۱۲۳۸-۱۳۹۱-۰۳-۳۰

یک روش جدید مبتنی بر معیارهای آماری توزیع برای تنظیم خودکار نرخ یادگیری آتاماتای یادگیر در محیط‌های پویا

محمدرضا ملاخلیلی میبدی، محمدرضا میبدی

چکیده - یکی از مسائل مطرح در ساخت سیستم های یادگیر نظیر شبکه های عصبی و یا آتاماتای یادگیر، تعیین نرخ یادگیری است. در اکثر موارد از یک الگوریتم کاهش یابنده در طول زمان برای تنظیم نرخ یادگیری استفاده می‌شود. در این مقاله یک روش جدید برای تغییر نرخ یادگیری و انطباق سیستم یادگیرنده با وضعیت محیط، برای استفاده در آتاماتای یادگیر پیشنهاد شده است. این روش جدید از برخی معیارهای آماری مربوط به توزیع فعلی به دست آمده برای بردار احتمالات متناظر با اقدام‌های آتاماتا به منظور تعیین افزایش یا کاهش نرخ یادگیری استفاده می‌کند. مزیت این روش در آن است که بر خلاف روش های موجود فعلی، در طول فرآیند یادگیری هم افزایش و هم کاهش مقدار نرخ یادگیری را-بسته به نتایج مقایسه معیارهای آماری- انجام می‌دهد و به صورت خودکار نرخ یادگیری را تنظیم می‌کند.

ضمن تشریح مبانی ریاضی این الگوریتم جدید، عملکرد این الگوریتم را در محیط‌های تصادفی نمونه بررسی کرده و با مقایسه نتایج به دست آمده نشان داده ایم، روش پیشنهادی جدید به دلیل اینکه در طول زمان یادگیری، همزمان و بر اساس معیارهای تعیین شده، افزایش و کاهش نرخ یادگیری را انجام می‌دهد، از انعطاف پذیری بیشتری نسبت به روش‌های قبلی برای انطباق با محیط‌های تصادفی پویا برخوردار است و مقادیر یادگرفته شده به مقادیر حقیقی نزدیکتر هستند.

***کلید واژه*- آتاماتای یادگیر، نرخ یادگیری پویا، تنظیم نرخ یادگیری، نابرابری چبیشف.**

۱- مقدمه

بحث تنظیم خودکار نرخ یادگیری یکی از مسائلی است که در مباحث مرتبط با الگوریتم ها و سیستم های یادگیری و خصوصا در مباحث مربوط به شبکه های عصبی به کرات مورد بررسی قرار گرفته است [۱] [۲][۳]. اکثر الگوریتم های موجود، از یک نرخ یادگیری با مقدار بالا شروع کرده و در حین فرآیند آموزش به کمک یک تابع کاهش یابنده با زمان (مثلا $(\alpha(t) = \alpha(0)e^{-\frac{t}{T}})$ آن را کاهش می‌دهند. به این ترتیب پس از مدتی نرخ یادگیری مقدار معمولا کوچکی دارد [۲].

مشکل از اینجاست که اگر از این مرحله به بعد، نمونه‌هایی وارد شوند که مشخصه های آماری متفاوتی با مشخصه های آماری نمونه های قبلی داشته باشند، سیستم یادگیر قادر به یادگیری آنها نیست.

بنابراین نرخ یادگیری کاهش یاینده با زمان، با وجود مزیت هایی که در همگرایی سریع دارد، باعث می‌شود سیستم یادگیر از قدرت تطابق پذیری و یادگیری خوبی در برخورد با محیط‌های تصادفی گوناگون برخوردار نباشد[۳].

کارهایی که در زمینه انطباقی کردن نرخ یادگیری در آتاماتای یادگیر انجام شده است را می‌توان به دو گروه تقسیم کرد: وابسته به مساله و مستقل از مساله. در روشهای پیشنهادی وابسته به مساله، بر اساس نوع مساله‌ای که آتاماتای یادگیر و یا اجتماعی از آتاماتاهای یادگیر درصدد حل آن هستند روشهایی برای تطبیقی کردن نرخ یادگیری پیشنهاد شده است[۴][۵][۶][۷]. در تمام این نمونه‌ها، بر حسب نوع مساله، روشی برای به روزرسانی نرخ یادگیری پیشنهاد شده است که سرعت همگرایی و حل مساله را افزایش داده است. رده وسیعی از این روش‌های انطباقی را در [۴] می‌توانید مشاهده کنید که در آن نویسنده با توجه به مساله یافتن درخت پوشای کمینه در گراف های تصادفی، از پاره ای ویژگیهای توزیع احتمال اقدام ها به شکل مستقم برای تطبیق کردن نرخ یادگیری و رسیدن به همگرایی سریع‌تر استفاده کرده است. مشابه همین ایده نیز در [۸] و [۶] مورد استفاده قرار گرفته است. اما در گروه دوم، مستقل از مساله یا ساختار اجتماع آتاماتاهای یادگیر، روشهایی برای این کار پیشنهاد می‌شود.

از آنجا که غالبا، استفاده از آتاماتاهای یادگیر به شکل شبکه‌ای ساختارمند صورت می‌گیرد [۹]، بنابراین روشهای پیشنهادی برای انطباقی کردن نرخ یادگیری غالبا وابسته به مساله و ساختار بوده و از مساله‌ای به مساله دیگر و از ساختاری به ساختار دیگر متفاوت هستند. پژوهش مستقلی که به بررسی تنظیم خودکار نرخ یادگیری درآتاماتای یادگیر (فارغ از مساله ای که آتاماتا در صدد حل آن است و یا ساختار شبکه‌ای از آتاماتاها که برای حل مساله مورد استفاده قرار می‌گیرد) پرداخته باشد، مشاهده نشده است. عموم پژوهش‌های این بخش مربوط به شبکه‌های

مقاله در تاریخ ۳۰ خرداد ماه ۱۳۹۱ دریافت شد.

محمدرضا ملاخلیلی میبدی، دانشکده مهندسی کامپیوتر، دانشگاه آزاد اسلامی واحد علوم و تحقیقات تهران، حصارک تهران (email: m.meybodi@srbiau.ac.ir)

محمدرضا میبدی، دانشکده مهندسی کامپیوتر و فنآوری اطلاعات، دانشگاه صنعتی امیرکبیر، تهران (email: mmeybodi@aut.ac.ir).

عصبی می‌باشند که برخی از آنها به لحاظ عمومیتی که در مبانی دارند- نظیر نرخ یادگیر کاهش‌یابنده با زمان- برای سایر سیستم‌های یادگیر و از جمله آتاماتای یادگیر نیز قابل استفاده هستند.

به نظر می‌رسد ایده های تطبیقی کردن نرخ یادگیری با هدف تسریع در همگرایی فرآیند یادگیری صورت گرفته است و مساله انطباقی کردن نرخ یادگیری در محیط‌های پویا چندان مدنظر قرار نگرفته است.

