سیستم‌های فازی و کاربردها
سال ششم، شماره اول (بهار و تابستان ۱۴۰۲)
صص. ۱۰۹ تا ۱۲۶
شاپای چاپی: ۲۷۱۷-۴۴۰۹
شاپای الکترونیکی: ۲۷۱۷-۳۹۸۴
DOI:10.22034/JFSA.2023.383519.1159

تنظیم اولیه‌ی معماری یادگیری تقویتی فازی با استفاده از روش تکرار ارزش

فرزانه نادی، ولی درهمی* و فریناز اعلمی‌یان هرندی

دانشکده مهندسی کامپیوتر، دانشگاه یزد، یزد، ایران
دانشکده مهندسی کامپیوتر، دانشگاه یزد، یزد، ایران
دانشکده مهندسی برق و کامپیوتر، دانشگاه صنعتی اصفهان، اصفهان، ایران

تاریخ دریافت: ۱۴۰۱/۱۱/۰۸ تاریخ پذیرش: ۱۴۰۲/۰۲/۰۶
نوع مقاله: علمی-پژوهشی

چکیده

این پژوهش روشی جدید در استفاده از داده‌های تعاملی عامل و محیط برای تنظیم اولیه‌ی معماری یادگیری تقویتی فازی ارائه می‌دهد. کندی سرعت آموزش و نحوه‌ی تعیین مقدار توابع عضویت ورودی دو چالش مهم در معماری یادگیری تقویتی فازی هستند. تنظیم اولیه‌ی پارامترهای سیستم با استفاده از داده‌های تعاملی می‌تواند راهکار مناسبی برای رفع چالش‌های اشاره شده باشد. در این پژوهش ابتدا با تعامل عامل با محیط و جمع‌آوری داده آموزشی، ماتریس احتمال انتقال حالت-عمل به حالت بعدی و امید پاداش آنی حالت-عمل به حالت بعدی محاسبه می‌شود. با توجه به پیوسته بودن فضای مورد بررسی، جهت تولید دو ماتریس مذکور از خوشه‌بندی استفاده می‌شود. هر خوشه یک حالت از محیط لحاظ شده و بدین صورت یک تقریب احتمال گذر از یک خوشه به خوشه‌ی دیگر با توجه به داده‌ها تعیین می‌شود. سپس پارامترهای سیستم فازی با تعمیم روش تکرار ارزشِ برنامه‌سازی پویا برای فضای پیوسته تنظیم می‌گردد. نحوه‌ی استفاده روش پیشنهادی با یک مثال به طور کامل(ادامه دارد)

عبارات و کلمات کلیدی: سیستم فازی، یادگیری تقویتی، برنامه‌سازی پویا، خوشه‌بندی.

Email(s): farzane.nadi@gmail.com, vderhami@yazd.ac.ir and farinaz.alamiyan@gmail.com.

Mathematics Subject Classification: 47N70 ; 93C05 ; 93C85

شرح داده شده است. استفاده از این روش می‌تواند منجر به افزایش سرعت یادگیری و کمک در تنظیم توابع عضویت ورودی سیستم فازی گردد.

۱ مقدمه

یادگیری تقویتی[۱] یکی از شاخه‌های یادگیری ماشین است که در آن، عامل نحوه‌ی تعامل با محیط را با هدف بیشینه کردن پاداش دریافتی خود یاد می‌گیرد. به عبارتی عامل در هر موقعیت خاص، عمل مناسب را از بین اعمال مجاز و بر اساس جایزه و جریمه دریافتی انتخاب می‌کند [۱،۵،۶]. امروزه یادگیری تقویتی به دلیل نتایج امیدوارکننده در بسیاری از زمینه‌ها از جمله بازی‌های ویدیویی، سیستم‌های پیشنهاد دهنده، کنترل ربات، مدیریت انرژی و پردازش زبان‌های طبیعی استفاده می‌شود [۷-۱۱]. تنگنای ابعاد[۲] مشکلی شناخته شده در الگوریتم‌های یادگیری تقویتی است که یکی از راه‌های رهایی از آن، استفاده از یادگیری تقویتی پیوسته[۳] است [۱۲]. در یادگیری تقویتی پیوسته از ترکیب یک الگوریتم یادگیری تقویتی و تقریب‌زننده تابع استفاده می‌شود. از مشهورترین تقریب‌زننده‌های تابع می‌توان به سیستم‌های فازی اشاره کرد. سیستم‌های فازی به دلیل توانایی برجسته در تقریب توابع غیرخطی با درجه پیچیدگی متنوع به عنوان تقریب‌زننده‌های جامع قدرتمند شناخته می‌شوند. از مزیت دیگر سیستم‌های فازی می‌توان به مصالحه قابل قبول بین دقت نتایج و تفسیرپذیری قواعد اشاره کرد. الگوریتم‌های حاصل از ترکیب یادگیری تقویتی و این سیستم‌ها، الگوریتم‌های یادگیری تقویتی فازی[۴] نامیده می‌شوند [۲]. سه معماری نقاد-تنها[۵]، عملگر-تنها[۶] و عملگر-نقاد[۷] از معماری‌های یادگیری تقویتی فازی هستند که هر یک مزیت و عیب مختص به خود را دارند.

یکی از مهم‌ترین چالش‌ها در یادگیری تقویتی فازی، بحث تاخیر در یادگیری و یکی از راه‌های افزایش سرعت یادگیری، استفاده از داده‌های جمع‌آوری شده توسط ناظر است.

