
بسم الله الرحمن الرحيم

دانشگاه صنعتی اصفهان

دانشکده مهندسی برق و کامپیوتر

بهبود کیفیت و سرعت یادگیری در سیستم‌های چندعامله با استفاده از ماتریس ارجاع و انتگرال فازی

پایان‌نامه کارشناسی ارشد مهندسی کامپیوتر – هوش‌مصنوعی و رباتیک

داریوش حسن‌پور آده

استاد راهنما

دکتر مازیار پالهنگ

۱۳۹۵

دانشگاه صنعتی اصفهان

دانشکده مهندسی برق و کامپیوتر

پایان‌نامه کارشناسی ارشد رشته مهندسی کامپیوتر – هوش‌مصنوعی و رباتیک آقای

داریوش حسن‌پور آده

تحت عنوان

بهبود کیفیت و سرعت یادگیری در سیستم‌های چندعامله با استفاده از ماتریس ارجاع و انتگرال فازی

در تاریخ ... توسط کمیته تخصصی زیر مورد بررسی و تصویب نهایی قرار گرفت:

۱ ـ استاد راهنمای پایان‌نامه	دکتر مازیار پالهنگ
۳ ـ استاد داور (اختیاری)	دکتر ...
۴ ـ استاد داور (اختیاری)	دکتر ...
سرپرست تحصیلات تکمیلی دانشکده	دکتر محمد رضا تابان

تشکر و قدردانی

پروردگار منّان را سپاسگزارم .....

کلیه حقوق مادی مترتب بر نتایج مطالعات، ابتکارات و نوآوری‌های ناشی از تحقیق موضوع این پایان‌نامه متعلق به دانشگاه صنعتی اصفهان است.

دلتنگی‌های آدمی را باد ترانه‌ای می‌خواند
رویاهایش را آسمان پر ستاره نادیده می‌گیرد
و هر دانه‌ی برفی به اشکی نریخته می‌ماند.
سکوت سرشار از سخنان ناگفته است؛
از حرکات ناکرده،
اعتراف به عشق‌های نهان،
و شگفتی‌های به زبان نیامده،
دراین سکوت حقیقت ما نهفته است؛
حقیقت تو و من.

برای تو و خویش
چشمانی آرزو می‌کنم،
که چراغ‌ها و نشانه‌ها را در ظلمات‌مان ببیند.
گوشی،
که صداها و شناسه‌ها را در بیهوشی‌مان بشنود.
برای تو و خویش،
روحی،
که این‌همه را در خود گیرد و بپذیرد.
و زبانی
که در صداقت خود ما را از خاموشی خویش بیرون کشد،
و بگذارد از آن‌چیزها که در بندمان کشیده است، سخن بگوییم.

پنجه درافکنده‌ایم با دست‌هایمان
به جای رها شدن
سنگین سنگین بر دوش می‌کشیم
بار دیگران را
به جای همراهی کردن‌شان!
عشق ما نیازمند رهایی است نه تصاحب
در راه خویش ایثار باید نه انجام وظیفه...

بی‌اعتمادی دری است
خودستایی، چفت و بست غرور است
و تهی دستی، دیوار است و لولاست
زندانی را که در آن محبوس رآی خویش‌ایم
دلتنگی‌مان را برای آزادی و دلخواه دیگران بودن
از رخنه‌هایش تنفس می‌کنیم...

ـ مارگوت بیگل

فهرست مطالب

فهرست تصاویر

چکیده

واژه‌های کلیدی: ۱ـ سیستم‌های چندعامله، ۲ـ یادگیری مشارکتی، ۳ـ یادگیری تقویتی، ۴ـ دانش غیرافزایشی، ۵ـ انتگرال فازی.

فصل اول

مفاهیم علمی پیش‌نیاز پایان‌نامه

۱ـ۱ اندازه‌گیری و انتگرال فازی

برای درک روش پیشنهادی نیاز به داشتن اطلاعات پایه در مورد اندازه‌گیری‌های فازی[1] و انتگرال فازی داریم که با هدف جمع‌آوری اطلاعات[2] ارائه شده‌اند. اندازه‌گیری‌های فازی پیش‌زمینه‌ای بر انتگرال‌های فازی هستند که قبل از آنکه آشنایی با انتگرال‌های فازی نیاز به معرفی اندازه‌گیری‌های فازی داریم. اگر فرض کنیم که تعداد منبع اطلاعاتی $X = \{x_1, x_2, \cdots, x_n\}$ که این منابع اطلاعاتی اطلاعات دریافتی از سنسورها، پاسخ‌های داده شده به یک پرسشنامه و غیره باشند. اندازه‌گیری فازی میزان ارزش اطلاعاتی این منابع را در اختیار ما می‌گذارد. معمولا اندازه‌گیری فازی توسط تابع $g : 2^{|X|} \to [0, 1]$ تعریف می‌شود که ورودی آن یک زیرمجموعه‌ای از منابع اطلاعاتی می‌باشد و خروجی آن یک مقدار مابین صفر و یک که میزان ارزش اطلاعاتی که آن زیرمجموعه از منابع اطلاعاتی ورودی تابع را مشخص می‌کند.

این تابع باید دارای شرایط مرزی تعریف شده و یکنوختی باشد که در ادامه به معرفی شرایط می‌پردازیم [۱]:

[1] Fuzzy measures

[2] Aggregate Information

۱. شرایط مرزی: اگر اطلاعاتی در دست نداریم ارزش صفر را دارد و کلیه اطلاعاتی حداکثر ارزش ۱ را دارد.

$$g(\emptyset) = 0, \quad g(X) = 1 \tag{۱-۱}$$

۲. یکنواختی ـ غیر کاهشی: اگر اطلاعات بیشتری به دست آمد ارزش کلیه اطلاعات که شامل اطلاعات جدید می‌باشد حداقل به اندازه زمانی است که آن اطلاعات جدید بدست نیامده است.

$$A \subseteq B \subseteq X \Rightarrow g(A) \leq g(B) \leq 1 \tag{۱-۲}$$

مقادیر تابع g یا توسط کارشناس ارائه می‌شود یا توسط یک تابعی مدل می‌شود، یکی از توابع معروف برای تخمین مقادیر تابع g تابع اندازه‌گیری‌ـλ سوگنو[1] می‌باشد که به صورت زیر تعریف می‌شود [۲].

$$g(\{x_1, \cdots, x_l\}) = \frac{1}{\lambda}\left[\prod_{i=1}^{l}(1+\lambda g_i) - 1\right] \tag{۱-۳}$$

که در معادله ۱ـ۳ مقدار g_iها مقادیر ارزش هریک از منابع اطلاعاتی است و λ بگونه‌ای تعیین می‌گردد که $g_\lambda(X) = 1$ شود که این مقدار برابر با جواب معادله‌ی زیر باشد.

$$\lambda + 1 = \prod_{i=1}^{n}(1+\lambda g_i), \quad \lambda \in (-1, \infty) \tag{۱-۴}$$

نکته‌ای که در رابطه با تابع اندازه‌گیری‌ـλ سوگنو باید توجه کرد این است که به ازای مقادیر n مختلف باید ریشه‌یابی بروی متغییر λ صورت گیرد؛ این ویژگی باعث می‌شود که این تابع در بعضی از کاربردها کارایی نداشته باشد.

انتگرال فازی در واقع یک تعمیمی به روش میانگین وزنی[2] می‌باشد بطوری که نه تنها مشخصه‌های مهم تک تک ویژگی‌ها را در نظر می‌گیرد بلکه اطلاعات تعاملات بین ویژگی‌ها را نیز در نظر می‌گیرید [۳]. از میان انتگرال‌های فازی دو انتگرال سوگنو[3] و چوکت[4] از الگوریتم‌هایی هستند که می‌توانند بروی هر اندازه‌گیری فازی مورد استفاده واقع شود [۴]. فرض کنیم که تابعی چون $h : X \to [0, 1]$ وجود دارد که مقادیر منابع اطلاعتی را

[1] Sugeno λ-Measure
[2] Weighted Arithmetic Mean
[3] Sugeno
[4] Choquet

به بازه‌ی $[1,0]$ نگاشت می‌کند. در واقع h تابع پشتیبان[1] منابع اطلاعاتی می‌باشد. انتگرال فازی سوگنو به صورت زیر تعریف می‌شود [۵]:

$$\int_s h \circ g = S_g(h) = \bigvee_{i=1}^{n} h(x_{\pi_i^s}) \wedge g(A_i^s) \tag{۵-۱}$$

$$h \xrightarrow{\pi^s} h(\pi_1^s) \leq h(\pi_2^s) \leq \cdots \leq h(\pi_n^s) \tag{۶-۱}$$

$$A_i^s = \{x_{\pi_i^s}, x_{\pi_2^s}, \cdots, x_{\pi_n^s}\} \tag{۷-۱}$$

در انتگرال سوگنو لازم است که مقادیر منابع اطلاعاتی را مرتب کنیم که π^s عملگر جایگشت انتگرال فازی سوگنو می‌باشد. نمادهای $\vee$ و $\wedge$ به ترتیب عمگرهای max و min می‌باشد. انتگرال فازی چوکت به صورت زیر تعریف می‌شود [۶]:

$$\int_c h \circ g = \mathcal{C}_g(f) = \sum_{i=1}^{n} \Big(f(x_{\pi^c_{(i)}}) - f(x_{\pi^c_{(i-1)}}) \Big) \cdot g(A_i^c) \tag{۸-۱}$$

$$f \xrightarrow{\pi^c} f(\pi_1^c) \leq f(\pi_2^c) \leq \cdots \leq f(\pi_n^c) \tag{۹-۱}$$

$$A_i^c = \{x_{\pi_i^c}, x_{\pi_2^c}, \cdots, x_{\pi_n^c}\} \tag{۱۰-۱}$$

$$\pi_0^c = 0, \quad x_{\pi_0^c} = 0 \tag{۱۱-۱}$$

در رابطه‌ی بالا $f : X \to \mathbb{R}$ می‌باشد که از وجه تمایز انتگرال فازی چوکت با سوگنو می‌باشد و π^c عملگر جایگشت انتگرال فازی چوکت می‌باشد. [۶].