ادامه مقاله بدین صورت سازماندهی شده است. در بخش دوم ضمن بررسی مختصر آتاماتای یادگیر، نامساوی چبیشف را به عنوان یک نامساوی کاربردی در توزیع‌های تصادفی که مستقل از توزیع متغیر تصادفی رابطه‌ای میان مقدار متغیر تصادفی، مقدار میانگین و واریانس آن وضع می‌کند، را بررسی خواهیم کرد. در ادامه همین بخش ایده به کارگیری این نامساوی به عنوان یک معیار تشخیص پویایی محیط مورد بررسی قرار می‌گیرد. در بخش سوم، الگوریتم پیشنهادی مبتنی بر نامساوی چبیشف ارائه خواهد شد. بخش چهارم به بررسی تجربی این الگوریتم در تعدادی محیط پویا اختصاص داده شده است. بخش پنجم به جمع بندی مطالب ارائه شده در مقاله اختصاص یافته است.

۲- مبانی روش پیشنهادی

۲-۱ بررسی نحوه یادگیری در آتاماتای یادگیر

یک آتاماتای یادگیر به عنوان مدلی از یک سیستم یادگیر است که در محیط های تصادفی ناشناخته عمل می‌کند. آتاماتا در هر دور یک اقدام از میان مجموعه محدود اقدام های خود انتخاب کرده و با بررسی عکس العمل محیط نسبت به این اقدام، احتمال انتخاب اقدام های بعدی را بهبود می‌بخشد[۹].

شکل ۱- آتاماتای یادگیر و نحوه تعامل آن با محیط

یک آتاماتای تصادفی را می‌توان بصورت یک ماشین حالت متناهی در نظر گرفت. به بیان ریاضی نیز می‌توان آنرا بصورت یک پنج‌تایی مانند زیر نشان داد :

$$SA \equiv \{\alpha, \beta, F, G, \phi\} \qquad (۱)$$

که

تعداد اقدامهای آتاماتا	r
مجموعه اقدامهای آتاماتا	$\alpha \equiv \{\alpha_1, \alpha_2, ..., \alpha_r\}$
مجموعه ورودیهای آتاماتا	$\beta \equiv \{\beta_1, \beta_2, ..., \beta_r\}$
تابع نگاشت وضعیت فعلی و ورودی به وضعیت بعدی	$F \equiv \phi \times \beta \to \phi$
تابع خروجی که وضعیت فعلی را به خروجی بعدی نگاشت می‌کند.	$G \equiv \phi \to \alpha$
مجموعه وضعیتهای داخلی آتاماتا در زمان n	$\phi(n) \equiv \{\phi_1, \phi_2, ..., \phi_k\}$

مجموعه α شامل خروجیهای (اقدامهای) آتاماتا است که آتاماتا در هر گام یک اقدام از r اقدام این مجموعه را برای اِعمال بر محیط انتخاب می‌نماید. مجموعه ورودی‌ها (β) ورودیهای آتاماتا را مشخص می‌کند. توابع F و G وضعیت فعلی ورودی را به خروجی بعدی (اقدام بعدی) آتاماتا نگاشت می‌کنند. اگر نگاشتهای F و G قطعی باشند، آتاماتا یک آتاماتای قطعی نامیده می‌شود. در چنین حالتی با فرض یک وضعیت اولیه و ورودی مشخص، حالت بعدی و خروجی بصورت یکتا مشخص شده‌اند. در حالتیکه نگاشتهای F و G تصادفی باشند، آتاماتا بعنوان آتاماتای تصادفی معرفی می‌شود.

آتاماتای تصادفی را می‌توان به دو دسته آتاماتای تصادفی با ساختار ثابت و آتاماتای تصادفی با ساختار متغیر تقسیم‌بندی کرد. در آتاماتای تصادفی با ساختار ثابت احتمال اقدامهای آتاماتا ثابت هستند. درحالیکه در آتاماتای تصادفی با ساختار متغیر احتمالات اقدامهای آتاماتا در هر تکرار به‌روز می‌شوند (تغییر احتمالهای اقدامها بر اساس الگوریتم یادگیری انجام می‌شود). وضعیت داخلی آتاماتا ϕ، توسط احتمالات اقدامهای آتاماتا بازنمایی می‌شوند. برای سادگی هر وضعیت داخلی آتاماتا را مطابق با یک اقدام مشخص آتاماتا در نظر می‌گیرند. بنابراین می‌توان وضعیت داخلی آتاماتا ϕ را با بردار احتمال اقدامهای آتاماتا P که بصورت زیر نشان داده می‌شود، جایگزین نمود.

$$P(n) \equiv \{p_1(n), p_2(n), ..., p_r(n)\} \qquad (۲)$$

به گونه ای که:

$$\sum_{i=1}^{r} p_i(n) = 1, \quad \forall n \quad , \quad p_i(n) = \text{Prob}[\alpha(n) = \alpha_i] \qquad (۳)$$

در آغاز فعالیت آتاماتا، احتمال اقدامهای آن بصورت مساوی با هم برابر با $p_i = \frac{1}{r}$ قرار داده می‌شوند. (که r تعداد اقدامهای آتاماتا می‌باشد.)

محیط: محیط تصادفی را به‌طور ریاضی می‌توان بصورت یک سه‌تایی $E \equiv \{\alpha, \beta, c\}$ توصیف کرد، به‌طوریکه:

مجموعه ورودیهای محیط	$\alpha \equiv \{\alpha_1, \alpha_2, ..., \alpha_r\}$
مجموعه خروجیهای محیط	$\beta \equiv \{\beta_1, \beta_2, ..., \beta_r\}$
مجموعه احتمالهای جریمه	$c \equiv \{c_1, c_2, ..., c_r\}$

ورودی محیط یکی از r اقدام انتخاب شده آتاماتا است. خروجی(پاسخ) محیط به هر اقدام i توسط β_i مشخص می شود [۱۰]

ارتباط آتاماتای تصادفی با محیط در شکل ۱ نشان داده شده است. از این مجموعه به همراه الگوریتم یادگیری تحت عنوان آتاماتای یادگیر تصادفی نام برده می‌شود. به این ترتیب آتاماتای یادگیر تصادفی را

می‌توان با چهارتایی

$$SLA \equiv \{\alpha, \beta, p, T, c\} \quad (۴)$$

تعریف کرد. به طوریکه :

مجموعه اقدامهای آتاماتا/مجموعه ورودیهای محیط

$$\alpha \equiv \{\alpha_1, \alpha_2, ..., \alpha_r\}$$

مجموعه ورودیهای آتاماتا/ مجموعه خروجیهای محیط

$$\beta \equiv \{\beta_1, \beta_2, ..., \beta_r\}$$

بردار احتمال اقدامهای آتاماتا

$$p \equiv \{p_1, p_2, ..., p_r\}$$

الگوریتم یادگیری

$$T \equiv p(n+1) = T[\alpha(n), \beta(n), p(n)]$$

مجموعه احتمالات جریمه که معرف محیط می‌باشند.