---

[1] Reinforcement Learning (RL)

[2] Curse of dimensionality

[3] Continues RL

[4] Fuzzy Reinforcement Learning (FRL)

[5] Critic Only

[6] Actor Only

[7] Actor-Critic

بدین ترتیب دسته‌ی جدیدی از الگوریتم‌ها به نام یادگیری تقویتی برون‌خط[8] پدید می‌آید. مطالعات اخیر نشان می‌دهد یادگیری تقویتی برون‌خط یک الگوی امیدوارکننده در یادگیری با استفاده از مجموعه داده‌های ثابت است و برای مسائل نیازمند به واکنش‌های بلادرنگ همچون رانندگی خودکار و هدایت ربات [۱۳،۱۴] پرکاربرد است. از دیگر مزیت‌های یادگیری تقویتی برون‌خط استفاده از آن برای محیط‌هایی است که تعامل مستقیم عامل یادگیرنده با محیط امکان‌پذیر نیست و یا هزینه‌بر است [۱۵]. به عنوان نمونه می‌توان به استفاده از یادگیری تقویتی برون‌خط برای هدایت ربات به سمت نقطه هدف [۱۶]، حرکت نرم ربات چرخ‌دار در زمین‌های ناهموار بیرونی [۱۷] و هدایت بازوی ربات برای انجام وظایفی از جمله عمل برداشتن و گذاشتن[9] [۱۸] اشاره کرد.

هدف در این مقاله، ارائه روشی نوین برای مقداردهی ارزش‌های اولیه عمل‌ها و تنظیم توابع عضویت ورودی در الگوریتم یادگیری تقویتی فازی با استفاده از داده‌های جمع‌آوری شده است. قصد داریم با داده‌های جمع‌آوری شده‌ی حاصل از تعامل عامل با محیط، ماتریس احتمال انتقال و امید پاداش آنی حالت-عمل به حالت بعدی را محاسبه نماییم. سپس با تعمیم روش تکرار ارزش[10] در برنامه‌سازی پویا[11] به حالت پیوسته، از این ماتریس‌ها بهره برده و سیستم فازی سوگنوی مرتبه صفر[12] موردنظر را تنظیم اولیه نماییم.
بنابراین سهم علمی پژوهش به شرح زیر است:

۱- استفاده از خوشه‌ها در فضای پیوسته جهت تعمیم ایده‌های گسسته در این فضا

۲- گسترش روش تکرار ارزش برنامه‌سازی پویا در محیط پیوسته با استفاده از سیستم فازی

ساختار مقاله بدین گونه است: در بخش دوم مفاهیم پایه مرور می‌شوند. روش پیشنهادی در بخش سوم معرفی و در بخش چهارم در قالب یک مثال شرح داده می‌شود. نتیجه‌گیری در بخش پنجم بیان می‌گردد.

[8] Offline RL
[9] Pick and place
[10] Value Iteration
[11] Dynamic Programming (DP)
[12] Zero Order TSK

۲ مفاهیم پایه

در این بخش مفاهیم پایه سیستم فازی، یادگیری تقویتی فازی، برنامه‌سازی پویا، و خوشه‌بندی به اختصار بیان می‌شود. در زیربخش یادگیری تقویتی فازی، یادگیری سارسای فازی[۱۳][۱۹] به عنوان مشهورترین این سیستم‌ها توضیح داده می‌شود.

۱.۲ سیستم فازی

سیستم‌های فازی ساختاری شفاف(جعبه سفید[۱۴]) برای استنتاج ارائه می‌دهند و شبیه سازی منطق و طرز فکر انسانی را در یک پیاده‌سازی انعطاف‌پذیر فراهم می‌کنند. این سیستم ها در کاربردهای مختلف به عنوان کنترلگرهای فازی، پایگاه قواعد فازی و مدل های فازی استفاده می‌شوند[۴،۳].
قواعد در سیستم های فازی با استفاده از روابط اگر-آنگاه[۱۵]و در قالب مقدم و تالی تعبیه می شوند. اولین ساختار کاربردی سیستم‌های استنتاج فازی توسط ممدانی معرفی شد که دربخش تالی قواعد همانند بخش مقدم از برچسب‌های فازی استفاده می‌شود.این ساختار، مدل ممدانی[۱۶] در سیستم‌های فازی است. پس از آن، دسته دیگری از سیستم‌های فازی با نام سیستم فازی سوگنو معرفی شد که در بخش مقدم قواعد فازی از بر چسب ها فازی استفاده شده است. اما تالی قواعد فازی در این نوع سیستم‌های فازی، چند جمله‌ای واضح از متغیرهای ورودی است.
سیستم‌های فازی از پنج بخش اصلی تشکیل شده‌اند [۲]:
پایگاه قواعد: حاوی قواعد فازی به فرمت اگر-آنگاه
پایگاه داده فازی: حاوی مشخصات توابع عضویت مجموعه‌های فازی
موتور استنتاج فازی: تصمیم‌گیر بر اساس قواعد فازی
فازی‌ساز[۱۷]: تبدیل‌کننده ورودی غیرفازی به مجموعه فازی
فازی‌زدا[۱۸]: تبدیل‌کننده خروجی فازی به مقدار خروجی قطعی

[13] Fuzzy Sarsa Learning (FSL)
[17] Fuzzifier
[18] Defuzzifier

۲.۲ یادگیری تقویتی فازی

ایده‌ی زیربنای تمام نظریه‌های یادگیری و هوشمندسازی، یادگیری از طریق تعامل با محیط است. یادگیری تقویتی، به معنای یادگیری انتخاب عمل مناسب در هر موقعیت بر اساس پاداش و جریمه‌های دریافتی است. به بیانی دیگر، چالش اصلی در یادگیری تقویتی، نحوه نگاشت موقعیت‌ها به عمل‌ها است، به نحوی که سیگنال پاداش عددی (سیگنال تقویتی) بیشینه شود [۲۰،۱]. نکته مهم این است که در این روش برخلاف اکثر روش‌های یادگیری ماشین، به یادگیرنده گفته نمی‌شود که در یک موقعیت خاص چه عملی را باید انجام دهد، بلکه یادگیرنده باید "خودش" با امتحان کردن عمل‌های مختلف در آن موقعیت، دریابد امید دریافت پاداش کدام یک از عمل‌هایش برای آن موقعیت بیشتر است. در یک سیستم یادگیری تقویتی، شش عنصر اصلی قابل شناسایی هستند. این عناصر عبارتند از: عامل، حالت محیط، سیاست، تابع پاداش، تابع ارزش و مدلی از محیط [۲۰،۱].