انتگرال‌های فازی سوگنو و چوکت در حالت کلی دارای تفاوت‌هایی هستند که از جمله‌ی مهم‌ترین این ویژگی‌ها تفاوت تعریف توابع h و f در این انتگرال‌ها می‌باشد که باعث می‌شود انتگرال چوکت برای تبدیل‌های مثبت خطی[2] مناسب باشد؛ بدین معنی که تجمیع اعداد کاردینال[3] (که اعداد دارای مفاهیم واقعی هستند) را انتگرال چوکت بهتر مدل می‌کند در حالی انتگرال سوگنو برای اعداد ترتیبی[4] مناسب است [۷]. به همین علت در این پژوهش انتگرال فازی چوکت مورد استفاده قرار گرفته است زیرا که ورودی انتگرال اعداد کاملا معنی‌دار می‌باشد و اعمال تابع h بروی مقادیر منابع اطلاعاتی، معانی آن‌ها را تغییر داده و اطلاعات بدرد نخوری را تولید خواهد کرد.

---

[1] Support

[2] Positive Linear Transformation

[3] Cardinal Aggregation

[4] Ordinal Numbers

فصل دوم

روش پیشنهادی

۲ـ۱ مقدمه

در این فصل جزییات روش پیشنهادی به طور مفصل معرفی خواهد شد، روش ارائه شده در حالت کلی از دو قسمت تشکیل شده است؛ اولین و مهم‌ترین قسمت ارائه یک معیار خبرگی جدید به نام معیار خبرگی «ارجاع» که برای هر عامل در هر چرخه یادگیری محاسبه و در یک «ماتریس ارجاع» نگه‌داری می‌شود. دومین قسمت مربوط به ترکیب دانش‌های عامل‌ها هستند که با استفاده از یک مدل انتگرال فازی، صورت می‌گیرد. همانطور که در فصل بعدی نیز نشان داده خواهده شد استفاده از مدل انتگرال فازی به دلیل خواصی مهمی که این مدل دارد باعث می‌شود سرعت و کیفیت یادگیری به طرز چشم‌گیری افزایش یابد. در این فصل ابتدا به معرفی معیار «ارجاع» و دلیل استفاده از آن می‌پردازیم سپس یادگیری مشارکتی چندعامله با استفاده از ماتریس ارجاع و انتگرال فازی معرفی خواهد شد و در نهایت نشان داده خواهد شد که چرا استفاده از انتگرال فازی نتایج بهتری را نسبت به مدل های سنتی چون مدل مجموع وزنی[1] را ارائه می‌دهد.

[1] Weighted Sum

۲ـ۲ معیار خبرگی ـ ماتریس ارجاع و خاطره

در دنیای واقعی «خبرگی» تعاریف متعددی به خود گرفته است، در روانشناسی خبرگی به معنی عملکرد برتر عامل تلقی می‌شود. در جامعه شناسی خبره به فردی گفتی برچسب خبرگی توسط یک گروهی به فرد زده شده است و آن گروه به توانایی که آن فرد در اختیار دارد علاقه‌مند[1] است. در فلسفه خبره به فردی گفته می‌شود که دانشی که فرد تازه‌کار در اختیار ندارد را دارا می‌باشد [۸]. اگر تعاریف مختلف «خبرگی» را بررسی کنیم می‌بینیم که همه‌ی تعاریف در واقع تعبیری از میزان کیفیت عمکرد عامل نسبت به دیگر عامل‌ها می‌باشد. این تعبیر کلی از «خبرگی» انگیزه‌ای شد که درصدد معرفی معیاری برآیم که در حالت کلی بتوان به کلیه‌ی تعاریف «خبرگی» قابل تعمیم باشد.

تئوری ۲ـ۱. *فرض می‌کنیم عامل A در محیط $\mathcal{E}$ در پی رسیدن به یک مجموعه اهداف $\mathcal{G} \subseteq \{g_1, g_2, \cdots, g_n\}$ می‌باشد. میزان خبرگی عامل رابطه‌ی معکوسی با میزان تلاش عامل برای رسیدن به اهداف تعریف شده خود دارد.*

طبق آنچه که در تئوری بالا آورده شده است از بین چند عاملی که در یک محیط و یک مجموعه از اهداف فعالیت می‌کنند، عاملی خبره‌تر است که تلاش کمتری برای رسیدن به آن مجموعه اهداف می‌کند. شاید این مساله در نگاه اول نامتعارف به ذهن برسد ولی در فعالیت‌های روزمره ما انسان‌ها نیز به کررات شاهد این امر می‌باشیم. به عنوان مثال رانندگی دو فرد مبتدی و حرفه‌ای را در نظر بگیریم؛ فرد مبتدی هنگام رانندگی تمام حواس خود را معطوف به رانندگی می‌کند تلاش بسیار زیادی برای کنترل نسبت میزان کلاچ و گاز می‌کند و هنگام رانندگی به طور طبیعی رانندگی نمی‌کند و ... ولی فرد خبره کلیه موارد ذکر شده را بطور خودکار و طبیعی انجام می‌دهد بطوری که انگار رانندگی مانند دیگر رفتارهای طبیعی وی چون نفس کشیدن می‌باشد، که بصورت خودکار صورت می‌پذیرد. از این گونه مثال‌ها از کاربرد تئوری ۲ ـ ۱ در زندگی روزمره ما زیاد می‌توان یافت.

توجه شود که در تئوری ۲ ـ ۱ عبارت «میزان تلاش» عامل می‌تواند در کاربردهای مختلف تعابیر مختلفی به خود بگیرد، مثلا در مثال راننده‌ی مبتدی و خبره میزان نسبت مسافت طی شده بر زمان رانندگی را می‌توان به عنوان «میزان تلاش» عامل در نظر گرفت که در شرایط یکسان راننده‌ی خبره‌تر به طور نسبی در زمان کوتاه‌تری یک مسافت مشخصی را طی خواهد کرد(در رد کردن پیچ و خم‌های ترافیک و مدت زمان ترمز و ... زمان کمتری را تلف می‌کند). یا به عنوان مثال دیگر، دانشجوی قوی و دانشجوی ضعیف را مورد بررسی قرار دهیم، دانشجویی خبره هست که زمان کمتری را صرف حل صحیح یک مساله خاص کند(با فرض اینکه دانشجوها حتما باید مساله را حل کنند). همانطور که دیدیم کمیت «میزان تلاش» عامل برای مسائل مختلف معیار متفاوتی را دربر می‌گیرد ولی همگی از همان اصل معرفی شده در تئوری ۲ ـ ۱ تبعیت می‌کنند.

[1] Interested

در یادگیری مشارکتی با استفاده از تئوری ۲ـ۱ می‌توان با تعریف ۲ـ۱ یک معیار خبرگی جدید را معرفی کرد که مبنی و پایه‌ی دستآوردهای این پژوهش می‌باشد.

تعریف ۲ـ۱ (معیار خبرگی «میزان ارجاع»)**.** فرض می‌کنیم مجموعه‌ای از عامل‌ها $\mathbb{A} = \{\mathcal{A}_1, \mathcal{A}_2, \cdots, \mathcal{A}_m\}$ در محیط $\mathcal{E}$ در پی رسیدن به یک مجموعه اهداف $\mathcal{G} \subseteq \{g_1, g_2, \cdots, g_n\}$ می‌باشند. اگر ما به طور مجازی و دلخواه محیط $\mathcal{E}$ را به k ناحیه مانند e_i افراز کنیم بطوری که $\mathcal{E} = \{\cup_{i=1}^{k} e_i \mid \forall i, j \in \{1, 2, \cdots, k\} \wedge i \neq j : e_i \cap e_j = \varnothing\}$ طبق تئوری ۲ـ۱ در هر ناحیه iام عاملی خبره‌تر است که میزان حضور آن عامل در آن ناحیه کمتر از دیگران است.

در تشریح آنچه که در تعریف ۲ـ۱ آمده است می‌توان گفت که در سیستم‌های چندعاملی که همگی عوامل در یک محیط به صورت مستقل در حال فعالیت هستند؛ محیط را به چند ناحیه دلخواه افراز می‌کنیم که اجتماع نواحی باهم کل محیط $\mathcal{E}$ را تشکیل دهند و هیچ دو ناحیه‌ای اشتراکی باهم نداشته باشند [۹]. در این چنین افرازی از محیط، در هرناحیه عاملی که نسبت به بقیه خبره‌تر است، نسبت به بقیه عوامل در همان ناحیه میزان تمایل حضور کمتری را از خود نشان می‌دهند. به عبارت دیگر عاملی که خبره‌تر است تمایل دارد کوتاه‌ترین مسیر رسیدن به اهداف خود را طی کند که نهایتا منجر خواهد شد که میزان حضور عامل در هریک از نواحی محیط کمینه شود.

آنچه که در تئوری ۲ـ۱ در مورد «میزان تلاش» عامل آمده است در تعریف ۲ـ۱ در به صورت «میزان حضور عامل در هر ناحیه» تعریف شده است. بطوری که طبق تئوری مطرح شده میزان خبرگی عامل در هر ناحیه رابطه‌ی معکوسی با میزان حضور عامل در همان ناحیه را دارد. زیرا اگر عامل نسبت به محیط خود شناخت کامل‌تری داشته در هنگام تلاش برای رسیدن به اهداف خود به علت شناخت خوبی که از محیط دارد کمتر در محیط پرسه می‌زند(کمتر تلاش می‌کند) و با تعداد گام کمتری به سمت اهداف خود حرکت می‌کند – در واقع مسیر بهتری/کوتاه‌تری برای رسیدن به هدف را می‌شناسد. این موضوع در نهایت منجر می‌شود که عاملی که در هر ناحیه خبره‌تر است در همان ناحیه میزان پرسه زدن(حضور/تلاش) کمتری نسبت به دیگر عامل‌ها که از خبرگی نسبی کمتری برخوردار است را داشته باشد.

تا به اینجا گفته شد که عاملی که از خبرگی بیشتری برخوردار است لزوما کمتر در محیط پرسه می‌زند و با طی کردن مسیر کوتاه‌تر به سمت اهداف خود، تلاش کمتری می‌کند ولی چند سوال در اینجا مطرح می‌شود که برای حل مساله نیازمند پاسخ به آن‌ها هستیم.