$$c \equiv \{c_1, c_2, ..., c_r\}$$

الگوریتم یادگیری خطی: الگوریتم یادگیری یک رابطه بازگشتی است که برای انجام تغییرات و به روزرسانی در بردار احتمال اقدام های آتاماتا در یک آتاماتای یادگیر تصادفی با ساختار متغیر مورد استفاده قرار می‌گیرد. فرض کنید یک آتاماتای یادگیر تصادفی ساختار متغیر در زمان k از میان مجموعه اقدامهای α عمل $\alpha_i(k)$ را انتخاب کرده باشد. همچنین فرض کنید بردار احتمال انتخاب اقدامهای آتاماتا را با $\mathbf{p}(k)$ نمایش داده‌ایم. اگر a وb پارامترهایی باشند که به ترتیب میزان افزایش یا کاهش احتمالات اقدام ها را مشخص می‌کنند و r تعداد اقدامهای قابل انجام توسط آتاماتای یادگیر باشد، بردار $\mathbf{p}(k)$ توسط الگوریتم یادگیری خطی ارائه شده در روابط زیر به روزرسانی می‌شود. مقدار a را پارامتر پاداش و b را پارامتر جریمه می نامند

$$p_j(k+1) = \begin{cases} (1-a)p_j(k) + a & j = i \\ (1-a)p_j(k) & \forall j \neq i \end{cases} \quad (۵)$$

$$p_j(k+1) = \begin{cases} (1-b)p_j(k) & j = i \\ (1-b)p_j(k) + \frac{b}{r-1} & \forall j \neq i \end{cases} \quad (۶)$$

رابطه (۵) زمانی مورد استفاده قرار می گیرد که عمل $\alpha_i(k)$ منجر به دریافت پاداش از محیط شده باشد و رابطه (۶) زمانی مورد استفاده قرار می گیرد که این عمل به دریافت جریمه از محیط منجر شده باشد.

اگر a=b باشد روابط یادگیری خطی (معادله‌های (۶)و(۵)) را الگوریتم L_{R-P} می‌نامند. اگر $a \gg b$ باشد آن را $L_{R-\varepsilon P}$ و اگر b=0 باشد آن را L_{R-I} می‌نامند[۱۱].

عامل موثر در کارآیی آتاماتای ساختار متغیر، الگوریتم های یادگیری هستند که برای به روزرسانی احتمال اقدام ها استفاده می‌شود.

با این مقدمات فرض کنید یک آتاماتای یادگیر با r اقدام در یک محیط تصادفی فعالیت می‌کند. محیط تصادفی اقدام انجام شده توسط آتاماتا را ارزیابی می‌کند و آتاماتا بر اساس این ارزیابی، بردار احتمالات اقدام های خود را به روزرسانی می‌کند. . فرض کنید P_i^t مقدار احتمال انتخاب اقدام i ام آتاماتا را در زمان t نشان دهد. ضمنا

$$\sum_{i=1}^{r} P_i^t = 1 \quad \forall\, t \in \{0,1, \dots \dots\} \quad (۷)$$

بردار احتمال انتخاب اقدام های آتاماتا در زمان t را با $\overline{P(t)}$ نشان میدهیم که $\overline{P(t)} = [P_1^t, P_2^t, \dots \dots . P_r^t]$

علاوه بر این محیط تصادفی بر اساس بردار احتمالات $\bar{C} = [C_1, C_2, \dots \dots \dots, C_r]$ اقدام های محیط را پاداش میدهد که در آن $\sum_{i=1}^{r} C_i = 1$ است.

در این آتاماتای یادگیر هر اقدام آتاماتا دارای یک نرخ یادگیری است. این بردار را با $\bar{\alpha} = [\alpha_1, \alpha_2, \dots \dots \dots, \alpha_r]$ نشان میدهیم. در این مقاله فرض می‌کنیم آتاماتای یادگیر برای به روزرسانی بردار احتمال انتخاب های خود از الگوریتم L_{R-I} استفاده می‌کند.

گزاره ۱: رابطه به روزرسانی بردار احتمال انتخاب اقدام آتاماتای یادگیر برای زمانی که اقدام k ام توسط آتاماتا صورت گرفته و آتاماتا از الگوریتم یادگیر خطی L_{R-I} استفاده می‌کند را می‌توان به صورت رابطه (۸) نوشت:

$$P_k^{t+1} = P_k^t * (1 - \alpha_k) + \alpha_k R^{t+1} \quad (۸)$$

در رابطه(۸)، R پاسخ محیط به اقدام انجام شده توسط آتاماتا است و داریم

$$R^{t+1} = \begin{cases} 1 & with\ probability \quad C_k \\ 0 & with\ probability \quad 1 - C_k \end{cases} \quad (۹)$$

رابطه یادگیری آتاماتا در این حالت را به شکل برداری (۱۰) میتوان نشان داد

$$\vec{\mathbf{P}}^{t+1} = \vec{\mathbf{P}}^{t} * (1 - \alpha) + \alpha \vec{\mathbf{R}}^{t+1} \quad (۱۰)$$

رابطه (۱۰) در حقیقت نشان میدهد که بردار احتمال انتخاب اقدام های آتاماتای یادگیر در طول فرآیند یادگیری، به بردار میانگین پاسخ های محیط همگرا می‌شود.

۲-۲ محیطهای پویا

موفقیت آتاماتای یادگیر در محیطهای پویا بستگی به تغییرات محیط و اطلاعاتی که توسط آتاماتای یادگیر از محیط قابل جمع آوری است ، دارد. در [۹] نویسنده یک تقسیم بندی مفصل از محیطهای پویا ارائه داده است. اما از یک دیدگاه کلی می‌توان محیطهای پویا را به دو گروه کلی تقسیم کرد: محیطهای پویا MSE و محیطهای پویا با تابع احتمال جریمه متغیر با زمان.

در MSEها فرض بر این است که محیط پویای واقعی خود از چند محیط ایستا[۱] تشکیل شده است و پویایی محیط ناشی از یک توالی از فرآیندهای جابجایی بین این محیطهای ایستا است[۱۲]. به بیان ریاضی، محیط پویای ξ توسط یک مجموعه$\{E_1, E_2, \dots \dots, E_H\}$ از محیطهای ایستا و یک ماتریس جابجایی T تعریف می‌شود. در این ماتریس T ، عنصر $T_{i,j}$ نشان دهنده احتمال آن است که اگر آتاماتای یادگیر در حال حاضر با محیط تصادفی E_i در تعامل است، در گام بعدی با محیط تصادفی E_j در تعامل باشد. در حقیقت در MSEها محیط تصادفی پویا از

[۱] در این مقاله محیط ایستا را معادل با Stationary Environment و محیط پویا را برای non-Stationary Environment در نظر گرفته ایم

مجموعه‌ای از محیطهای تصادفی ایستا تشکیل شده است که مجموعه حالات یک زنجیره مارکوف را تشکیل می‌دهند[۹] [۱۳][۱۲].

مدل دیگری که در متون مربوطه برای محیطهای پویا ارائه شده است، مدلی است که احتمالات جریمه اقدامها را غیرثابت و متغیر با زمان فرض کرده است[۱۴][۱۳]. یعنی احتمال جریمه اقدام در عین اینکه به اقدام انجام شده بستگی دارد، به زمان انجام آن نیز بستگی دارد.

مقالاتی که به ارائه الگوریتم های یادگیر مبتنی بر آتاماتای یادگیر برای محیطهای MSE پرداخته‌اند به دلیل تنوع این محیط ها بسیار زیاد هستند. ایده غالب آنها تشکیل سلسله مراتبی از آتاماتاهای یادگیر است که برخی سطوح وظیفه تعیین محیط ایستا را بر عهده دارند و برخی سطوح دیگر فرآیند یادگیری در آن محیط ایستا را بر عهده دارند[۱۳].

