalaf M. and Lewis F.L., 2005, "Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach", Automatica, vol. 41, pp. 779–791.

[34] Modares, H., Lewis, F. L., and Naghibi-Sistani, M. B., 2014, "Integral reinforcement learning and experience replay for adaptive optimal control of partially-unknown constrained-input continuous-time systems," Automatica, vol. 50, no. 1, pp. 193-202.

[35] Tatari F., Naghibi-Sistani M. B., Vamvoudakis K. G., 2015, "Distributed Learning Algorithm for Nonlinear Differential Graphical Games," in Transactions of the Institute of Measurement and Control, doi: 10.1177/0142331215603791.

[36] Zhang H., Cui L. and Luo Y., 2013, "Near-optimal control for nonzero-sum differential games of continuous-time nonlinear systems using single-network ADP", IEEE Transactions on Systems, Man, and Cybernetics, vol. 43, no. 1, pp. 206–216.

[37] Dierks, T., and Jagannathan, S., 2010, "Optimal control of affine nonlinear continuous-time systems using an online Hamilton-Jacobi-Isaacs formulation," In: Proceedings of the 49th Decision and Control Conference. Atlanta, GA: IEEE, 3048 – 3053.

[38] Lewis F. L., Vrabie D. and Syrmos V. L., Optimal Control. 3rd Edition. John Wiley, 2012.

[39] Abu-Khalaf M., and Lewis F. L., 2005, "Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach". Automatica 41: 779–791.

[40] Finlayson B.A., The Method of Weighted Residuals and Variational Principles. New York: Academic Press, 1990.

[41] Hornik K., Stinchcombe M. and White H., 1990, "Universal approximation of an unknown mapping and its derivatives using multi layer feedforward networks", Neural Networks, vol. 3, no. 5, pp. 551–560.

[42] Khalil H. K., Nonlinear System. Englewood Cliffs, NJ: Prentice-Hall, 1996.

leader and variable topology", Automatica, vol. 42, no. 7, pp. 1177–1182.

[19] Li X., Wang X. and Chen G., 2004, "Pinning a complex dynamical network to its equilibrium", IEEE Transactions on Circuits and Systems, vol. 51, no.10, pp. 2074–2087.

[20] Tang Z., 2015, "Leader-following consensus with directed switching topologies", Transactions of the Institute of Measurement and Control, vol. 37, no. 3, pp. 406-413.

[21] Xie D., Yuan D., Lu J., et al., 2013, "Consensus control of second-order leader–follower multi-agent systems with event-triggered strategy", Transactions of the Institute of Measurement and Control, vol. 35, no.4, pp. 426–436.

[22] Başar, T. and Olsder, G. J., Classics in applied mathematics, Dynamic noncooperative game theory (2nd ed.). Philadelphia: SIAM, 1999.

[23] Vamvoudakis, K. G., Lewis, F. L., and Hudas, G. R., 2012, "Multi-agent differential graphical games: online adaptive learning solution for synchronization with optimality", Automatica, vol. 48, no. 8, pp. 1598–1611.

[24] Sutton, R. S. and Barto, A. G., Reinforcement learning—an introduction. Cambridge, MA: MIT Press, 1998.

[25] Sen, S. and Weiss, G., Learning in multi-agent systems, in multi-agent systems: a modern approach to distributed artificial intelligence. (pp. 259–298). Cambridge, MA: MIT Press, 1999.

[26] Murray J.J., Cox C.J., Lendaris G.G., et al., 2002, "Adaptive dynamic programming", IEEE Transactions on Systems, Man, and Cybernetics, vol. 32, no. 2, pp. 140–153.

[27] Wei, Q. , Liu, D., and Lewis F. L., 2015, "Optimal distributed synchronization control for continuous-time heterogeneous multi-agent differential graphical games," Inform. Sci., vol. 317, pp. 96-113.

[28] Jiao, Q. , Modares, H., Xu, S., Lewis, F. L., and Vamvoudakis, K. G., 2016, "Multi-agent zero-sum differential graphical games for disturbance rejection in distributed control," Automatica, vol. 69, pp. 24-34.

[29] Abouheaf M. I. and Lewis F. L., 2013, "Multi-agent differential graphical games: Nash online adaptive learning solutions", 52nd IEEE Conference on Decision and Control, pp. 5803-5809.

[30] Abouheaf M. I., Lewis F. L. and Mahmoud M. S., 2014, "Differential graphical games: Policy iteration solutions and coupled Riccati formulation", European Control Conference, pp.1594-1599.