۱. میزان حضور عامل را در نواحی مختلف، که محیط از dـبعد تشکیل شده است چگونه مدل شود؟

۲. اگر عاملی که در هر چرخه یادگیری به یکی از نواحی کلا وارد نشد و میزان پرسه زدن عامل در آن ناحیه صفر شود؛ آیا این مقدار کمینه پرسه زدن، نشان دهنده‌ی خبرگی عامل در آن ناحیه است؟

۳. چگونه در معیار خبرگی ارائه شده باید مساله عدم حضور عامل در یکی از نواحی را مدل کرد، بگونه‌ای که اثر سوئی بر تجربه‌ی دیگر عامل‌ها در آن نواحی، در هنگام ترکیب دانش عامل‌ها نداشته باشد؟

پاسخ به این سوالات برای حل مساله با استفاده از معیار خبرگی پیشنهادی (تعریف ۲ـ۱) ضروری است. ما به ازای کلیه‌ی نواحی یک ماتریسی به نام «ماتریس ارجاع»(یا به اختصار REFMAT[1]) در نظر میگیرم که در ابتدا صفر مقداردهی شده‌اند و هر دفعه که عامل از موقعیتی به موقعیت دیگر می‌رود مقدار آن ناحیه‌ای که موقعیت جدید در آن واقع است را یک واحد افزایش می‌دهیم بدین وسیله میزان حضور عامل در نواحی مختلف را می‌شماریم. همانطور که در قسمت آزمایشات این پایان‌نامه نشان داده شده است در صورتی که از تابع انتخاب عمل بولتزمن استفاده کنیم میزان کوچک یا درشت بودن این نواحی در کیفیت نتیجه تاثیرگذار نیست. یعنی عملا چه ما در حالت کلی، کل محیط را به عنوان یک ناحیه در نظر بگیریم و میزان حضور عامل در این ناحیه را بشماریم (که معادل می‌شود با تعداد گام‌های عامل در طی رسیدن به هدف) یا در حالت جزئی به ازای هر موقعیت موجود را یک ناحیه در نظر بگیرم (که معادل می‌شود با تعداد ملاقات هر یکی از موقعیت‌ها توسط عامل) به یک نتیجه می‌رسیم.

به همین دلیل در پاسخ به سوال دوم، اگر تعداد نواحی زیاد باشد (مثلا هر موقعیت یک ناحیه باشد – حداکثر تعداد نواحی) ممکن است عامل در طی رسیدن به هدف برخی از نواحی را کلا ملاقات نکند و مقدار ارجاع به آن نواحی صفر شود و از طرفی طبق تعریف ۲ـ۱ عاملی که تعداد حضور کمتری در نواحی مختلف داشته باشد از خبرگی بیشتری در آن نواحی برخوردار است و در این شرایط که مقدار ارجاع عامل به ناحیه‌ای صفر باشد را نمی‌توان به خبرگی عامل در آن ناحیه نسبت داد زیرا که آن عامل در کل، آن ناحیه را ملاقات نکرده است که بخواهد تجربه‌ای را در تعامل با آن ناحیه کسب کند تا بتواند خبرگی خود را در آن ناحیه افزایش دهد. برای حل این مشکل و پاسخ به سوال سوم، ماتریسی جدیدی به نام ماتریس خاطره(یا به اختصار RCMAT[2]) را معرفی می‌کنیم. این ماتریس وظیفه‌ی نگه‌داری آخرین ارجاعات غیر صفر عامل را به هرکدام از نواحی تعریف شده را دارد و در زمان‌هایی که مقدار یک ناحیه در ماتریس REFMAT صفر باشد مقدار آن ناحیه از ماتریس RCMAT بروز رسانی می‌شود که میزان پرسه زدن عامل در آن ناحیه در آخرین باری عامل آن ناحیه را ملاقات کرده است را می‌دهد؛ در صورتی که مقدار پرسه زدن یک ناحیه در ماتریس REFMAT مقداری غیر صفر باشد مقدار ماتریس RCMAT با مقدار کنونی REFMAT آن ناحیه بروز رسانی می‌شود.

دلیل استفاده از ماتریس RCMAT این است که در یادگیری تقویتی عامل زمانی می‌توان دانش(سیاست/خبرگی)

[1] Reference Matrix

[2] Recall Matrix

خود را نسبت به نحوه‌ی عمل در یک موقعیت بهبود ببخشد که آن موقعیت را ملاقات کند. حال اگر عامل موقعیتی را ملاقات نکند دانش وی در آن موقعیت ثابت خواهد ماند به همین دلیل اگر عامل ناحیه‌ای را ملاقات نکند و مقدار REFMAT آن ناحیه صفر باشد می‌دانیم که دانش(خبرگی) عامل در آن ناحیه در این چرخه‌ی یادگیری ثابت مانده است و در صورتی که دوباره در آن ناحیه قرار می‌گرفت، حدودا به همان میزان آخرین ملاقات در آن محیط پرسه خواهد زد. به عبارت دیگر در یک چرخه یادگیری اگر هر ناحیه ملاقات نشده، مورد ملاقات واقع می‌شد، تقریبا به میزان آخرین تعداد ارجاع شده برای آن نواحی، مورد ارجاع واقع می‌شد.

۲ـ۳ یادگیری مشارکتی Q با استفاده از ماتریس ارجاع و انتگرال فازی

آنچه که تا به اکنون در مورد روش پیشنهادی این پژوهش آورده شده، معرفی یک معیار خبرگی که در برعکس بسیاری از معیارهای خبرگی که تا به کنون معرفی شده است [۱۰–۱۲] در تمامی موقعیت‌های دنیای واقعی به وفور مشاهده می‌شود و آن ارائه این تئوری است عامل خبره‌تر برای رسیدن به یک مجموعه از اهداف تلاش نسبی کمتری نسبت به دیگر عامل‌ها با خبرگی کمتر در شرایط یکسان می‌کند. حال که معیاری برای میزان خبرگی عامل‌ها در اختیار داریم چالش بعدی برای بهبود کیفیت و سرعت یادگیری مشارکتی ارائه‌ی روشی برای ترکیب دانش‌های عامل‌ها از محیط (جداول Q آن‌ها) با استفاده از معیار ارائه شده می‌باشد. روش ترکیب باید بگونه‌ای باشد که کیفیت و سرعت یادگیری مشارکتی عامل‌ها را در طی زمان نسبت زمانی که عامل‌ها بدون مشارکت یاد می‌گیرند بهتر کند. همچنین کیفیت و سرعت یادگیری همبستگی مستقیمی داشته باشند با تعداد عامل‌هایی که درحال اشتراک گذاری هستند؛ به عبارت دیگر در صورت افزایش تعداد عامل‌هایی که دانش‌های خود را به اشتراک می‌گذارند مدل ترکیب کننده‌ی دانش‌های آن عامل‌ها باید بتواند دانش بهتری تولید کند که نهایتا منجر به بهتر شدن کیفیت و سرعت کلی یادگیری عامل‌ها شود.

در این پژوهش ما انتگرال فازی را به عنوان مدل ترکیب کننده‌ی دانش‌های عامل‌ها پیشنهاد می‌دهیم. دلیل انتخاب این مدل ویژگی‌های منحصر به فردی است که این مدل کننده در اختیار دارد که مدل را کاملا مناسب برای ترکیب دانش عامل‌ها می‌کند؛ که در بخش‌های آتی فصل این ویژگی‌ها و دلایل مناسب بودن آن‌ها برای ترکیب دانش عامل‌ها آورده شده است. لازم به یادآوری است که همانطور که در قسمت ۱ـ۱ این پایان‌نامه آورده شده است ما از به دلایل فنی از انتگرال فازی چوکت استفاده می‌کنیم که در بخش های بعدی این دلایل نیز بطور مفصل شرح داده می‌شود.

الگوریتم ۱ الگوریتم پیشنهادی یادگیری مشارکتی برمبنای ماتریس REFMAT و انتگرال فازی

1: **procedure** REFMAT-COOPERATIVE-LEARNING(m)
Require: $m > 1$ ▷ The number of agents.
Ensure: Intialize the Q matrix
Ensure: Intialize the RCMAT $\leftarrow 0$
2: **while** not End Of Learning **do**
3: REFMAT $\leftarrow 0$
4: **if** In individual learning mode **then**
5: Visit the state s;
6: Select an action a based on an action selection policy;
7: Carry out the a and observe a reward r at the new state s';
8: $Q[s,a] \leftarrow Q[s,a] + \alpha(r + \lambda \max_{a'}(Q[s',a']) - Q[s,a])$;
9: Increment REFMAT($\phi(s')$) by one;
10: $s \leftarrow s'$;
11: **else if** In cooperative learning mode **then**
12: REFMAT, RCMAT $\leftarrow$ Swap(REFMAT, RCMAT);
13: $\text{CoQ}_{\text{FCI}} \leftarrow$ FCI_Combiner(All agents' Q and REFMAT matrices);
14: **for** each agent $i \leftarrow 1, m$ **do**
15: $Q_i \leftarrow \text{CoQ}_{\text{FCI}}$;

۲ـ۳ـ۱ الگوریتم پیشنهادی

در این قسمت به معرفی الگوریتم پیشنهادی می‌پردازیم. آنچه که در الگوریتم ۱ آمده است به دو قسمت تشکیل شده است، یک قسمت که مربوط یادگیری مستقل (خطوط ۵ تا ۱۰) و قسمت دیگری مربوط به یادگیری مشارکتی (خطوط ۱۲ تا ۱۵) می‌باشد. ورودی الگوریتم تعداد عامل‌ها می‌باشد و در ابتدا ماتریس‌های Q و REFMAT و RCMAT مقداردهی می‌شود. سپس تا زمانی که یادگیری پایان نیافته است ابتدا عامل‌ها در قسمت یادگیری مستقل به صورت جدا گانه در محیط فعالیت می‌کنند که رویه‌های آورده شده در خطوط ۵ تا ۸ و همچنین خط ۱۰ همان الگوریتم یادگیری Q متعارف می‌باشد [۱۳]. در قسمت یادگیری مستقل تنها خط ۹ می‌باشد که در روش پیشنهادی به شبه‌کد اضافه شده است و این تنها یک وظیفه‌ی بسیار ساده را انجام می‌دهد و آن شمارش میزان حضور عامل در هر کدام از نواحی از پیش تعیین شده است؛ $\phi(\cdot)$ یک تابع نگاشت از یک موقعیت به یک ناحیه از محیط می‌باشد.