همانگونه که قبلا هم متذکر شدیم، هیچ پژوهش مستقلی که فرآیند یادگیری در آتاماتای یادگیر را از طریق تطبیقی کردن نرخ یادگیری دنبال کرده باشد، در متون ومقالات تخصصی این حوزه یافت نشد و آنچه که از طریق تطبیقی کردن نرخ یادگیری صورت می‌گیرد، غالبا وابسته به مساله و ساختاری از آتاماتاها که برای حل مساله مورد استفاده قرار گرفته است، می‌باشد.

۲-۳ نابرابری چبیشف

برای تنظیم پویای نرخ یادگیری، به منظور انطباق در محیطهای پویا، نیاز به معیاری داریم تا بر اساس آن معیار میزان توانمندی نرخ یادگیری فعلی سیستم یادگیر را در رصدکردن تغییرات محیط بسنجیم. یکی از ابزارهای آماری مناسب در این مورد نابرابری موسوم به نابرابری چبیشف است. در نظریه احتمالات نابرابری چبیشف تضمین می‌کند که در هر نمونه تصادفی یا در هر توزیع احتمال، "تقریبا تمامی" مقادیر در نزدیکی میانگین خواهند بود. به طور دقیق تر این قضیه بیان می‌کند که حداکثر مقادیری که در هر توزیع می‌توانند بیش از k برابر انحراف معیار با میانگین فاصله داشته باشند $\frac{1}{k^2}$ است.[۱۵]

قضیه (نامساوی مارکوف): اگر X یک متغیر تصادفی و a یک عدد حقیقی مثبت باشد، در این صورت

$$Pr(|X| > a) \leq \frac{E(X^2)}{a^2} \qquad (۱۱)$$

برای اثبات این قضیه میتوانید به مرجع[۱۶] مراجعه کنید.

با استفاده از نامساوی مارکوف میتوان به نامساوی چبیشف رسید. اگر m نشان دهنده میانگین متغیر تصادفی X باشد با جایگذاری X-m در رابطه بالا به نامساوی، موسوم به نامساوی چبیشف خواهیم رسید.

$$Pr(|X - m| \geq a) \leq \frac{Var(X)}{a^2} \qquad (۱۲)$$

اگر Var(X) را با δ نشان دهیم تفسیر دیگری از نامساوی چبیشف به دست می‌آید

$$Pr(|X - m| \geq a\delta) \leq \frac{1}{a^2} \qquad (۱۳)$$

رابطه (۱۳) در حقیقت بیان می‌کند که احتمال اینکه یک متغیر تصادفی در خارج از بازه ای حول میانگین به شعاع aبرابر واریانس باشد از $\frac{1}{a^2}$ کمتر است. این رابطه برای مقادیر a>1 واجد اطلاعات مفید است.

نابرابری چندبعدی چبیشف تعمیمی از نابرابری چبیشف است که به کمک آن میتوان مرزی را برای اینکه یک بردار تصادفی از بردار مقادیر میانگین اش بیش از یک مقدار معین فاصله داشته باشد را تعیین کرد. این نابرابری بدین شکل فرمول بندی می‌شود.

نابرابری چبیشف (تعمیم یافته): فرض کنید $\mathbf{X}$ یک بردار تصادفی با میانگین $\boldsymbol{\mu} = E[\boldsymbol{X}]$ و ماتریس کوواریانس $\boldsymbol{V} = E[(X-\mu)(X-\mu)^T]$ باشد. اگر $\mathbf{V}$ یک ماتریس تعریف شده مثبت باشد در این صورت برای هر عدد حقیقی $t > 0$ داریم

$$\Pr\left((X-\mu)V^{-1}(X-\mu)^T\right) > qN) \leq 1/q^2 \qquad (۱۴)$$

که در رابطه (۱۴) $N = trace(V^{-1}V)$ است.

۲-٤ استفاده از نابرابری چبیشف به عنوان یک معیار در تطبیقی کردن نرخ یادگیری

در این قسمت با توجه به گزاره ۱ که نحوه تغییرات بردار احتمال انتخاب اقدام های آتاماتا را نشان میدهد و نابرابری چبیشف به ارائه معیاری برای کاهش یا افزایش نرخ یادگیری آتاماتای یادگیر خواهیم پرداخت

گزاره ۲: فرض کنید X متغیری است که بر اساس رابطه (۱۵) به روز رسانی می‌شود

$$X^{t+1} = X^t * (1-\alpha) + \alpha R^{t+1} \qquad (۱۵)$$

که درآن

$$R^{t+1} = \begin{cases} 1 & with\ probability \quad C^* \\ 0 & with\ probability \quad 1-C^* \end{cases} \qquad (۱۶)$$

است. ثابت می‌شود[۱۶]:

الف- $lim_{t\to\infty} E[X^t] = \mu = C^*$

ب- واریانس $\mathrm{X^t}$ محدود و دارای مقدار حدی زیر است

$$Var(X^t) = \delta^2 = \frac{\alpha}{2-\alpha} C^*(1-C^*)$$

ج- مطابق با نامساوی چبیشف داریم:

$$\forall q > 0 \qquad P(|X^t - C^*| \geq q\delta) \leq \frac{1}{q^2}$$

اثبات:

الف-$\langle R^{t+1}\rangle$ یک فرآیند تصادفی برنولی با پارامتر $\mathrm{C^*}$ است. علاوه بر این داریم:

$$E[X^{t+1}] = (1-\alpha)E[X^t] + \alpha E[R^{t+1}] \qquad (۱۷)$$
$$= (1-\alpha)E[X^t] + \alpha C^*$$

از رابطه بازگشتی (۱۷) داریم:

$$E[X^{t+1}] = (1-\alpha)^t E[X^0] + (1-(1-\alpha)^t).C^* \quad (۱۸)$$

با توجه به اینکه $0 < \alpha < 1$ رابطه (۱۸) نشان می‌دهد که

$$\lim_{t\to\infty} \mathrm{E[X^{t+1}]} = \mathrm{C^*} \qquad (۱۹)$$

ب: به طریق مشابه میتوان نشان داد که

$$Var[X^{t+1}] = \frac{\alpha}{2-\alpha} C^*(1-C^*)(1-(1-\alpha)^{2t+2}) \quad (۲۰)$$

با توجه به اینکه $0 < \alpha < 1$ رابطه (۲۰) نشان میدهد که

$$lim_{t\to\infty}\ Var[X^{t+1}] = \frac{\alpha}{2-\alpha} C^*(1-C^*) \qquad (۲۱)$$