با ترکیب روش‌های یادگیری تقویتی و سیستم‌های فازی به عنوان تقریب زننده‌های تابع، سیستم‌های یادگیری تقویتی فازی ارائه شدند. معماری‌های یادگیری تقویتی فازی در نگاهی گسترده، به سه دسته معماری نقاد-تنها، عملگر-تنها و عملگر-نقاد تقسیم می‌شوند. هر یک از این معماری‌ها مزایا و معایب خاص خود را دارند. اما معماری نقاد-تنها به دلیل درجه کاوش بالاتر و شفافیت بیشتر در ارائه‌ی دانش خبره کاربرد وسیع‌تری دارد.

در معماری نقاد-تنهای یادگیری تقویتی با استفاده از تقریب‌زننده‌های تابع، نگاشت میان فضای حالت و فضای عمل تقریب زده می‌شود. دو روش پایه‌ای در این معماری، یادگیری کیوی فازی[19] و یادگیری سارسای فازی نامیده شده‌اند. این روش‌ها بر اساس مدل فازی سوگنوی مرتبه صفر هستند و راهکاری برای تنظیم برخط تالی قواعد ارائه می‌دهند [۲].

یادگیری سارسای فازی بر خلاف یادگیری کیوی فازی که برون سیاست است، یک الگوریتم بر سیاست می‌باشد و ساختار قواعد آن با استفاده از سیستم فازی سوگنوی مرتبه صفر به صورت زیر است:

(۱) $\text{Rule}_i : if\ x_1 \text{ is } L_{i1}\ and \ldots and\ x_n \text{ is } L_{\text{in}},$

$$then\ (o_{i1} \text{ with value } w^{i1} or \ldots or\ (o_{im} \text{ with value } w^{im})),\ i = 1, 2, \ldots, R$$

که در آن R، x_k و n به ترتیب بیان‌کننده تعداد قواعد، ورودی kام، و تعداد متغیرهای

[19] Fuzzy Q-Learning (FQL)

ورودی است. $L_i = L_{i1} \times \ldots \times L_{\text{in}}$ شامل n مجموعه فازی محدب اکیدا نرمال با مرکزهای یکتا برای iامین قاعده است. در این رابطه، m تعداد عمل‌های گسسته ممکن برای هر قاعده، o_{ij} jامین عمل نامزد در قاعده iام و w^{ij} مقدار ارزش تقریب زده شده‌ی آن است. مقدار تالی هر قاعده در هر قدم زمانی با توجه به مقادیر w^{ij} انتخاب می‌شود [۱، ۲۰].

۳.۲ برنامه‌سازی پویا

در مبحث یادگیری تقویتی، اصطلاح برنامه‌سازی پویا به مجموعه‌ای از الگوریتم‌ها اتلاق می‌شود که می‌توانند برای محاسبه‌ی مقادیر ارزش و سیاست‌های بهینه در یک مسئله استفاده شوند. الگوریتم‌های برنامه‌سازی پویا به یک مدل کامل از محیط نیاز دارند. مدل محیط شامل مجموعه‌های حالت محیط ($\mathcal{S}$) و عمل ($A(s),\ s \in \mathcal{S}$) است و پویایی آن‌ها توسط مجموعه‌ای از احتمالات انتقال حالت-عمل به حالت بعدی ($P^a_{ss'}$) و امید پاداش‌های آنی حالت-عمل به حالت بعدی ($R^a_{ss'}$) به صورت زیر تعیین شده است:

$$P^a_{ss'} = \Pr\left\{ s_{t+1} = s' \,\middle|\, s_t = s,\ a_t = a \right\}$$
$$R^a_{ss'} = \mathrm{E}\left\{ r_{t+1} \middle|\, s_t = s,\ a_t = a,\ s_{t+1} = s' \right\} \qquad (۲)$$

این رابطه برای تمام $s \in \mathcal{S}$، $a \in A(s)$ و $s' \in \mathcal{S}^+$ ($\mathcal{S}^+$ همان مجموعه $\mathcal{S}$ است که حالت پایانی را نیز شامل می‌شود) برقرار است [۱، ۲۰].

ایده کلیدی یادگیری تقویتی، استفاده از توابع ارزش برای پیدا کردن سیاست‌های مناسب است. یکی از راه‌های یافتن توابع ارزش بهینه برای تمام حالت‌های محیط استفاده از روش تکرار ارزش در برنامه‌سازی پویا است:

$$V^*(s) = \max_a \sum_{s'} P^a_{ss'} [R^a_{ss'} + \gamma V^*(s')] \qquad (۳)$$

برای تمام $s \in \mathcal{S}$، $a \in A(s)$ و $s' \in \mathcal{S}^+$. در رابطه فوق، $0 \leq \gamma \leq 1$ فاکتور تخفیف است [۱].