بعد از طی یادگیری مستقل عامل‌ها به قسمت اشتراک گذاری دانش‌های خود(جداول Q) می‌رسند (خطوط ۱۲ تا ۱۵). در قسمت یادگیری مشترک ابتدا طبق آنچه که در در قسمت آورده شده است جداول REFMAT و RCMAT به صورت مشترک بروزرسانی می‌شود و سپس جداول Q و REFMAT تمامی عامل‌ها به مدل ترکیب کننده فازی معرفی شده در این پژوهش فرستاده می‌شود و مدل ترکیب کننده فازی وظیفه‌ی استخراج یک دانش جدید با در نظر گرفتن ورودی‌های آن برای جایگزینی دانش قابلی عامل‌ها می‌باشد.

الگوریتم ۲ تابع Swap معرفی شده در الگوریتم ۱

```
1: procedure Swap(REFMAT, RCMAT)
Require: size(REFMAT) = size(RCMAT)
2:     for each element r in REFMAT and its corresponding element c in RCMAT do
3:         if r = 0 then
4:             Update r = c;
5:         else
6:             Update c = r;
7:     return REFMAT, RCMAT
```

الگوریتم ۳ تابع FCI_Combiner معرفی شده در الگوریتم ۱

```
1: procedure FCI_Combiner(K⃗, R⃗)
Require: length(K⃗) = length(R⃗) = m
Ensure: Initialize CoQ_FCI
2:     for each state s do
3:         f⃗ ← {};                ▷ Contains the normalized valued of REFMATs' value for state s for all agents
4:         for each REFMAT in R⃗ do
5:             f⃗.add(REFMAT(φ(s)));
6:         A⃗ ← 1− normalize(f⃗);
7:         for each possible action a in state s do
8:             x⃗ ← {};            ▷ Contains the Q values of action a in state s for all agents
9:             for each Q in K⃗ do
10:                x⃗.add(Q[s, a]);
11:            CoQ_FCI[s, a] ← Σ_{i=1}^{m} (f(x_{π(i)}) − f(x_{π(i−1)})) · g(A⃗_i)    ▷ The Choquet Integral
12:    return CoQ_FCI;
```

الگوریتم تابع بسیار ساده می‌باشد و مقادیر غیر صفر ماتریس ارجاع را در ماتریس خاطره کپی می‌کند و مقادیر صفر ماتریس ارجاع را از ماتریس خاطره جایگزین می‌کند. این تابع در الگوریتم ۲ آمده است.

در این پژوهش در دوقسمت نوآوری صورت گرفته است، قسمت اول ارائه‌ی معیاری جدید برای سنجش معیار خبرگی که طبق تعریف ۲ـ۱ این معیار در خط ۹ الگوریتم ۱ پیاده‌سازی شده است؛ نوآوری دوم نحوه‌ی ترکیب اطلاعات دانش عامل‌ها با استفاده از انتگرال فازی که در خط ۱۳ الگوریتم ۱ و شرح جزییات پیاده‌سازی آن در الگوریتم ۳ آمده است.

ورودی‌های الگوریتم ۳ به ترتیب مجموعه‌ای از جداول Q و ماتریس‌های ارجاع(REFMAT) تمامی عامل‌ها می‌باشد بطوری که در ازای هر جدول Q یک ماتریس REFMAT متناظر وجود دارد. خروجی این الگوریتم یک جدول Q می‌باشد که از ترکیب جداول Q ورودی با درنظر گرفتن میزان خبرگی هرکدام از عامل‌ها که توسط ماتریس‌های REFMAT آن‌ها تعیین می‌شود. الگوریتم ۳ به ازای کلیه‌ی موقعیت‌ها(sها در خط ۲) ابتدا مقادیر REFMAT کلیه‌ی عامل‌ها در ناحیه‌ای که آن موقعیت در آن واقع است(که توسط تابع نگاشت $\phi(\cdot)$ بدست

می‌آید) را استخراج می‌کند و در برداری بنام $\vec{f}$[1] ذخیره می‌کند (خطوط ۴ و ۵) که در واقع میزان ارجاعات هرکدام از عامل‌ها در ناحیه‌ی $\phi(s)$ می‌باشد. بردار $\vec{f}$ معیاری برای سنجش میزان خبرگی کلی عامل‌ها در موقعیت s است، طبق آنچه که در تعریف ۲ـ۱ آمده است در هر ناحیه عاملی خبره‌تر است که مقدار REFMAT مربوط به آن ناحیه از دیگر عامل‌ها کمتر باشد. در نتیجه در خط ۶ بعد از عادی‌سازی[2] مقادیر REFMAT عامل‌ها در ناحیه‌ی $\phi(s)$ یک مکمل گیری صورت می‌گیرد تا عاملی که مقدار REFMAT کمتری دارد دارای بیشترین مقدار بعد از عادی‌سازی شود. در خط ۷ به ازای کلیه‌ی عمل‌های ممکن در موقعیت s ابتدا مقادیر Q تک‌تک عامل‌ها را در موقعیت s و عمل a در خطوط ۹ و ۱۰ در بردار $\vec{x}$ ذخیره می‌کنیم و در نهایت در خط ۱۱ با استفاده از انتگرال فازی چوکت معرفی شده در ۱ـ۸ مقدار Q مشارکتی حاصل از میزان خبرگی بردار $\vec{A}$ و مقادیر Qهای تک‌تک عامل‌ها در بردار $\vec{x}$ در موقعیت s و عمل a بدست محاسبه می‌شود.

۲ـ۳ـ۲ تعیین توابع $f(\cdot)$ و $g(\cdot)$ در انتگرال فازی چوکت

بطور خلاصه در الگوریتم ۳ دو بخش عمده دارد بخش اول مربوط استخراج میزان خبرگی عامل‌ها بگونه‌ای که عاملی که خبره‌تر از دارای مقدار خبرگی بیشتری باشد که این بخش در خطوط ۴ تا ۶ صورت می‌گیرد؛ بخش دیگر محاسبه‌ی مقادیر Q مشارکتی کلیه‌ی عمل‌های ممکن در یک موقیت با درنظر گرفتن میزان خبرگی عامل‌ها و مقادیر Q آن‌ها با استفاده از انتگرال فازی چوکت که در خطوط ۷ تا ۱۱ صورت می‌پذیرد.

آنچه که در خط ۱۱ الگوریتم ۳ مورد توجه واقع شود این است که توابع $f(\cdot)$ و $g(\cdot)$ چگونه تعریف باید تعریف شوند؟ برای تعیین تابع $f(\cdot)$ منطقی که در این پژوهش استفاده کردیم بدین صورت است که از آنجایی که خروجی تابع $g(\cdot)$ یک مقدار عددی[3] بدون واحد می‌باشد و همچنین برای اینکه خروجی انتگرال فازی خط ۱۱ را بتوان به عنوان مقادیر جدول Q مشارکتی جدید در نظر گرفت تا بتوانیم در خطوط ۱۵ الگوریتم ۱ به عنوان جدول Q تک‌تک عامل‌ها جایگذاری کنیم باید خروجی انتگرال فازی خط ۱۱ الگوریتم ۳ از جنس جدول‌های Q عامل‌ها باشد در نتیجه تابع $f(\cdot)$ باید یک تابع خطی بصورت ۲ـ۱ باشد تا خروجی انتگرال فازی همجنس مقادیر $\vec{x}$ باشد.

$$f(\omega) = a\omega + b \qquad (۲ـ۱)$$

متغییرهای a و b در ۲ـ۱ می‌تواند به عنوان پارامترهای سازگار[4] در میزان کیفیت جدول Q مشارکتی خروجی

[1] Factors
[2] Normalize
[3] Scalar
[4] Addaptive Parameters

الگوریتم ۴ الگوریتم Const-One برای تخمین تابع $g(\cdot)$ در الگوریتم ۳

```
1: procedure Const-One(A_i)
2:     if length(A_i) ≥ m then
3:         return 1;
4:     else if length(A_i) = 0 then
5:         return 0;
6:     else
7:         return 1;
```

الگوریتم ۳ موثر واقع شود ولی با این‌حال در این پژوهش مقادیر a و b هردو به ترتیب مقادیر ثابت ۱ و صفر در نظر گرفته شده‌اند که یعنی از تابع همانی به عنوان تابع $f(\cdot)$ استفاده شده است.

تابع $g(\cdot)$ یک ورودی مرتب شده طبق آنچه که در ۱ـ۱۰ آمده است می‌گیرد و در الگوریتم ۳ تعیین این تابع تاثیر زیادی بروی کیفیت خروجی الگوریتم خواهد داشت ولی چالش‌هایی برای تعیین این تابع داریم؛ تابع $g(\cdot)$ باید دارای ویژگی‌های زیر باشد:

۱. **پویا[1] باشد:** از آنجایی که تابع $g(\cdot)$ میزان اندازه‌گیری غیرافزایشی[2] منابع اطلاعاتی را در اختیار می‌گذارد [۱۴]، نیاز داریم تعیین کنیم که کدام منابع اطلاعاتی(در اینجا خبرگی عامل‌ها) در کنار هم چه ارزش افزوده‌ای دارد؛ ولی از آنجایی که در حین یادگیری مشترک روشی برای تعیین این ارزش افزوده نداریم بنابراین باید تابع $g(\cdot)$ بصورت پویا بتواند مقادیر این ارزش افزوده را تخمین بزند.

۲. **قابل گسترش[3] باشد:** زیرا که تعداد عامل‌ها در محیط متغیر است لذا باید تابع $g(\cdot)$ بگونه‌ای باشد به ازای تغییر تعداد عامل‌ها (که تغییر در تعداد اعضای بردار $\vec{A}$ را در پی دارد) قابل گسترش باشد.

یکی از روش تخمین $g(\cdot)$ که دو ویژگی بالا را داشته باشد، تابع اندازه‌گیری‌ـλ سوگنو می‌باشد ولی این تابع نیاز به ریشه‌یابی روی متغییر λ دارد که طبق آنچه که در ۱ـ۴ آمده است به ازای تعداد عامل‌های مختلف نیاز به ریشه‌یابی معادلات غیرخطی دارد که بدلیل پیچدگی محاسباتی این ریشه‌یابی و همچنین طبق نتایج حاصل از دستاوردهای این پژوهش که در فصل نتیجه‌گیری آورده شده است، در آزمایشات صورت گرفته در این پژوهش از تابع اندازه‌گیری‌ـλ سوگنو به عنوان تابع $g(\cdot)$ استفاده نشده است. یک سری توابع در این پژوهش بجهت استفاده، آزمایش و نتیجه‌گیری به عنوان $g(\cdot)$ معرفی شده است که این توابع در الگوریتم‌های ۴ تا ۷ آمده‌اند.