ج- با فرض $m = E[X]$ برای متغیر تصادفی X نامساوی مارکوف

$$P(|X| \geq a) \leq \frac{E(X^2)}{a^2}$$

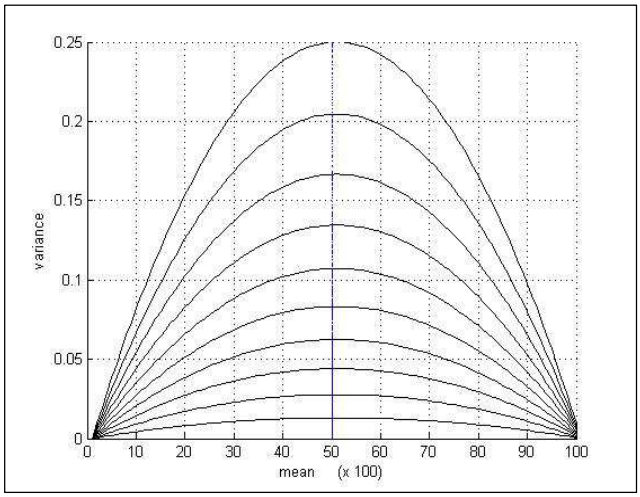
به صورت رابطه (۲۲) تبدیل خواهد شد

$$P(|X - m| \geq a) \leq \frac{Var(X)}{a^2} \qquad (۲۲)$$

با جایگذاری $a = q.\sqrt{Var(\mathrm{X^t})} = q.\delta$ و $m = E[\mathrm{X^t}] = \mathrm{C^*}$ در مورد متغیر تصادفی $X = \mathrm{X^t}$ در رابطه (۲۲) نتیجه ج حاصل خواهد شد ■

گزاره ۲ چند نکته را نشان میدهد:

اولا مقدار α کنترل کننده نرخ همگرایی است. روابط بازگشتی مربوط به میانگین و واریانس نشان میدهند که اگر α برابر یا ۱ باشد سرعت همگرایی بیشینه است. اگرچه مقدار α برابر با ۱ سرعت همگرایی بالایی دارد اما منجر به واریانس بزرگتری می‌شود(شکل ۲). مقادیر α کوچک تر گرچه همگرایی کندتری را سبب می‌شوند، اما جواب نزدیکتری به مقدار احتمال واقعی دارند. نکته حائز اهمیت در آن است که واریانس به صفر نمی‌رسد.



شکل ۲- رابطه میان واریانس با میانگین و نرخ یادگیری

دوم اینکه گرچه وجود واریانس ناخوشایند به نظر می‌رسد، اما یک مقدار محدود و کوچک واریانس باعث می‌شود با مشاهده داده های جدید، الگوریتم از گیرافتادن در نقاط بیشینه محلی نجات یابد. علاوه بر اینکه باعث می‌شود قابلیت انطباق با محیط های متغیر را نیز پیدا کند. سرعت این انطباق پذیری به کمک α یا همان نرخ یادگیری قابل کنترل است. ضمنا گیر نیفتادن در نقاط بیشینه محلی، تضمین نشده است و به مقدار α و شکل تابع توزیع بستگی دارد.

سوم اینکه نامساوی چبیشف یک بازه اطمینان از مقدار احتمالی تخمین زده شده توسط آتاماتا نسبت به واریانس تخمین به دست می‌دهد. این ویژگی می‌تواند به انتخاب یک α مناسب کمک کند.

تمام الگوریتم هایی که سعی می‌کنند به گونه ای پویا نرخ یادگیری را تنظیم کنند، از این نکته استفاده می‌کنند که در زمانی که همگرایی رخ میدهد، نرخ یادگیری را کاهش می‌دهند. به بیانی دیگر، چنانچه اختلاف میان احتمال تخمین جدید و میانگین تخمین های قبلی بزرگ باشد، نرخ یادگیری افزایش می‌یابد. این حالت زمانی رخ می‌دهد که یک بیشینه محلی وجود داشته باشد و یا محیط تصادفی، تغییر کند. در الگوریتم پیشنهادی جدید از ویژگی سوم مطرح در گزاره ۲ به عنوان عامل تشخیص دهنده اختلاف میان تخمین جدید ومیانگین تخمین های قبلی استفاده می‌کنیم و به کمک آن در موارد لازم، نرخ یادگیری را افزایش میدهیم.

در اکثر سیستم های یادگیر و از جمله در آتاماتای یادگیر، نرخ یادگیری در مدت آموزش، به شکل پویا –کاهش یابنده با زمان– تغییر می‌کند. دلیل این امر هم واضح است. در ابتدای یادگیری ، مقادیر بزرگتر نرخ یادگیری، باعث آموزش سریع تر می‌گردند. به تدریج و با افزایش تعداد نمونه‌ها، سیستم یادگیرنده، سعی می‌کند بیشتر متکی بر تجربیات آموزشی گذشته باقی بماند تا اینکه سعی کند از نمونه های جدید برای یادگیری استفاده کند. این منطق باعث می‌شود، در ابتدای فرآیند یادگیری، سیستم یادگیرنده جسورانه تر و با گذر زمان محافظه کارانه تر عمل کند.

اکثر سیستم‌های یادگیری، از یک نرخ یادگیر پویای کاهش یابنده در حین فرآیند یادگیری استفاده می‌کنند. نرخ یادگیری با این مفهوم پارامتری است که میزان فراموش‌کاری سیستم را تعریف می‌کند. مقادیر کوچکتر این پارامتر، یعنی اتکای بیشتر سیستم به تجربیات گذشته و مقادیر بزرگتر به معنای نادیده گرفتن تجربیات گذشته است.

در بیشتر سیستم‌ها، منطق بالا پاسخگوی نیازها می‌باشد. مساله اینجاست که کاهش نرخ یادگیری در طول زمان باعث می‌شود به مرور زمان، انطباق پذیری سیستم یادگیر کاهش یابد و نسبت به تغییرات در محیط عکس‌العمل مناسب نشان ندهد. بنابراین برای تنظیم اتوماتیک نرخ یادگیری در طول مدت آموزش بایستی معیاری داشته باشیم تا بر اساس آن نسبت به افزایش (در صورت بروز تغییرات جدی در محیط) یا کاهش (در صورت یکنواخت بودن پاسخ محیط و نزدیک شدن به همگرایی) نرخ یادگیری اقدام کنیم. این معیار بایستی قادر به تعیین میزان اهمیت تغییرات در محیط باشد.

۳- الگوریتم پیشنهادی

با توجه به رابطه یادگیری مورد استفاده توسط آتاماتای یادگیر برای هر اقدام و تفسیر برداری آن، می‌توان از نابرابری چبیشف به شکل ساده یا برداری آن برای تنظیم اتوماتیک نرخ یادگیری استفاده کرد. منطق کار بدین صورت است که :

الف– درحالت ساده: در این حالت برای هر یک از اقدام‌های آتاماتا یک نرخ یادگیری در نظر می‌گیریم. بر اساس نامساوی چبیشف (یا همان مارکوف رابطه (۱۳))، چنانچه اختلاف مقدار احتمال انتخاب یک اقدام آتاماتا با میانگین مقادیر قبلی آن از یک مقدار آستانه بیشتر باشد (نامساوی چبیشف)، به منزله تغییرات زیاد درمحیط بوده و نرخ یادگیری را افزایش می‌دهد. همزمان با افزایش نرخ یادگیری در هر بار، مقدار میانگین با صفر مقداردهی شده تا الگوریتم قابلیت گریز از بیشینه‌های محلی را داشته باشد (گرچه همان‌طور که توضیح دادیم، این فرار از بیشینه‌های محلی تضمین شده نیست)

در این الگوریتم برای اجتناب از سربار محاسباتی، از یک کران بالا برای مقدار واریانس استفاده می‌کنیم. برای رسیدن به این کران بالا قسمت ب گزاره ۲ و اثبات آن را در نظر بگیرید. نشان داده شد که مقدار واریانس در زمان t+1 برابر است با:

$$Var[X^{t+1}] = \frac{\alpha}{2-\alpha} C^*(1-C^*)(1-(1-\alpha)^{2t+2}) \quad (۲۳)$$