۴.۲ خوشه‌بندی

الگوریتم خوشه‌بندی را می‌توان یکی از مهمترین الگوریتم‌های یادگیری بدون ناظر در نظر گرفت. هدف از خوشه‌بندی برای یک مجموعه داده معین تقسیم داده‌ها به چند گروه است به طوری که داده‌های یک گروه تا حد امکان مشابه باشند درحالی‌که داده‌ها در گروه‌های مختلف حداکثر میزان تفاوت را دارا باشند [۲۱]. خوشه‌بندی در بسیاری از زمینه‌ها از جمله یادگیری ماشین، تشخیص الگو، تجزیه و تحلیل تصویر، بازیابی اطلاعات، بیوانفورماتیک، فشرده‌سازی داده‌ها، گرافیک کامپیوتری، صفحات وب و رباتیک استفاده می‌شود [۲۲-۲۴].

خوشه‌بندی سخت[۲۰] و نرم[۲۱] دو گروه اصلی در رویکردهای متنوع خوشه‌بندی هستند. در خوشه‌بندی نرم یا فازی هر داده می‌تواند به بیش از یک خوشه تعلق داشته باشد، اما در خوشه‌بندی سخت، هر نقطه داده تنها می‌تواند به یک خوشه تعلق داشته باشد. به طور کلی، خوشه‌بندی سخت را می‌توان به خوشه‌بندی سلسله مراتبی[۲۲] و خوشه‌بندی تفکیکی[۲۳] طبقه‌بندی کرد. در الگوریتم‌های نوع اول، نقاط داده به ساختارهای درختی سلسله مراتبی دسته‌بندی می‌شوند. خوشه‌بندی تفکیکی یک مجموعه داده را بر اساس شباهت آن‌ها از طریق یک فرآیند تکراری به چند گروه تقسیم می‌کند. دو دسته اصلی از روش‌های تفکیکی، روش‌های مبتنی بر فاصله[۲۴] و روش‌های مبتنی بر چگالی[۲۵] هستند که به ترتیب بر اساس فاصله بین داده‌ها و توزیع چگالی داده‌ها، خوشه‌ها را تشکیل می‌دهند. خوشه‌بندی $Kmeans$ از شناخته شده‌ترین الگوریتم‌های خوشه‌بندی تفکیکی است [۲۵] که در این مقاله از آن بهره برده شده است.

[20] Hard clustering
[21] Soft clustering
[22] Hierarchical clustering
[23] Partitional clustering
[24] Distance-based clustering
[25] Density-based clustering

۳ روش پیشنهادی

رویکرد جدیدی که در این پژوهش دنبال شده است بدست آوردن مدلی از محیط با استفاده از داده‌های جمع‌آوری شده است تا بتوان از این مدل برای تنظیم اولیه‌ی سیستم فازی بهره جوییم. یکی از روش‌های مناسب در این حوزه، برنامه‌سازی پویا است که در آن به دو ماتریس احتمال انتقال حالت-عمل به حالت بعدی و امید پاداش آنی حالت-عمل به حالت بعدی (رابطه‌ی ۲) نیاز داریم. لذا باید با استفاده از داده‌های جمع‌آوری شده این دو ماتریس را محاسبه نماییم. با توجه به پیوسته بودن محیط، روش جدیدی به نام تکرار ارزش فازی[۲۶] برای انجام این روند در محیط‌های پیوسته ارائه داده‌ایم تا بتوان برای تنظیم اولیه‌ی سیستم فازی در روش یادگیری تقویتی فازی استفاده نمود.

فرض کنید تعدادn داده شامل جفت حالت و عمل در محیط موردنظر جمع‌آوری شده است. هر حالت شاملN_f متغیر حالت پیوسته و تعداد عمل‌های مجاز در هر حالتp عمل است. یکی از راه‌های معمول برای تعیین فضای حالت با استفاده از داده‌های جمع‌آوری شده، استفاده از الگوریتم‌های خوشه‌بندی است [۱۷]. در گام اول، داده‌ها با استفاده از الگوریتم $Kmeans$ به تعدادی خوشه تقسیم می‌شود. الگوریتم $Kmeans$ داده‌ها را به خوشه‌هایی تقسیم می‌کند که داده‌های موجود در هر خوشه بیشترین شباهت را با هم و با داده‌های سایر خوشه‌ها بیشترین تفاوت را داشته باشند. این مهم با استفاده از روش تکراری و به منظور حداقل نمودن رابطه زیر انجام می‌شود:

$$F = \sum_{k=1}^{N_c} \sum_{j=1}^{N_k} \sqrt{\sum_{i=1}^{N_f} (C_{\mathrm{ki}} - P_{\mathrm{ji}})^2} \qquad (۴)$$

که N_c، N_k، N_f، C_k و P_j به ترتیب نشان‌دهنده‌ی تعداد خوشه‌ها، تعداد اعضای خوشه‌ی مورد بررسی، تعداد ویژگی‌های هر داده، مرکز خوشه‌ی مورد بررسی و داده‌ی مورد

[26] Fuzzy Value Iteration

بررسی هستند.

پس از تعیین خوشه‌ها معیار مهم دیگری به نام احتمال انتقال برای تمام خوشه‌ها بررسی می‌شود. این بررسی، با هدف حداکثر نمودن شباهت داده‌های موجود در یک خوشه (حالت سیستم) از نظر خوشه‌های بعد از اِعمال عمل‌ها صورت می‌گیرد. شکل ۱ نشان‌دهنده‌ی شبه کد روش پیشنهادی می‌باشد:

```
Inputs: Data and its assigned cluster
Initialize a flag for all clusters and set them to 0
For each cluster c with flag == 0:
  Generate a transaction matrix for all data in cluster c
  Calculate the difference percentage in the transaction matrix (Diff)
  while Diff >= 10%:
    Change cluster samples step by step using transaction matrix
    Set flag of all clusters to 0
  Set flag of cluster c to 1
Outputs: Data and its new assigned cluster
```

شکل ۱: *شبه‌کد روش پیشنهادی*

ورودی روش پیشنهادی، خوشه‌های تعیین‌شده توسط الگوریتم خوشه‌بندی $Kmeans$ می‌باشد و در ادامه، سه بلوک مهم از روش پیشنهادی توضیح داده شده است:

۱- تولید ماتریس انتقال با استفاده از نمونه داده‌های خوشه

در این گام، برای تمام خوشه‌هایی که قبلا بررسی نشده‌اند (متغیر پرچم[۲۷] آن خوشه صفر است)، ماتریس انتقال تولید می‌شود. رابطه‌ی ۵ نمایی کلی از ماتریس انتقال برای یک خوشه‌ی فرضی را نشان می‌دهد. سطرها در ماتریس انتقال، متناظر داده‌های خوشه‌ی مورد بررسی است (d_1 تا d_{N_k}) و هر ستون نشان‌دهنده‌ی عمل‌های مجاز (a_1 تا a_p) می‌باشد.