در الگوریتم ۴ به ازای هر ورودی دلخواه مقدار ثابت ۱ به عنوان خروجی برگشت داده می‌شود، این بدین معنی است که ارزش افزوده‌ی هرنوع ترکیب اطلاعاتی(خبرگی) برای ما دارای حداکثر ارزش می‌باشد و این مساله

[1] Dynamic
[2] Non-additive
[3] Expandable

الگوریتم ۵ الگوریتم Max برای تخمین تابع $g(\cdot)$ در الگوریتم ۳

1: **procedure** Max($\vec{A}_i$)
2: **if** length($\vec{A}_i$) $\geq m$ **then**
3: **return** 1;
4: **else if** length($\vec{A}_i$) = 0 **then**
5: **return** 0;
6: **else**
7: **return** $\max_{\vec{A}_i}$;

الگوریتم ۶ الگوریتم Mean برای تخمین تابع $g(\cdot)$ در الگوریتم ۳

1: **procedure** Mean($\vec{A}_i$)
2: **if** length($\vec{A}_i$) $\geq m$ **then**
3: **return** 1;
4: **else if** length($\vec{A}_i$) = 0 **then**
5: **return** 0;
6: **else**
7: **return** $\frac{\sum_{j=i}^{m} \vec{A}_i(j)}{\text{length}(\vec{A}_i)}$;

باعث می‌شود که نتیجه‌ی انتگرال فازی خط ۱۱ الگوریتم ۳ مقداری معادل با مقدار خبره‌ترین عامل(عاملی که کمترین پرسه را در محیط مربوطه داشته) را به عنوان مقدار جدید جدول Q مشارکتی تولید کند.

در الگوریتم ۵ میزان خبرگی خبره‌ترین عامل به عنوان خروجی تابع $g(\cdot)$ برگشت داده می‌شود. در الگوریتم ۶ خروجی، میانگین خبرگی عامل‌ها در نظر گرفته شده است و در الگوریتم ۷ طبق رابطه‌ی نوشته شده میانگین kام میزان خبرگی‌ها به عنوان خروجی برمی‌گردد به طوری که بزرگترین خبرگی در عدد k و کوچکترین خبرگی در عدد ۱ و هر آنچه که مابین این دو خبرگی وجود دارد در اندیس ترتیب مرتب شده آن‌ها ضرب می‌شود و میانگین این مجموع محاسبه می‌شود و برگشت داده می‌شود؛ توجه شود که ورودی‌های الگوریتم‌های ۴ تا ۷ طبق آنچه که در ۱ـ۱۰ آمده است یک مجموعه‌ی مرتب می‌باشد.

۲ـ۴ علت کارکرد انتگرال فازی چوکت در انتقال دانش

در این قسمت به بررسی شهودی اینکه چرا انتگرال فازی چوکت برای انتقال(ترکیب) دانش‌های عامل‌ها می‌تواند موثر واقع باشد می‌پردازیم. این شهود بعدها در آزمایش‌ها نشان داده خواهد شد که صحت دارد. انتگرال فازی چوکت یک سری ویژگی‌ها دارد که برای انتقال دانش مدل می‌کند. از مهم‌ترین ویژگی‌ها را می‌توان به موارد زیر اشاره کرد [۵].

۱. **محدود است:** اگر شرایط مرزی و یکنوایی تابع $g(\cdot)$ برقرار باشد انتگرال فازی هیچ‌گاه بیشتر از حداکثر

الگوریتم ۷ الگوریتم K-Mean برای تخمین تابع $g(\cdot)$ در الگوریتم ۳

```
1: procedure K-Mean(A_i)
2:     if length(A_i) ≥ m then
3:         return 1;
4:     else if length(A_i) = 0 then
5:         return 0;
6:     else
7:         return (Σ_{j=i,k=1}^{m,length(A_i)} k·A_i(j)) / (Σ_{j=1}^{length(A_i)} j);
```

مقدار $f(x_{\pi_i})$ها و کمتر از حداقل مقدار آن‌ها خروجی نمی‌دهد [۶]. یعنی دانش تولیدی خارج از محدوده‌ی دانش فعلی عامل‌ها نمی‌باشد فقط ترکیب بهینه‌ای از این دانش‌ها به عنوان خروجی برگشت داده می‌شود که این در کاربرد یادگیری تقویتی به این معنی است که هیچ‌گاه مقادیر جدول Q بیشتر یا کمتر از آنچه که تجربه شده نمی‌شود و این باعث می‌شود که ضمانت همگرایی یادگیری تقویتی Q با اعمال انتگرال فازی چوکت نقض نشود و الگوریتم حتما همگرا شود؛ ولی در صورتی که روشی خارج از دانش کنونی عامل‌ها خروجی دهد ضمانتی برای همگرایی عامل‌ها وجود نخواهد داشت.

۲. **می‌تواند اندازه‌گیری‌های غیرافزایشی مدل کند:** معمولا روش‌هایی که تا به‌کنون در این زمینه ارائه شده است از میانگین وزنی خبرگی عامل‌ها برای بدست آوردن جدول Q مشترک استفاده کرده‌اند [۱۰–۱۲]. این درحال هست که میانگین وزن‌دار قسمتی از مدل اندازه‌گیری‌های غیرافزایشی می‌باشد. بنابرین با درنظر گرفتن مدل‌های غیرافزایشی که در ماهیت مساله هست قدرت و انعطاف بیشتری در اختیار داریم نسبت به روش‌هایی که فقط از میانگین وزنی استفاده کرده‌اند.

تعریف ۲ـ۲ (اندازه‌گیری‌های غیرافزایشی)**.** اگر فرض کنیم (X, A) فضای قابل اندازه‌گیری باشد که X مجموعه‌ی مرجع[1] و $A \subseteq X$، آنگاه تابع مجموعه‌ای مانند μ که $\mu : A \to [0, 1]$ اندازه‌گیر غیرافزایشی می‌گویند هرگاه شرایط زیر را ارضا کند [۱۴].

- $\mu(\emptyset) = 0, \quad \mu(X) = 1$
- $A \subseteq B \Rightarrow \mu(A) \leq \mu(B)$

تورا و همکاران [۱۴] یک مجموعه جامعی در مورد اندازه‌گیری‌های غیرافزایشی ایجاد کرده‌اند که جزئیات این مطلب خارج از حوصله‌ی این نوشتار است و در صورت تمایل به کسب اطلاعات بیشتر در مورد اندازه‌گیری‌های غیرافزایشی و انتگرال‌های فازی می‌توانید به آن مراجعه نمایید.

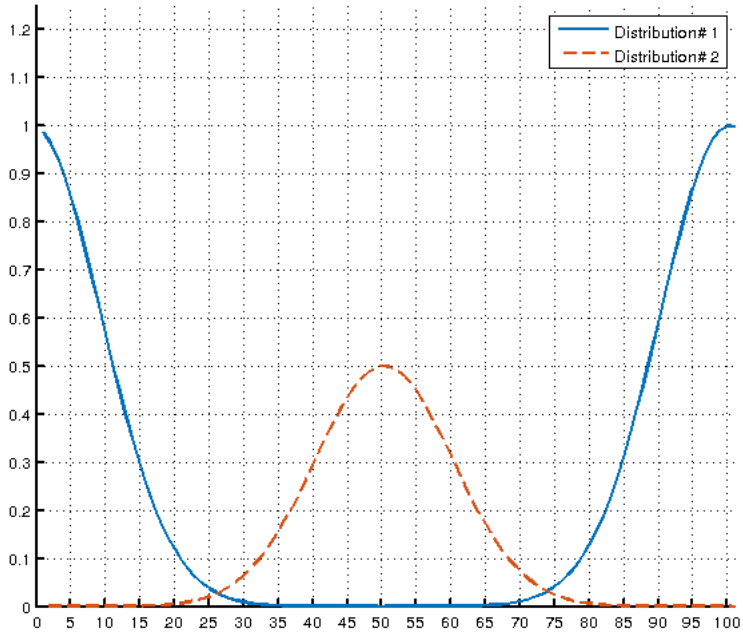
[1] Reference Set

فصل سوم

نتایج عملی

۳ـ۱ مقدمه

در این فصل به ارائه‌ی آزمایش‌های صورت گرفته بروی روش پیشنهادی می‌پردازیم و در طی این آزمایش‌ها روش پیشنهادی را با روش کوتاه‌ترین مسیر تجربه شده(یا به اختصار SEP) مقایسه می‌کنیم که آخرین و مدرن‌ترین روش ارائه شده در جهت بهبود یادگیری مشارکتی می‌باشد [۱۲]. آزمایش‌ها بروی دو محیط «پلکان مارپیچ» و «صید و صیاد» صورت گرفته است. آزمایش‌ها به دو دسته تقسیم بندی شده است؛ دسته اول آزمایش‌هایی که روش پیشنهادی را در مقابل روش SEP قرار می‌دهد و عملکرد روش پیشنهادی را مورد سنجش قرار می‌دهد. دسته دوم آزمایش‌ها مربوط به آزمون رفتار روش پیشنهادی در صورت تغییر در پارامترهای متخلف آن می‌باشد. همچنین اثر استفاده از سیاست‌های انتخاب عمل مختلف در الگوریتم ۱ نیز بررسی شده است. در روش‌های مرتبط مدرن قبلی [۱۱، ۱۲] که این پژوهش ادامه‌ی کار آن‌ها می‌باشد فقط از سیاست انتخاب عمل Boltzmann استفاده کرده‌اند؛ در این پژوهش علاوه بر Boltzmann تاثیر استفاده از روش ε – greedy بروی هردو روش پیشنهادی و SEP نیز مورد بررسی واقع گردیده است.