رابطه (۲۳) نشان میدهد که واریانس تابعی از میانگین (C^*)و نرخ یادگیری (α) است. شکل ۲ رابطه میان واریانس و میانگین را به ازای نرخ های مختلف یادگیری نشان می‌دهد. همانگونه که این شکل نیز نشان می‌دهد، هرچقدر میانگین به صفر یا یک نزدیک‌تر باشد، واریانس کوچک‌تر است. برعکس در میانگین برابر با ۰٫۵ واریانس، بیشینه مقدار را دارد. علاوه بر این، می‌توان دید که هرچقدر نرخ یادگیری به ۱ نزدیکتر باشد (نمودار بالایی) واریانس مقدار بیشتری دارد و بالعکس، در مقادیر کوچکتر نرخ یادگیری (پایین ترین نمودار)، واریانس کمتری داریم. برای سهولت در محاسبه واریانس، از یک کران بالا برای آن استفاده می‌کنیم. بدین صورت که مقدار واریانس به ازای میانگین ۰٫۵ را در محاسبات در نظر می‌گیریم. بدین ترتیب محاسبه واریانس در هر دور تنها تابعی از نرخ یادگیری خواهد بود. این کران بالا با مقدار q کوچکتر در الگوریتم جبران می‌شود.

Proposed Algorithm L_{R-I} (1)
1: **Parameters**: Real TSH<<1, q>1, $\vec{\alpha}$ Learning Rate Vector,
2: **Initialization:** $p_j \leftarrow \frac{1}{K}, \mu_j \leftarrow \frac{1}{K}$, $\delta_j \leftarrow 0$, $t_j \leftarrow 0$ **for** j ← 1 **to** K
3: **loop**
4 Draw randomly an action $\boldsymbol{i}$ according to probabilities $p_0, \ldots, p_K$
5: Receive either *reward* or *penalty*
6: **if** *reward* **then**
7: for $\boldsymbol{j} \leftarrow$ **1** ***to K*** do
8: if $\boldsymbol{j} \neq \boldsymbol{i}$ then
9: $p_j \leftarrow (1-\alpha_i)p_j$ //penalt
10: **else**
11: $p_i \leftarrow p_i + \alpha_i(1-p_i)$ //reward
12: **end if**
13: **end for**
14: **for** $\boldsymbol{j} \leftarrow$ **1** ***to K*** do
15: update (μ_j)
16: update (δ_j)
17: **if** $|p_j - \mu_j| > q\delta_j$
18: increment (α_j); $t_j \leftarrow 0$; reset μ_j
19: **else if** $(1-\alpha_j)^{t_j} <$ TSH
20: decrement (α_j); $t_j \leftarrow 0$
21: **else** $t_j \leftarrow t_j + 1$
22: **endif**
23: **end for**
24: **endif**
25: **end loop**

شکل ۳: الگوریتم شماره ۱ مبتنی بر نامساوی چبیشف در حالت ساده

برای محاسبه میانگین نیز از یک میانگین گیری روی مقادیر مربوط به احتمالات استفاده می‌کنیم. هر زمان که نرخ یادگیری افزایش می‌یابد، میانگین های قبلی را در نظر نگرفته و میانگین گیری را روی مقادیر جدید آغاز می‌کنیم. این کار باعث گریز از بیشینه های محلی می‌شود. برای افزایش و کاهش مقدار نرخ یادگیری آنها را در یک مقدار ثابت بزرگتر از ۱ ضرب (برای افزایش) یا تقسیم (برای کاهش) می‌کنیم.

با این اصلاحات الگوریتم نهایی پیشنهادی، در حالت ساده آن به شکل ۳ خواهد بود.

ب- در حالت برداری: در این حالت، بردار احتمال انتخاب اقدام‌های آتاماتای یادگیر را در نظر گرفته و به این صورت بر خلاف حالت قبل، آتاماتای یادگیر یک نرخ یادگیری دارد که بر اساس نامساوی چبیشف در حالت برداری، مقدار آن تنظیم می‌شود. هر زمان که بردار جدید یادگرفته شده توسط آتاماتای یادگیر، بیش از یک میزان مشخص از بردار میانگین مقادیر قبلی فراگرفته شده توسط آتاماتا، فاصله داشته باشد، به معنای وجود تغییرات گسترده در محیط است. برای اینکه بتوان اثر این تغییرات را منعکس کرد آتاماتا بایستی نرخ یادگیری را افزایش دهد. ملاک سنجش میزان تغییرات نابرابری چبیشف برداری است. همانند حالت ساده، در اینجا نیز با هربار افزایش مقدار نرخ یادگیری برای گریز از بیشینه محلی، مقدار میانگین با صفر مقداردهی می‌شود.

به این ترتیب الگوریتم پیشنهادی جدید در حالت برداری آن به شکل ۴ خواهد بود

Proposed Algorithm L_{R-I} (2)
1: **Parameters**: Real q>1, α < 1 Learning Rate
2: **Initial:** $p_j \leftarrow \frac{1}{K}, \mu_j \leftarrow p_j$, $dt \leftarrow 0$, $t \leftarrow 0$ **for** j ← 1 **to** K
3: **loop**
4: Draw randomly an action $\boldsymbol{i}$ according to probabilities $p_0, \ldots, p_K$
5: Receive either *reward* or *penalty*
6: **if** *reward* **then**
7: **for** $\boldsymbol{j} \leftarrow$ **1** ***to K*** **do**
8: **if** $\boldsymbol{j} \neq \boldsymbol{i}$ **then**
9: $p_j \leftarrow (1-\alpha)p_j$ //penalt
10: **else**
11: $p_i \leftarrow p_i + \alpha(1-p_i)$ //reward
12: **end if**
13: **end for**
14: $t \leftarrow t+1$;
15: $dt \leftarrow dt+1$;
16: $\vec{\mu} = mean(\vec{P}(i-dt-1), \ldots, \vec{P}(i))$
17: $\vec{V} = Cov(\vec{\mu}, \vec{P})$
18: $N = trace(\vec{V}\,\vec{V}^{-1})$
19: **if** $(\vec{P}-\vec{V})^T \vec{V}^{-1}(\vec{P}-\vec{V}) > qN$
20: increment (α); $t \leftarrow 0$; $dt \leftarrow 0$;
21: **else if** $(1-\alpha)^t <$ **TSH**
22: decrement (α); $t \leftarrow 0$;
23: **end if**
24: **end if**
25: **end loop**

شکل ٤: الگوریتم پیشنهادی جدید مبتنی بر نامساوی چبیشف به شکل برداری

در هردو الگوریتم پیشنهادی مبتنی بر نابرابری چبیشف، اگر با نرخ یادگیری α پس از t بار، مقدار $(1-\alpha)^t$ از یک مقدار آستانه ثابت کمتر باشد، بیانگر آن است که واریانس به مقدار حدی خود- به ازای آن نرخ یادگیری خاص- نزدیک شده است. این نشان‌دهنده ثبات در محیط است و میتوان با کاهش نرخ یادگیری، اثر پذیری آتاماتا از نمونه‌های ورودی جدید را کاهش داد. دلیل انتخاب این ملاک برای کاهش نرخ یادگیری به قسمت ب اثبات گزاره ۲ برمی‌گردد که واریانس را به شکل تابعی از میانگین واقعی نشان میدهد. بر اساس آنچه که در قسمت ب گزاره ۲