[27] Flag

$$
\begin{array}{c} \\ d_1 \\ \vdots \\ d_{N_k} \end{array}
\begin{array}{c}
\begin{array}{ccc} a_1 & \dots & a_p \end{array} \\
\begin{bmatrix} ? & \dots & \square \\ \square & \ddots & \square \\ \square & \square & \square \end{bmatrix}
\end{array}
\qquad (۵)
$$

برای تکمیل ماتریس انتقال از داده‌های جمع‌آوری شده استفاده می‌شود. برای مثال اگر سیستم در وضعیت داده‌ای d_1 با اِعمال عمل a_1 به وضعیت داده‌ای d'_3 منتقل شود و داده‌ی d'_3 متعلق به خوشه‌ی شماره ۲ (C_2) باشد، مقدار ؟ در ماتریس فوق، C_2 خواهد شد. برای تعیین خوشه‌ی بعدی پس از اِعمال هر عمل بر روی داده‌های یک خوشه با کمک ماتریس انتقال، از اکثریت مقادیر ستون متناظر با آن عمل استفاده می‌شود. بدین معنی که اگر در ماتریس انتقال بالا، اکثریت مقادیر ستون اول C_2 باشد، خوشه‌ی بعدی خوشه‌ی مورد بررسی پس از اِعمال عمل a_1، خوشه‌ی C_2 خواهد بود (بدین معنی که احتمال گذر به خوشه‌ی C_2 از سایر خوشه‌ها بیشتر می‌باشد).

۲- محاسبه درصد تفاوت خوشه

برای محاسبه‌ی درصد تفاوت در هر خوشه کافی است اکثریت مقادیر هر ستون مشخص و سایر مقادیر به عنوان تفاوت در نظر گرفته شود. درصد کل برای هر خوشه با توجه به تعداد کل تفاوت‌ها و تعداد کل سلول‌های ماتریس انتقال متناظر با آن خوشه، طبق فرمول زیر محاسبه می‌شود:

$$Diff\_precent. = \frac{\sum_{\text{each col.}} (Num\ of\ all\ cells) - (Num\ of\ max\ occurr.\ val.)}{(Num\ of\ all\ cells)} \qquad (۶)$$

برای مثال در یک مسئله با ۵ عمل مجاز و ۱۰ خوشه، ماتریس انتقال یک خوشه‌ی فرضی به صورت زیر است:

$$\begin{bmatrix} C_۱ & C_۴ & C_۳ & C_۷ & C_۵ \\ C_۱ & C_۴ & C_۳ & C_۷ & C_۵ \\ C_۱ & C_۴ & C_۳ & C_۷ & C_۵ \\ C_۱ & C_۳ & C_۳ & C_۱ & C_۹ \\ C_۱ & C_۴ & C_۳ & C_۷ & C_۵ \\ C_۲ & C_۲ & C_۱ & C_۷ & C_۹ \end{bmatrix}$$

در این خوشه ۶ نمونه داده (سطرهای ماتریس فوق) وجود دارد. در ستون اول اکثریت مقادیر $C_۱$ می‌باشد بنابراین در این خوشه، خوشه‌ی بعدی پس از اِعمال عمل شماره‌ی ۱، خوشه‌ی $C_۱$ است. خوشه‌های بعدی پس از اِعمال سایر عمل‌های مجاز به ترتیب $C_۴$، $C_۳$، $C_۷$ و $C_۵$ هستند. درصد تفاوت محاسبه شده برای ماتریس انتقال فوق طبق رابطه‌ی ۶ به صورت زیر محاسبه می‌شود:

$$Diff\_precent. = \frac{(۶-۵) + (۶-۴) + (۶-۵) + (۶-۵) + (۶-۴)}{۳۰} \approx ۲۳ \quad (۷)$$

۳- جابجایی نمونه داده‌های خوشه

اگر در خوشه مورد بررسی، تفاوت بیش از ۱۰ درصد باشد، گام‌های زیر بر روی سطرهای متفاوت (هر نمونه داده در خوشه) به ترتیب اِعمال می‌شود:

۱- در مورد سطرهایی که تمام خوشه‌های بعدی متناظر با هر عمل شبیه خوشه‌ی دیگری باشد، نقطه‌ی داده متناظر با آن سطرها به خوشه‌ی مشابه تعلق می‌گیرد.