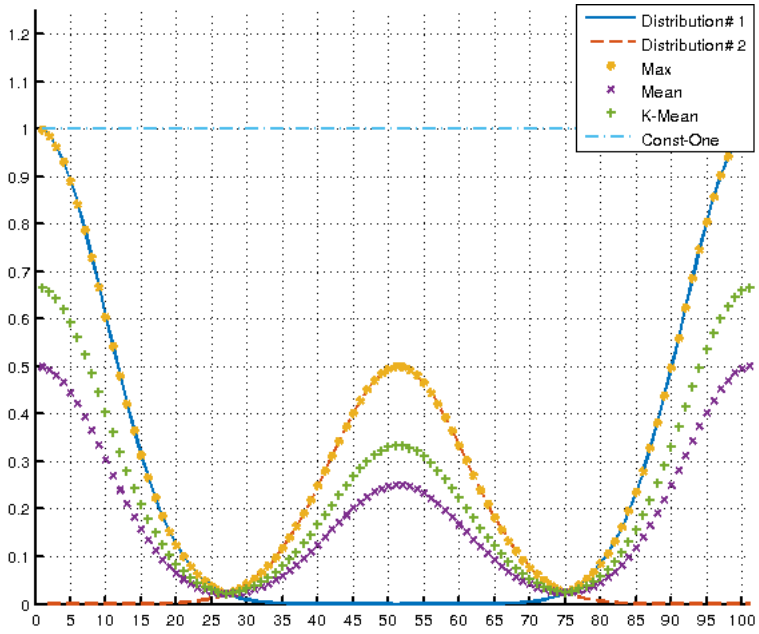
شکل ۳ـ۱: دو توزیع فرضی بجهت نمایش نحوه‌ی رفتار الگوریتم‌های ۴ تا ۷ بروی آن‌ها.

۳ـ۲ رفتار الگوریتم‌های معرفی شده برای $g(\cdot)$

در این قسمت به بررسی رفتار الگوریتم‌های ۴ تا ۷ معرفی شده برای $g(\cdot)$ بروی دو توزیع فرضی خواهیم پرداخت، زیرا که در طی اجرای آزمایش‌های مختلف نتایج تاثیر این توابع بر اجرای الگوریتم پیشنهادی ۱ آورده شده است، لذا بجهت درکت علت تاثیرات مختلف هرکدام ازین توابع بروی نتیجه‌ی الگوریتم پیشنهادی در آزمایش‌ها، درک نحوه‌ی رفتار الگوریتم‌های ۴ تا ۷ ضروری است.

برای نمایش نحوه‌ی رفتار هرکدام از الگوریتم‌ها دو توزیع فرضی شکل ۳ـ۱ فرض شده است. در صورت اعمال الگوریتم‌های ۴ تا ۷ بروی دو توزیع آورده شده در شکل ۳ـ۱ توزیع‌های جدیدی بصورت آنچه که در شکل ۳ـ۲ آمده است بدست می‌آیند. همانطور که در شکل ۳ـ۲ می‌بینیم اعمال الگوریتم Const-One بروی دو توزیع مقدار ثابت ۱ را برمی‌گرداند. اعمال الگوریتم Max در هر نقطه حداکثر مقدار هر دو توزیع را برمی‌گرداند. الگوریتم Mean میانگین دو توزیع را در هر نقطه حساب می‌کند و در نهایت الگوریتم K-Mean میانگین kام هردو توزیع را محاسبه میکند که همانطور که می‌بینیم میانگین kام به سبب ماهیت الگوریتم به سمت بیشترین مقدار پیش‌قدر[1] می‌باشد.

[1] Bias



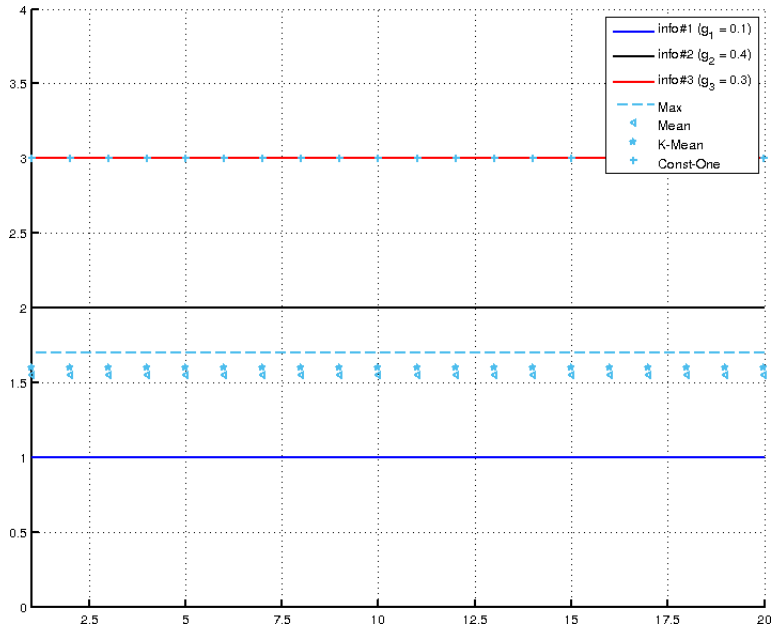
شکل ۳ـ۲: نمایش توزیع‌های جدید بدست آمده بعد از اعمال الگوریتم‌های ۴ تا ۷ بروی دو توزیع فرضی شکل ۳ـ۱

۳ـ۲ـ۱ تعابیر مختلف انتگرال فازی چوکت از داده‌ها برمبنای $g(\cdot)$

الگوریتم‌های ۴ تا ۷ به تنهایی فقط در نقش یک عملگر بازی می‌کند ولی در هنگام ترکیب دانش با انتگرال فازی چوکت به دانش خروجی الگوریتم از دیدگاه‌های متفاوتی نگاه می‌کنند. از آنجایی که در فصل‌های قبلی نیز آورده شد انتگرال فازی در واقع یک تعمیم الگوریتم دهنده‌ی میانگین وزنی می‌باشد که علاوه بر ویژگی‌هایی که روش میانگین وزنی ارائه می‌دهد می‌تواند اندازه‌گیری‌های غیرافزایشی را نیز مدل کند. لذا با تغییر تابع $g(\cdot)$ می‌توان باعث شد که انتگرال فازی چوکت تعابیر مختلفی از داده‌های ورودی خود ارائه دهد. از بین الگوریتم‌ها فقط الگوریتم Const-One دارای تعبیر صریح ریاضی می‌باشد که در ۳ـ۱ آمده است، بقیه‌ی الگوریتم‌ها دارای تعابیر صریح نیستند و فقط می‌توانیم بر اساسی نمایشی که در شکل ۳ـ۲ آمده است شهودی از نحوه‌ی تغییر رفتار انتگرال فازی به ازای هریک از الگوریتم‌ها ارائه داد.

$$g = \text{Const-One}(\cdot) \equiv \begin{cases} g(X) & = 1 \\ g(\emptyset) & = 0 \Rightarrow \mathcal{C}_g(f) \equiv \max\{f(x_{\pi^c_{(1)}}), \cdots, f(x_{\pi^c_{(n)}})\} \\ g_{A \subseteq X}(A) & = 1 \end{cases} \quad (۱ـ۳)$$

برای نمایش شهودی نحوه‌ی تغییر رفتار انتگرال فازی چوکت در شکل ۳ـ۳ سه منبع اطلاعاتی با مقادیر $y = 1$ و $y = 2$ و $y = 3$ در نظر گرفته شده است و مقدار ارزش هرکدام از این‌ها به ترتیب $g = [0.1 \quad 0.4 \quad 0.3]^T$ در نظر گرفته شده است. سپس انتگرال فازی چوکت را با در نظر گرفتن تابع همانی به عنوان تابع $f(\cdot)$ بروی این ۳ منبع اطلاعاتی اعمال کردیم و همانطور که می‌بینیم مقداری که انتگرال فازی چوکت به ازای



شکل ۳ـ۳: نمایش رفتار انتگرال فازی بروی منابع اطلاعاتی $y = 1$ و $y = 2$ و $y = 3$ به ازای توابع $g(\cdot)$های مختلف.

$g = \text{Const-One}(\cdot)$ تولید می‌کند برابر با حداکثر مقدار منابع اطلاعاتی دریافتی می‌باشد. در حالت کلی هرچقدر میانگین تابع $g_{A\subseteq X}(A)$ به سمت مقدار ۱ متمایل باشد خروجی انتگرال فازی چوکت به سمت بیشینه مقدار منابع اطلاعاتی پیش‌قدر می‌شود و در صورتی که این میانگین به سمت صفر متمایل باشد خروجی به کمینه مقدار پیش‌قدر می‌شود.

۳ـ۳ مقایسه‌ی روش پیشنهادی با روش کوتاه‌ترین مسیر تجربه شده

در این قسمت به مقایسه‌ی روش پیشنهادی با روش «کوتاه‌ترین مسیر تجربه شده» که از بروزترین تکنیک ارائه شده در این شاخه از یادگیری مشارکتی می‌باشد می‌پردازیم [۱۲]. کلیه‌ی این آزمایش‌ها در دو محیط «پلکان مارپیچ» و «صید و صیاد» صورت گرفته است. نتیجه‌ی هر آزمایش حاصل میانگین ۲۰ اجرای مستقل تمامی الگوریتم‌ها می‌باشد. همچنین به غیر از مواردی که صراحتا قید شده است تعداد عامل‌ها ۳ عدد می‌باشد – البته بدیهی است که یادگیری مستقل تک عامله(یا به اختصار IL[1]) شامل این قاعده نمی‌باشد. همچنین در کلیه‌ی آزمایش‌ها عامل‌ها از ۲۰۰ چرخه یادگیری بهره می‌برند و در هر چرخه عامل ۵ بار تلاش می‌کند که در مجموع ۱۰۰۰ تلاش صورت می‌گیرد. کلیه‌ی پارامترهای مربوط قسمت یادگیری مستقل الگوریتم ۱ اعمال شده در آزمایشات این فصل منطبق بر پارامترهای تعریف شده در [۱۲] می‌باشد که نتایج قایل قیاس باشند. در ضمن در این فصل اختصارهای جدول ۳ـ۱ را نیز داریم.