دیدیم مقدار واریانس در زمان تابعی از $(1-\alpha)^{2t}$ است که در آن α نرخ یادگیری و t تعداد دفعاتی است که نرخ یادگیری بدون تغییر برای آموزش مورد استفاده قرار گرفته است. بدین ترتیب الگوریتم میتواند از $(1-\alpha)^{t}$ به عنوان معیاری استفاده کند که میزان ثبات در محیط را نشان می‌دهد و در حقیقت به جای آن که منتظر بماند تا بر اثر افزایش مقدار t مقدار $Var[X^{t+1}]$ به مقدار حدی $\frac{\alpha}{2-\alpha}C^*(1-C^*)$ نزدیک شود، با کاهش مقدار نرخ یادگیری α این فرآیند تسریع شود

٤- بررسی نتایج شبیه سازی:

برای بررسی نحوه عملکرد روش پیشنهادی جدید، یک آتاماتای تصادفی با ۲ اقدام درنظر گرفته ایم که توسط یک محیط تصادفی مورد ارزیابی قرار می‌گیرد. محیط تصادفی، یک محیط پویا در نظر گرفته شده است. بدین صورت که تابع ارزیابی اقدام آتاماتا توسط محیط یک مقدار ثابت فرض نشده و در طول زمان آموزش تغییر می‌کند. الگوریتم مورد استفاده توسط آتاماتای یادگیر الگوریتم L_{R-I} و L_{R-I} های پیشنهادی است.

برای انجام شبیه‌سازی، توابع مختلفی بر حسب زمان به عنوان تابع پاداش یا جریمه محیط درنظر گرفته‌ایم. برای هر محیط، یک آتاماتا را هم‌زمان به دو شیوه آموزش داده‌ایم. روش اول، همان روش معمول مورد استفاده در آتاماتای یادگیر است به این صورت که با توجه به پویایی محیط، آتاماتای یادگیر از یک نرخ یادگیری کوچک –بین 0.1 و 0.2 – آغاز کرده و با کاهش آن به اندازه 0.01 در هر دور اجرای الگوریتم (تا رسیدن به آستانه 0.001) فرآیند یادگیری را انجام داده است. روش دوم مبتنی بر الگوریتم‌های پیشنهادی جدید است. الگوریتم‌های پیشنهادی جدید نیز از یک نرخ یادگیری دلخواه (و غالبا بزرگ نزدیک به ۱) آغاز می‌کنند. نتایج هر دو الگوریتم پیشنهادی در مقایسه با روش معمول، در ادامه آورده شده و نتایج آن شرح داده شده است. معیار مقایسه، میزان انطباق بردار آموزش داده شده در روش جدید پیشنهادی و روش معمول با بردار ارزیابی محیط یا همان احتمال جریمه اقدام انجام شده توسط آتاماتا است (که قاعدتا مقدار ثابتی نداشته و بر حسب زمان در تغییر است). از نرم (norm) بردار احتمال انتخاب اقدام های یادگرفته شده و بردار احتمالی واقعی به عنوان معیاری جهت مقایسه میزان انطباق پذیری آتاماتا با محیط استفاده کرده ایم. نتایج را برای محیط‌های پویای مثال، در ادامه مشاهده می‌کنید.

٤-۱ بررسی نتایج الگوریتم پیشنهادی ۱

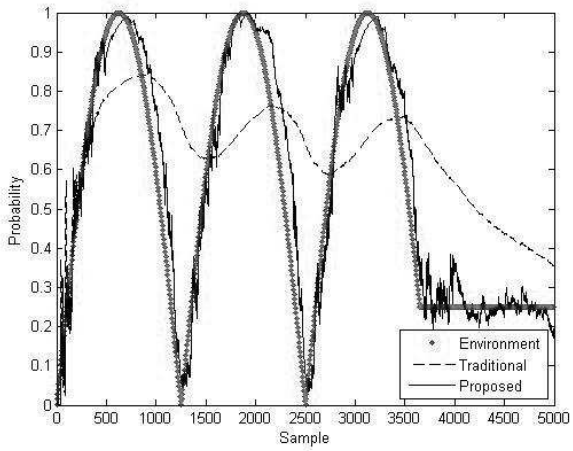
در اینگروه از آزمایش ها از الگوریتم شماره ۱ (شکل ۳) استفاده شده است.

آزمایش ۱: در اولین نمونه، از تابع رابطه (۲۴) به عنوان تابع ارزیابی اقدام شماره ۱ آتاماتا استفاده کرده ایم:

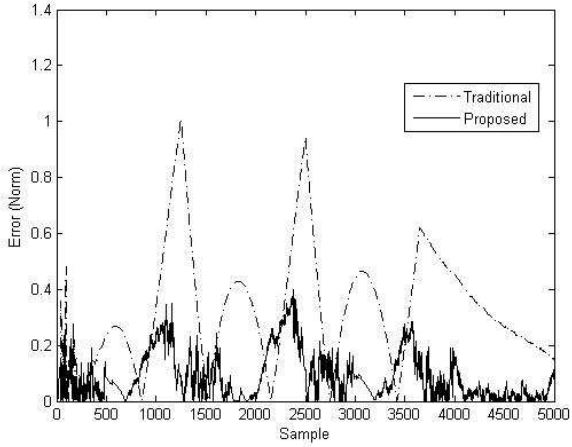
$$f(i)=\begin{cases}\left|\sin\left(\frac{4i\pi}{n}\right)\right| & i<0.73n\\ \left|\sin\left(\frac{4t\pi}{n}\right)\right| & t=0.73, i\geq 0.73n\end{cases} \quad (۲۴)$$

در رابطه(۲۴)،i معرفi امین نمونه و n نشاندهنده تعداد کل نمونه ها است. برای بررسی عملکرد روش جدید و مقایسه آن، همزمان آتاماتا را با الگوریتم ۱ و نیز الگوریتم L_{RI} با نرخ یادگیری کوچک $\alpha=0.002$ و تکنیک سرد کردن تدریجی با ضریب کاهش 0.001 در هر دور، آموزش داده‌ایم ($\alpha^{t+1}=0.999*\alpha^{t}$). نتیجه مقایسه این دو درشکل ۵ آورده شده است. شکل ۶ مقدار نرم (فاصله) هر یک از دو بردار آموزش داده شده به روش معمول و روش پیشنهادی با بردار واقعی را نشان می‌دهد. شکل ۷ نحوه تغییر (افزایش یا کاهش) نرخ یادگیری را در الگوریتم پیشنهادی نشان می‌دهد.