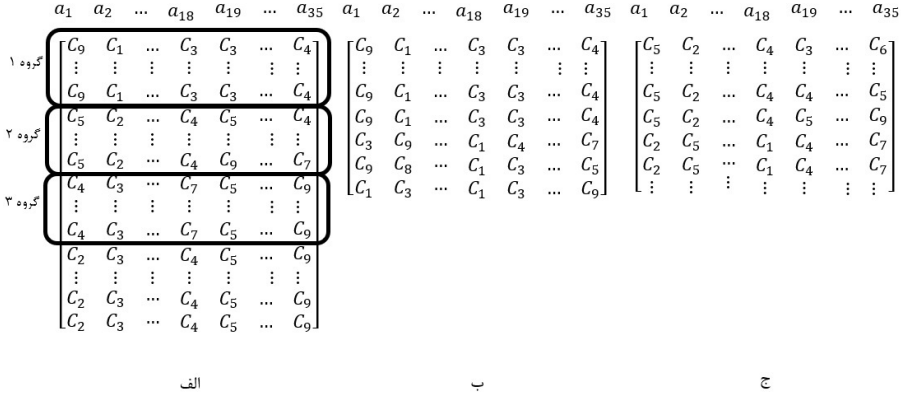
۲- نقاط داده‌ی متناظر با سطرهایی که خوشه‌های بعدی پس از اِعمال بیش از ۵۰ درصد عمل‌ها با خوشه‌های بعدی خوشه‌ی مورد بررسی متفاوت است، از خوشه‌ی فعلی جدا خواهد شد و به خوشه با شباهت بیش از ۵۰ درصد تمام عمل‌ها منتقل می‌شود.

۳- تمام سطرهایی که خوشه‌های بعدی آن‌ها با خوشه‌های بعدی خوشه‌ی مورد بررسی یکسان نیست اما شبیه یکدیگر است، در خوشه جدیدی قرار می‌گیرد.

مثال زیر گام‌های بالا را در مسئله‌ی واقعیِ مورد بررسی نشان می‌دهد:

تعداد ۱۵۵۹۷ داده شامل جفت حالت و عمل در محیط جمع‌آوری شده است و تعداد عمل‌های مجاز در هر حالت ۳۵ عمل است. داده‌ها به ۹ خوشه ($C_۱$ تا $C_۹$) تقسیم شده‌اند و تغییرات در خوشه‌ی $C_۱$ طبق گام‌های فوق شرح داده شده است. داده‌های گروه ۱، طبق

گام اول از خوشه‌ی C_1 جدا و به خوشه‌ی C_2 افزوده می‌شوند. خوشه‌ی بعدی پس از اِعمال ۱۸ عمل از ۳۵ عمل ممکن (بیش از ۵۰ درصد از عمل‌ها) برای داده‌های گروه ۲ با خوشه‌ی C_3 یکسان است، بنابراین با اجرای گام دوم الگوریتم، داده‌های گروه ۲ از خوشه‌ی C_1 جدا و به خوشه C_3 اضافه می‌شوند. گام سوم از الگوریتم هم باعث ایجاد خوشه‌ی جدید C_{10} با استفاده از داده‌های گروه ۳ می‌شود چون خوشه‌ی بعدی پس از اِعمال بیش از ۵۰ درصد عمل‌ها با هیچ خوشه‌ی دیگری شبیه نیست.



شکل ۲: *مثالی از سه گام روش پیشنهادی*

الف: ماتریس انتقال خوشه‌ی C_1 ب: ماتریس انتقال خوشه‌ی C_2 ج: ماتریس انتقال خوشه‌ی C_3

گام‌های بالا تا زمانیکه درصد تفاوت در خوشه‌ی مورد بررسی بیشتر از ۱۰ باشد، به ترتیب اجرا می‌شود. اگر هر یک از گام‌های بالا باعث ایجاد تفاوت در داده‌های یک خوشه شود، متغیر پرچم برای تمام خوشه‌ها (بجز خوشه مورد بررسی) صفر خواهد شد تا مجدد همه‌ی خوشه‌ها بررسی شود. در نهایت به متغیر پرچم متناظر با خوشه‌ی مورد بررسی مقدار یک داده می‌شود.

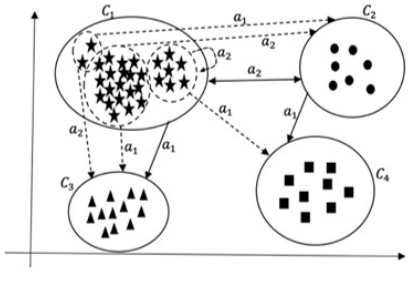
خروجی روش پیشنهادی، خوشه‌های بهبودیافته است که به عنوان حالات سیستم در نظر گرفته می‌شود. خوشه‌های ورودی و خوشه‌های خروجی در روش پیشنهادیِ، به ترتیب حالت‌های سیستم قبل و بعد از اجرای الگوریتم هستند. با بررسی ماتریس احتمال انتقال حالت-عمل به حالت بعدی برای تمام حالت‌ها و عمل‌ها برای خوشه‌های ورودی و خروجی الگوریتم، مشخص شد ماکزیمم احتمال انتقال حالت-عمل به حالت بعدی در تمام حالت-عمل‌ها بهبود قابل ملاحظه‌ای دارد.

در نهایت با استفاده از خوشه‌های خروجی روش پیشنهادی می‌توان نواحی قواعد فازی[۲۸] را مشخص نمود. سپس می‌توان با استفاده از حالت‌های سیستم و روش تکرار ارزش، مقادیر اولیه‌ی ارزش عمل‌ها در تالی قواعد فازی کنترلگر را با استفاده از رابطه ۳ تعیین نمود. در نهایت قواعد فازی کنترلگر همانند رابطه‌ی ۱ است که در آن متغیرهای n و m به ترتیب مقدار ۴ و ۳۵ می‌گیرند.

۴ ارائه یک مثال

در این بخش مثالی از روش تکرار ارزش پیوسته با هدف محاسبه‌ی مقادیر ارزش تالی قواعد فازی کنترلگر بیان می‌شود.