[1] Individual Learning

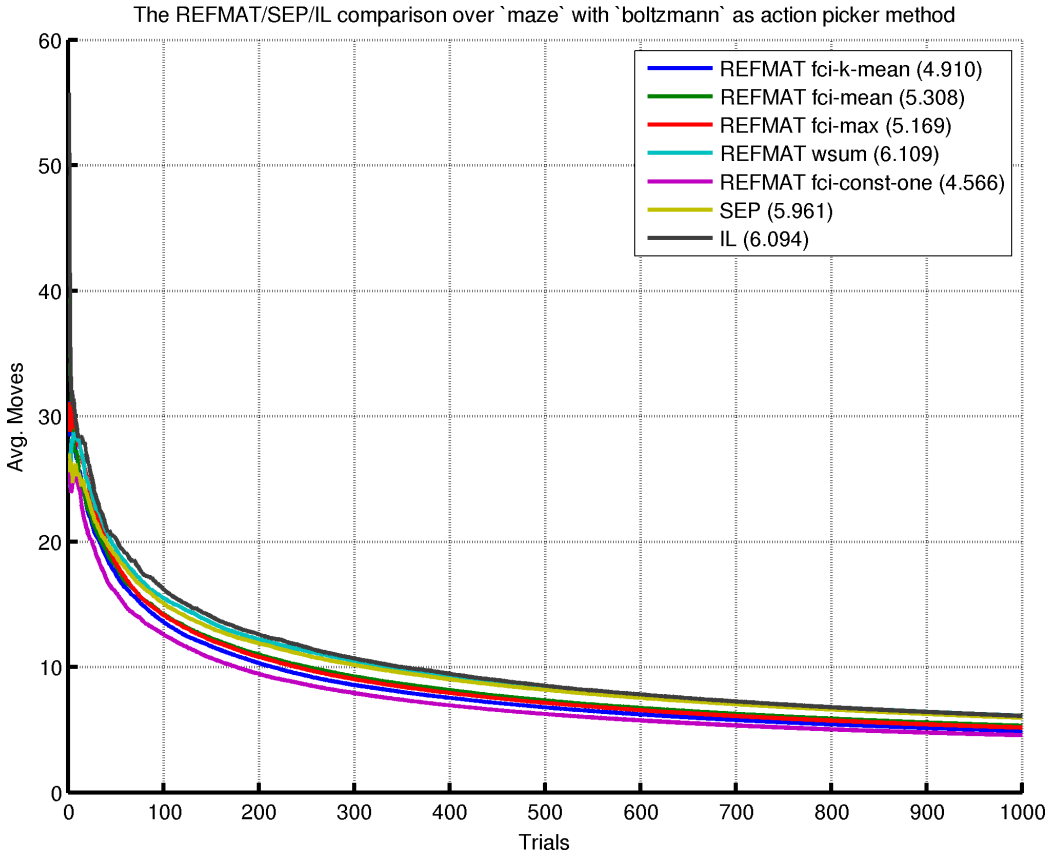
جدول ۳ـ۱: لیست اختصارهای استفاده شده در این فصل

اختصار	معنی
REFMAT	روش پیشنهادی
IL	یادگیری مستقل تک عامله
SEP	روش کوتاه‌ترین مسیر تجربه شده
wsum	میانگین وزنی
fci-max	الگوریتم Max به عنوان مدل کننده‌ی تابع $g(\cdot)$
fci-mean	الگوریتم Mean به عنوان مدل کننده‌ی تابع $g(\cdot)$
fci-k-mean	الگوریتم K-Mean به عنوان مدل کننده‌ی تابع $g(\cdot)$
fci-const-one	الگوریتم Const-One به عنوان مدل کننده‌ی تابع $g(\cdot)$
Rand-Walk	جستجوی کاملا مکاشفانه محیط

در این فصل در حالت کلی ما در دو بخش سیاست انتخاب عمل «بولتزمن» و «$-\varepsilon$حریصانه» (که از این به بعد به اختصار «تابع بولتزمن» و «تابع حریصانه» خطاب خواهیم کرد.) به مقایسه‌ی نتایج می‌پردازیم. طبق آنچه که در ادامه مشاهده خواهیم کرد چه در صورت استفاده از تابع بولتزمن و چه تابع حریصانه روش پیشنهادی چه در سرعت یادگیری و چه در کیفیت یادگیری بهتر از روش SEP می‌باشد.

برای اینکه نشان دهیم که استفاده از انتگرال فازی در بهبود نتیجه تاثیر بسزایی دارد از تابع میانگین وزنی (یا به اختصار wsum[1]) نیز استفاده کرده‌ایم. بدین صورت که بجای اینکه بعد از استخراج میزان خبرگی هر عامل جداول Q آن‌ها را به نسبت خبرگی‌ای که دارند باهم جمع می‌کنیم تا جدول Q مشارکتی تولید شود. تابع میانگین وزنی روشی است که در پژوهش‌های اخیر به کررات از آن استفاده کرده‌اند [۱۰–۱۲]. که یکی از اهداف ما در این پژوهش نمایش قدرت انتگرال‌های فازی در کاربردهای مختلف می‌باشد به‌طوری که اگر در پژوهش‌های قبلی به درستی از انتگرال فازی بهره برده می‌شد می‌توان به قطع گفت که می‌توانستند نتایج بهتری را بدست بیاورند.

[1] Weighted Sum



شکل ۳ـ۴: مقایسه در سرعت و کیفیت یادگیری در محیط پلکان مارپیچ با تابع بولتزمن

۳ـ۳ـ۱ مقایسه در محیط پلکان مارپیچ

آزمایش‌های مربوط به این قسمت در ۴ بخش صورت گرفته است؛ ۱. مقایسه در سرعت و کیفیت یادگیری، ۲. مقایسه در پیچیدگی زمانی، ۳. مقایسه در میزان باروری، ۴. مقایسه در تاثیر تعداد عامل‌ها میزان کیفیت و سرعت یادگیری می‌باشد.

سیاسیت انتخاب عمل «بولتزمن»

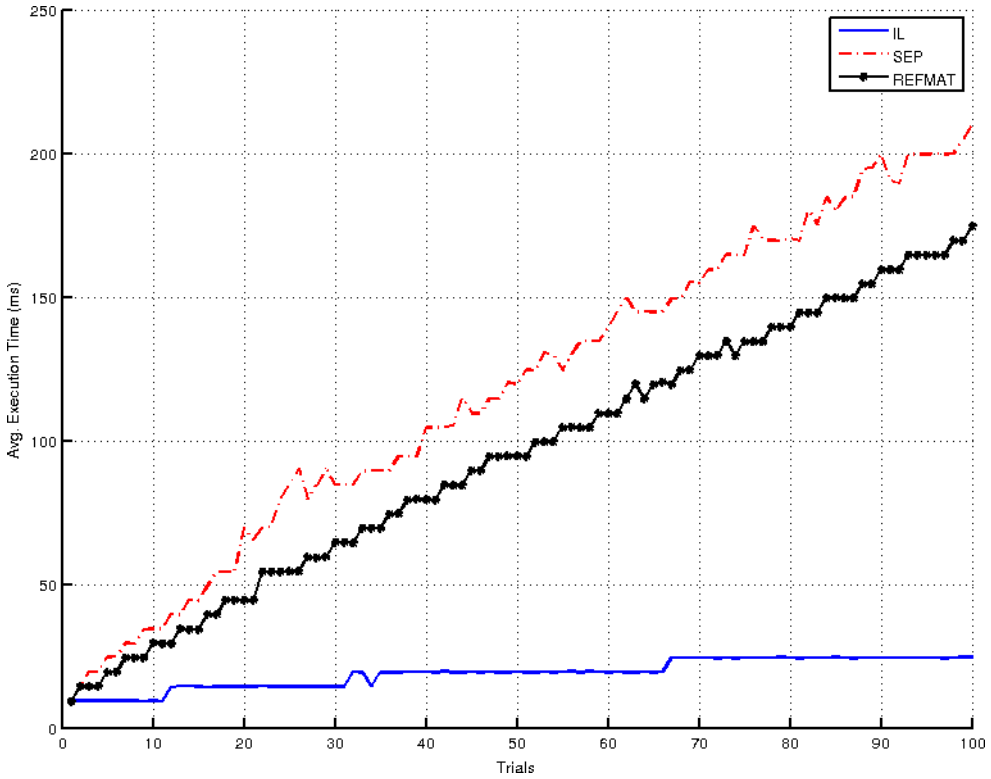
مقایسه در سرعت و کیفیت یادگیری: نتایج حاصل از اجرای الگوریتم‌ها در محیط پلکان مارپیچ در شکل ۳ـ۴ آمده است. در این شکل محور افقی تعداد تلاش‌های یادگیری عامل را نشان می‌دهد که در تلاش اول عامل بدون دانش اولیه شروع به تعامل با محیط می‌کند و در تلاش ۱۰۰۰ام عامل به اجرای خود پایان می‌دهد. محور عمودی نموار میانگین تجمعی تعداد قدم‌های عامل را نشان می‌دهد. اعداد کناری برچسب‌ها (گوشه بالا سمت راست) متوسط تعداد قدم در آخرین تلاش عامل می‌باشد که انتظار می‌رود عامل آگاهی نسبی کاملی از محیط دارد را نشان می‌دهد که این عدد هرچقدر کمتر باشد نشان می‌دهد که عامل در طی رسیدن به هدف تعداد گام کمتری برداشته است و در نتیجه دانش و شناخت بهتری از محیط دارد.

جدول ۳ـ۲: مقایسه در کیفیت یادگیری در محیط پلکان مارپیچ با تابع بولتزمن

			REFMAT				
	IL	SEP	wsum	fci-mean	fci-max	fci-k-mean	fci-const-one
IL	%0.0						
SEP	%2.2	%0.0					
wsum	%-0.2	%-2.3	%0.0				
fci-mean	%14.9	%12.5	%15.1	%0.0			
fci-max	%18.0	%15.5	%18.2	%2.7	%0.0		
fci-k-mean	%24.0	%21.4	%24.2	%7.9	%5.1	%0.0	
fci-const-one	%33.6	%30.7	%33.8	%16.2	%13.2	%7.7	%0.0

همانطور که مشاهده می‌شود روش SEP دارای ٪۲ بهبود نسبت به IL می‌باشد در حالی که روش پشنهادی در زمانی که از انتگرال فازی استفاده می‌کند در بدترین حالت دارای ٪۱۸ بهبود و در بهترین حالات دارای ٪۳۳ بهبود می‌باشد که نسبت به روش SEP تقریبا ۹ الی ۱۶ برابر نتیجه را بهبود داده است. در صورتی که از میانگین وزنی بجای انتگرال فازی استفاده شود نتایج با اختلاف اندکی (کمتر از ۱ - ٪) بدتر از یادگیری IL بوده است که نشان می‌دهد که استفاده از انتگرال فازی چقدر می‌تواند نسبت به روش‌های سنتی و معمولی چون میانگین وزنی موثر واقع شود. نتایج این قسمت را می‌توان در جدول ۳ـ۲ خلاصه کرد.