شکل زیر نشان‌دهنده‌ی خوشه‌های حاصل از اجرای الگوریتم خوشه‌بندی بر داده‌های جمع‌آوری شده است:



شکل ۳: *خوشه‌های حاصل از اجرای الگوریتم خوشه‌بندی*

که داده‌ها به چهار خوشه (c_1، c_2، c_3 و c_4) تقسیم شده‌اند. با در نظر گرفتن دو عمل مجاز در این مثال (a_1 و a_2)، ماتریس انتقال برای خوشه‌ی شماره ۱ با داشتن ۲۸ نمونه داده، به صورت زیر می‌باشد (اعداد کنار ماتریس تعداد نمونه داده‌های هر وضعیت را نشان می‌دهد):

[28] Fuzzy Rule Patch

$$
\begin{array}{cc}
a_۱ & a_۲ \\
\left[\begin{array}{cc}
c_۳ & c_۲ \\
\vdots & \vdots \\
c_۳ & c_۲ \\
c_۴ & c_۱ \\
\vdots & \vdots \\
c_۴ & c_۱ \\
c_۲ & c_۳ \\
c_۲ & c_۳
\end{array}\right. &
\begin{array}{l}
\left.\begin{array}{l} \\ \\ \\ \end{array}\right\} ۱۹ \\
\\
\left.\begin{array}{l} \\ \\ \\ \\ \end{array}\right\} ۷
\end{array}
\end{array}
\qquad (۸)
$$

طبق ماتریس انتقال فوق، بیشترین نمونه‌های موجود در این خوشه (۱۹ نمونه)، پس از اِعمال عمل $a_۱$ و $a_۲$ به ترتیب به خوشه‌ی شماره ۳ و ۲ منتقل می‌شوند. درصد تفاوت در این خوشه ۳۲،۱ می‌باشد بنابراین روش پیشنهادی سعی می‌کند با جابجایی نمونه داده‌های این خوشه، درصد تفاوت نمونه‌های آن را از نظر خوشه بعدی پس از اِعمال عمل‌ها کاهش دهد. ابتدا ۷ نمونه داده‌ی متفاوت با اکثریت نمونه داده‌ها را بررسی می‌کند. خوشه‌های بعدی پس از اِعمال هر دو عمل مجاز، شبیه اکثریت نمونه داده‌های موجود در خوشه شماره‌ی ۲ می‌باشد (طبق ماتریس انتقال و شکل شماره ۳)، پس تمام ۷ نمونه داده‌ی متفاوت را به خوشه‌ی شماره‌ی ۲ منتقل می‌کند. با این انتقال، درصد تفاوت در این خوشه کمتر از ۱۰ درصد می‌شود، پس بررسی این خوشه تمام و بررسی خوشه‌ی شماره‌ی ۲ شروع می‌شود. این فرآیند ادامه می‌یابد تا تمام خوشه‌ها بررسی شوند.

در گام بعد کافی است مقادیر $P^a_{ss'}$ و $R^a_{ss'}$ برای تمام حالت و عمل‌ها (۴ حالت و ۲ عمل) محاسبه شود. سپس مقادیر ارزش اولیه ($V^*(c_۱)$، $V^*(c_۲)$، $V^*(c_۳)$ و $V^*(c_۴)$) محاسبه تا به عنوان وزن‌ها در رابطه ۱ استفاده شود و آموزش کنترلگر با روش یادگیری تقویتی فازی آغاز شود.

۵ نتیجه‌گیری و پیشنهادات

در این پژوهش یک راهکار برای تعمیم روش تکرار ارزش برنامه‌سازی پویا جهت تنظیم اولیه‌ی پارامترهای یک معماری یادگیری تقویتی فازی ارائه شد. در این روش ابتدا داده‌های تعاملی عامل و محیط جمع‌آوری شد. با توجه به پیوسته بودن محیط مورد بررسی، با استفاده از خوشه‌بندی راهکاری جهت تقریب ماتریس احتمال انتقال حالت-عمل به حالت بعدی و ماتریس امید پاداش آنی ارائه شد. با این روش می‌توان بر چالش‌های محاسبه‌ی این ماتریس‌ها در محیط پیوسته فائق آمد. با داشتن ماتریس‌های فوق روش تکرار ارزش که یک روش برون‌خط است اجرا شده و آنگاه مقدار ارزش عمل‌های نامزد برای هر حالت بدست آمد. با داشتن مقدار ارزش تقریب زده شده ضمن آنکه می‌توان بهترین عمل نامزد را بدست آورد، امکان کاوش موثر در فضای عمل با استفاده از مقدارهای تقریب زده شده ارزش عمل‌ها به صورت برخط میسر می‌شود. بدیهی است که مقداردهی اولیه باعث افزایش سرعت یادگیری و کاهش شکست‌ها در آموزش برخط برای تنظیم نرم خواهد شد. بکارگیری این ایده در مثال ذکر شده نشان داد که خوشه‌بندی می‌تواند کمک مناسبی جهت تقریب دو ماتریس فوق باشد. همچنین می توان با توجه به خوشه‌های تولید شده نواحی قواعد فازی را مشخص نموده و با کمک آن در خصوص تعیین توابع عضویت ورودی تصمیم‌گیری نمود.

در ادامه‌ی این پژوهش و کارهای آینده از این روش برای ناوبری ربات تعقیب کننده‌ی هدف استفاده می‌شود تا بتوان ساختار توابع عضویت ورودی را تعیین نمود و همچنین مقدار ارزش اولیه را برای تالی قواعد مشخص کرد.

مراجع

[۱] و. درهمی، ف. اعلمی‌یان هرندی، م.ب. دولتشاهی (۱۳۹۶)، یادگیری تقویتی، انتشارات دانشگاه یزد.

[۲] و. درهمی، ف. اعلمی‌یان هرندی (۱۳۹۷)، مروری بر روش‌های یادگیری تقویتی فازی با معماری نقاد-تنها، سیستم‌های فازی و کاربردها، ۱ (۲)، ۱۱-۳۴.

[۳] الیاسی (۱۴۰۱)، طراحی یک کنترل‌کننده تطبیقی افق پیش‌رونده مبتنی بر سیستم استنتاج فازی TSK برای یک سیستم دینامیکی غیرخطی، سیستم‌های فازی و کاربردها، ۴ (۱)، ۱۷۱-۱۸۸.

[۴] ح. فهیمی، ج. چاچی، ا. کاظمی‌فرد (۱۴۰۱)، شبکه‌های عصبی در تحلیل اطلاعات فازی از تصاویر شبکیه چشم، سیستم‌های فازی و کاربردها، ۴ (۲)، ۱-۲۰.

[5] McClement, D. G., Lawrence, N. P., Backström, J. U., Loewen, P. D., Forbes, M. G., & Gopaluni, R. B. (2022). Meta-reinforcement learning for the tuning of PI controllers: An offline approach. *Journal of Process Control*, *118*, 139-152.

[6] Elguea-Aguinaco, Í., Serrano-Muñoz, A., Chrysostomou, D., Inziarte-Hidalgo, I., Bøgh, S., & Arana-Arexolaleiba, N. (2023). A review on reinforcement learning for contact-rich robotic manipulation tasks. *Robotics and Computer-Integrated Manufacturing*, *81*, 102517.

[7] Vinyals, O., Babuschkin, I., Chung, J., Mathieu, M., Jaderberg, M., Czarnecki, W. M., ... & Silver, D. (2019). Alphastar: Mastering the real-time strategy game starcraft ii. *DeepMind blog*, *2*, 20.

[8] Afsar, M. M., Crump, T., & Far, B. (2022). Reinforcement learning based recommender systems: A survey. *ACM Computing Surveys*, *55*(7), 1-38.

[9] Yang, T., Zhao, L., Li, W., & Zomaya, A. Y. (2020). Reinforcement learning in sustainable energy and electric systems: A survey. *Annual Reviews in Control*, *49*, 145-163.

[10] Uc-Cetina, V., Navarro-Guerrero, N., Martin-Gonzalez, A., Weber, C., & Wermter, S. (2022). Survey on reinforcement learning for language processing. *Artificial Intelligence Review*, 1-33.

[11] Lobbezoo, A., Qian, Y., & Kwon, H. J. (2021). Reinforcement learning for pick and place operations in robotics: A survey. *Robotics*, *10*(3), 105.

[12] Beltran-Hernandez, C. C., Petit, D., Ramirez-Alpizar, I. G., Nishi, T., Kikuchi, S., Matsubara, T., & Harada, K. (2020). Learning force control for contact-rich manipulation tasks with rigid position-controlled robots. *IEEE Robotics and Automation Letters*, *5*(4), 5709-5716.

[13] Wu, K., Wang, H., Esfahani, M. A., & Yuan, S. (2021). Learn to navigate autonomously through deep reinforcement learning. *IEEE Transactions on Industrial Electronics*, *69*(5), 5342-5352.

[14] Kiran, B. R., Sobh, I., Talpaert, V., Mannion, P., Al Sallab, A. A., Yogamani, S., & Pérez, P. (2021). Deep reinforcement learning for autonomous driving: A survey. *IEEE Transactions on Intelligent Transportation Systems*, *23*(6), 4909-4926.

[15] Lou, X., Yin, Q., Zhang, J., Yu, C., He, Z., Cheng, N., & Huang, K. (2022). Offline reinforcement learning with representations for actions. *Information Sciences*, *610*, 746-758.

[16] Fathinezhad, F., Derhami, V., & Rezaeian, M. (2016). Supervised fuzzy reinforcement learning for robot navigation. *Applied Soft Computing*, 40, 33-41.

[17] Harandi, F. A., Derhami, V., & Jamshidi, F. (2019). A new feature selection method based on task environments for controlling robots. *Applied Soft Computing*, 85, 105812.

[18] Chebotar, Y., Hausman, K., Lu, Y., Xiao, T., Kalashnikov, D., Varley, J., Irpan, A., Eysenbach, B., Julian, R.C., Finn, C. & Levine, S. (2021). Actionable Models: Unsupervised Offline Reinforcement Learning of Robotic Skills. *Proceedings of the 38th International Conference on Machine Learning (PMLR)*, 139, 1518-1528.

[19] Derhami, V., Majd, V. J., & Ahmadabadi, M. N. (2008). Fuzzy Sarsa learning and the proof of existence of its stationary points. *Asian Journal of Control*, *10*(5), 535-549.

[20] R. S. Sutton and A. G. Barto. Reinforcement learning: An introduction. MIT press Cambridge, 1998.

[21] Zhou, K., Yang, S., & Shao, Z. (2017). Household monthly electricity consumption pattern mining: A fuzzy clustering-based model and a case study. *Journal of cleaner production*, *141*, 900-908.

[22] Saini, P., Kaur, J., & Lamba, S. (2021). A Review on Pattern Recognition Using Machine Learning. *Advances in Mechanical Engineering: Select Proceedings of CAMSE 2020*, 619-627.

[23] Li, C., Kulwa, F., Zhang, J., Li, Z., Xu, H., & Zhao, X. (2021). A review of clustering methods in microorganism image analysis. *Information technology in biomedicine*, 13-25.

[24] Subramaniam, M., Kathirvel, A., Sabitha, E., & Basha, H. A. (2021). Modified firefly algorithm and fuzzy c-mean clustering based semantic information retrieval. *Journal of Web Engineering*, 33-52.

[25] M. Yazdian-Dehkordi, F. Nadi, S. Abbasi (2022). Adaptive Gaussian Density Distance for Clustering, *Tabriz Journal of Electrical Engineering*, 52 (3), 205-215.