مقایسه در پیچیدگی زمانی: در این قسمت به مقایسه‌ی پیچیدگی زمانی روش پیشنهادی با روش SEP مورد بررسی قرار می‌گیرد، برای محاسبه‌ی پیچیدگی زمانی به روش ریاضی کار بسیار دشوار و پرخطایی می‌باشد؛ در اینجا ما بجای محاسبه‌ی پیچیدگی زمانی ریاضی دو الگوریتم از مدت زمانی که طول می‌کشد برنامه در سیستم اجرا و خاتمه یابد استفاده می‌کنیم. در شکل ۳ـ۵ میانگین زمانی ۲۰ اجرای مستقل برحسب میلی‌ثانیه به ازای هریک از تعداد تلاش‌ها آورده شده است. همان‌طور که در این شکل مشاهده‌ی می‌شود الگوریتم IL دارای حداکثر سرعت اجرا می‌باشد زیرا که هیچ سربار محاسباتی یادگیری مشترک را ندارد؛ هدف یادگیری اشتراکی این است که می‌خواهد در ازای یک سری سربار محاسباتی کیفیت و سرعت «یادگیری» عامل‌ها را افزایش دهد. با در نظر داشتن این موضوع همانطور که قبلا دیدیم روش پیشنهادی سرعت و کیفیت یادگیری را بیشتر از روش SEP افزایش می‌دهد و در اینجا نیز می‌بینیم که دارای پیچیدگی زمانی کمتری نسبت به روش SEP می‌باشد که نشان از بهینه‌گی روش پیشنهادی نسبت به روش SEP می‌دهد.



شکل ۳ـ۵: مقایسه در پیچیدگی زمانی روش‌ها به ازای تعداد تلاش‌های متفاوت برحسب میلی‌ثانیه

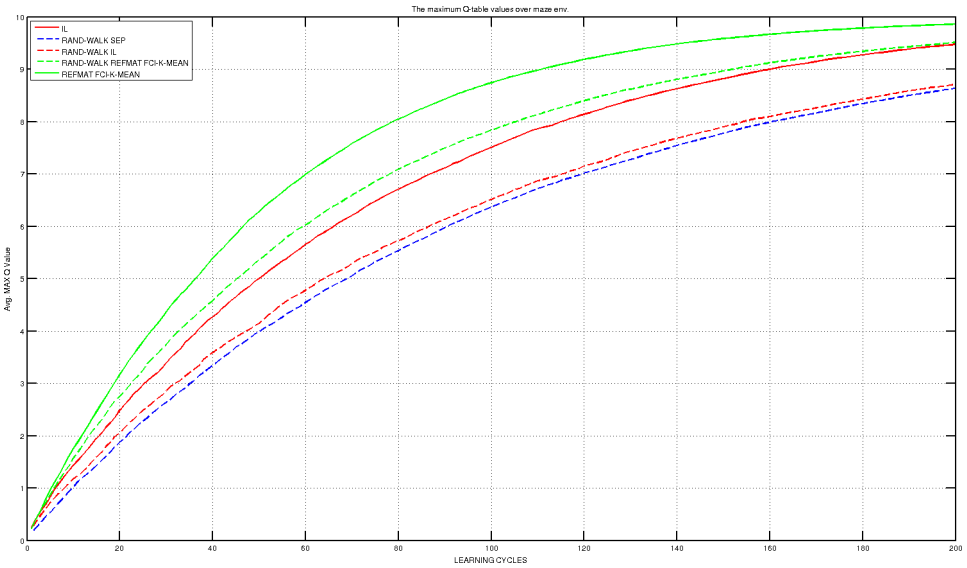
مقایسه در میزان باروری:

تعریف ۳ـ۱ (سرعت باروری)**.** اگر فرض کنیم الگوریتم یادگیری تقویتی $\psi_Q(\mathcal{E})$ وجود دارد که در محیط $\mathcal{E}$ فعالیت می کند و دانش خود را در جدولی مانند Q ذخیره می‌کند، سرعت باروری الگوریتم $\psi_Q(\mathcal{E})$ را سرعت همگرایی حداکثر مقدار جدول Q به سمت حداکثر پاداش محیط قابل دریافت تعریف می‌کنیم.

تعریف ۳ـ۲ (میزان باروری)**.** انتگرال سرعت باوری را میزان باروری الگوریتم $\psi_Q(\mathcal{E})$ که در محیط $\mathcal{E}$ فعالیت می کند و دانش خود را در جدولی مانند Q ذخیره می‌کند، تعریف می‌کنیم.

طبق تعاریف ۳ـ۱ و ۳ـ۲ الگوریتمی میزان باروری بیشتری دارد که سریع‌تر مقادیر جدول Q خود را به سمت بیشنه مقداری که می‌توانند داشته باشد(یعنی بیشنه پاداشی که از محیط می‌توانند کسب کند) سوق دهد. معمولا این در الگوریتم‌های یادگیری تقویتی Q این کار با نتظیم مقدار سرعت یادگیری α صورت می‌گیرد که باعث می‌شود الگوریتم‌ها با سرعت بیشتری به یادگیری نحوه‌ی تعامل با محیط بپردازند. لذا در شرایط یکسان می‌توان گفت الگوریتمی بهتر عمل می‌کند که نحوه‌ی تعامل با محیط را سریع‌تر نسبت به دیگر الگوریتم‌ها یاد می‌گیرید. بدین منظور تعریف ۳ـ۲ به نظر می‌رسد که می‌تواند معیار مناسبی برای مقایسه‌ی سرعت یادگیری الگوریتم‌ها باشد.

در شکل



شکل ۳ـ۶: نمودار باروری الگوریتم‌ها مختلف

آورده شده است حداکثر میزان جدول Q روش‌ها در هر تلاش آورده شده است. همانطور که قبلا در تعریف محیط پلکان مارپیچ آورده شده است حداکثر مقدار پاداش این محیط مقدار ۱۰ می‌باشد لذا همان‌طور که مشاهده می‌شود الگوریتم‌ها با شیب‌های متفاتی حداکثر مقدار جداول خود را به سمت حداکثر مقدار پاداش قابل دریافت از محیط سوق می‌دهند. در این شکل سرعت باروری شیب نمودار در هر تلاش می‌باشد و میزان باروری مساحت زیر نمودار می‌باشد.

مراجع

[1] V. Torra and Y. Narukawa, "The interpretation of fuzzy integrals and their application to fuzzy systems," *International journal of approximate reasoning*, vol. 41, no. 1, pp. 43–58, 2006.

[2] K. Leszczyński, P. Penczek, and W. Grochulski, "Sugeno's fuzzy measure and fuzzy clustering," *Fuzzy Sets and Systems*, vol. 15, no. 2, pp. 147–158, 1985.

[3] A. F. Tehrani, W. Cheng, and E. Hullermeier, "Preference learning using the choquet integral: The case of multipartite ranking," *IEEE Transactions on Fuzzy Systems*, vol. 20, no. 6, pp. 1102–1113, 2012.

[4] L. M. De Campos and M. Jorge, "Characterization and comparison of sugeno and choquet integrals," *Fuzzy Sets and Systems*, vol. 52, no. 1, pp. 61–67, 1992.

[5] M. Grabisch, "Fuzzy integral in multicriteria decision making," *Fuzzy sets and Systems*, vol. 69, no. 3, pp. 279–298, 1995.

[6] T. Murofushi, M. Sugeno, and M. Machida, "Non-monotonic fuzzy measures and the choquet integral," *Fuzzy sets and Systems*, vol. 64, no. 1, pp. 73–86, 1994.

[7] M. Grabisch, "The application of fuzzy integrals in multicriteria decision making," *European journal of operational research*, vol. 89, no. 3, pp. 445–456, 1996.

[8] "Expert - wikipedia." `https://en.wikipedia.org/wiki/Expert`. (Accessed on 11/12/2016).

[9] E. Schechter, *Handbook of Analysis and its Foundations*, ch. 1, p. 16. Academic Press, 1996.

[10] M. N. Ahmadabadi, M. Asadpur, S. H. Khodanbakhsh, and E. Nakano, "Expertness measuring in cooperative learning," in *Intelligent Robots and Systems, 2000.(IROS 2000). Proceedings. 2000 IEEE/RSJ International Conference on*, vol. 3, pp. 2261–2267, IEEE, 2000.

[11] E. Pakizeh, M. Palhang, and M. M. Pedram, "Multi-criteria expertness based cooperative q-learning," *Applied intelligence*, vol. 39, no. 1, pp. 28–40, 2013.

[12] M. ali mirzaei badizi, “Speed-up cooperative learning in multi-agent systems using shortest experimented path,” Master’s thesis, Department of Electrical and Computer Engineering, Isfahan University of Technology, Isfahan University of Technology, Isfahan 84156-83111, Iran, 3 2015.

[13] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*, vol. 1. MIT press Cambridge, 1998.

[14] V. Torra, Y. Narukawa, and M. Sugeno, *Non-Additive Measures*, pp. 3–7. Springer, 2014.

Improvments in speed and quality of learning in multi-agent systems using the reference matrix and fuzzy integral

Dariush Hasanpour Adeh

d.hasanpoor@ec.iut.ac.ir

[DATE]

Department of Electrical and Computer Engineering

Isfahan University of Technology, Isfahan 84156-83111, Iran

Degree: M.Sc. Language: Farsi

Supervisor: Assoc. Prof. Maziar Palhang (palhang@cc.iut.ac.ir)

Abstract

Key Words:
Multi-agent Systems, Cooperative Learning, Reinforcement Learning, Non-addive Knowledges, Fuzzy Integral

Isfahan University of Technology

Department of Electrical and Computer Engineering

Improvments in speed and quality of learning in multi-agent systems using the reference matrix and fuzzy integral

A Thesis

Submitted in partial fulfillment of the requirements
for the degree of Master of Science

by

Dariush Hasanpour Adeh

Evaluated and Approved by the Thesis Committee, on ...

1. Maziar Palhang, Assoc. Prof. (Supervisor)
2. ..., Prof. (Examiner)
3. ..., Prof. (Examiner)

Mohamad Reza Taban, Department Graduate Coordinator