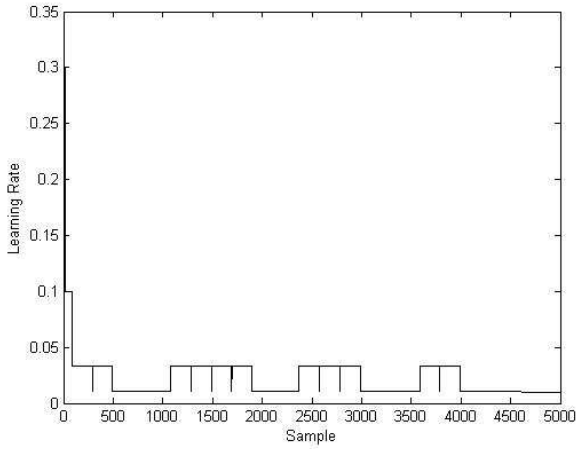
همچنان که شکل ۷ نشان می‌دهد، افزایش مقدار نرخ یادگیری در موقعیت‌هایی رخ داده که تغییرات بزرگی در محیط ایجاد شده است و در نتیجه، بردار احتمال انتخاب اقدام‌های آتاماتا به بردار واقعی محیط نزدیک‌تر است



شکل ٥- خط تیره درشت تابع ارزیابی اقدام آتاماتا توسط محیط را نشان میدهد. خط تیره نازک نحوه تغییر بردار احتمال مربوط به اقدام آتاماتای آموزش داده شده به شیوه جدید و خطوط نقطه چین، همان بردار، آموزش داده شده به شیوه معمول را نشان میدهد. (آزمایش۱)



شکل ٦-فاصله بردارهای آموزش داده شده به هریک از دو شیوه با بردار واقعی در آزمایش شماره ۱

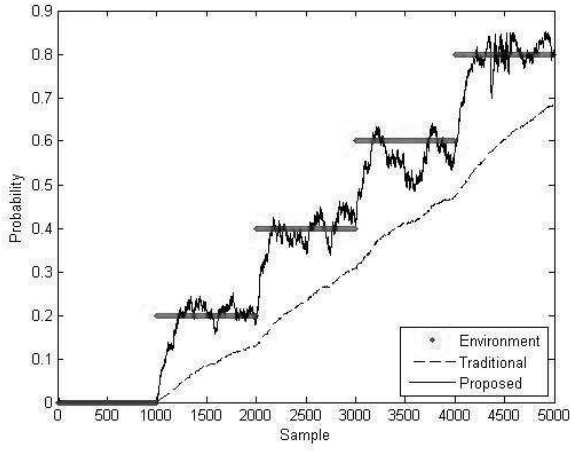


شکل ۷-تغییرات نرخ یادگیری در آزمایش ۱

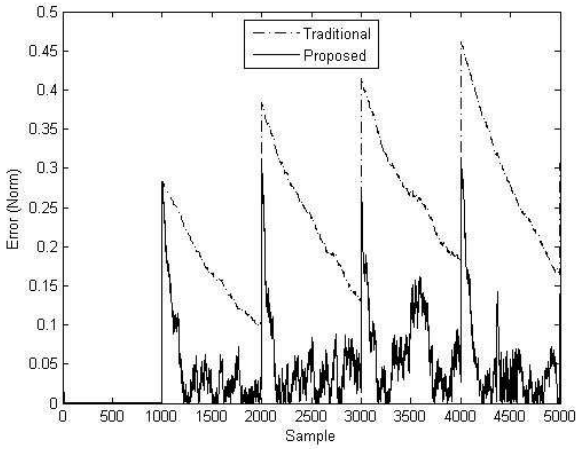
آزمایش ۲:

در این آزمایش از یک محیط تصادفی استفاده کرده ایم که تابع ارزیابی آن در طول زمان به شکل پله‌ای تغییر می‌کند. این تابع $f(i) = 0.2 * \left\lfloor \frac{5i}{n} \right\rfloor$ است که i معرفi امین نمونه و n نشان‌دهنده تعداد کل نمونه ها می‌باشد.

نتایج بررسی عملکرد مقایسه ای الگوریتم پیشنهادی و الگوریتم معمول (بهینه شده) ، در تنظیم نرخ یادگیری را در شکل ۸ و شکل ۹ مشاهده می‌کنید. شکل ۸ نشان میدهد که الگوریتم پیشنهادی انطباق بیشتری با رفتار پویای محیط داشته و شکل ۹ نیز موید میزان خطای کمتر روش جدید در مقایسه با روش معمول است.



شکل ۸- مقایسه روش معمول (خطوط نازک بریده) در مقایسه با روش پیشنهادی(خطوط تیره پیوسته) در یک محیط پویا (خطوط تیره پلکانی) در آزمایش ۲



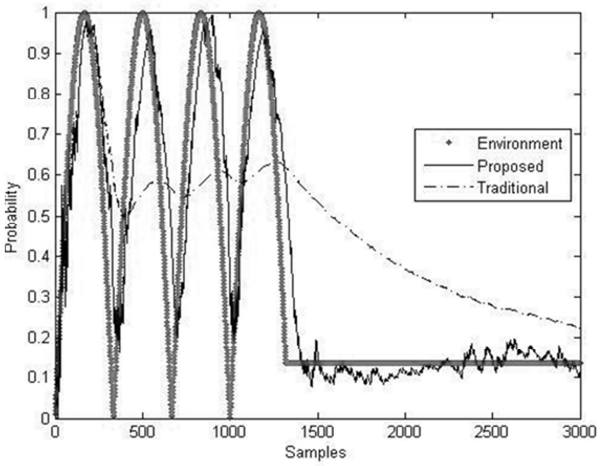
شکل ۹-فاصله بردارهای آموزش داده شده به هریک از دو شیوه با بردار واقعی در آزمایش شماره ۲

٤-۲ بررسی نتایج الگوریتم پیشنهادی ۲

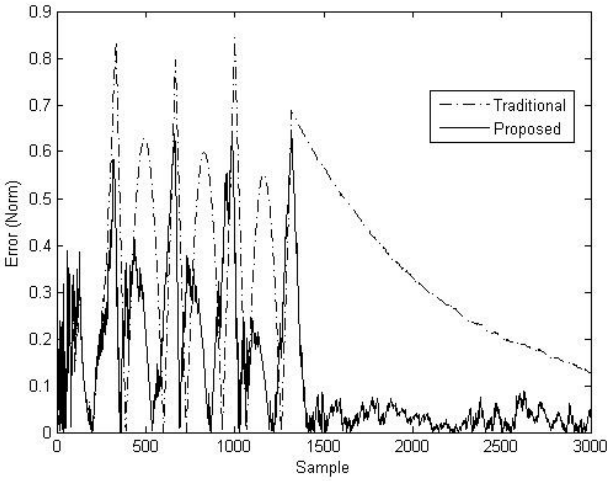
در این گروه از آزمایش‌ها، از الگوریتم ۲ (شکل ۴) استفاده کرده‌ایم. الگوریتم جدید از یک نرخ یادگیری دلخواه بزرگ آغاز کرده. در هر دور اجرا بر اساس میزان فاصله بردار جدید احتمال انتخاب به دست آمده برای اقدام های آتاماتا از بردار میانگین، اقدام به کاهش یا افزایش نرخ یادگیری نموده است.

در این الگوریتم، آتاماتای یادگیر یک نرخ یادگیری دارد و نامساوی چبیشف در حالت برداری آن به عنوان معیار تشخیصی برای تعیین آنکه آیا تغییر بزرگی در محیط رخ داده است یا نه مورد استفاده قرار می‌گیرد. نتیجه را در مورد نمونه هایی از محیط های پویا در ادامه بررسی کرده‌ایم.

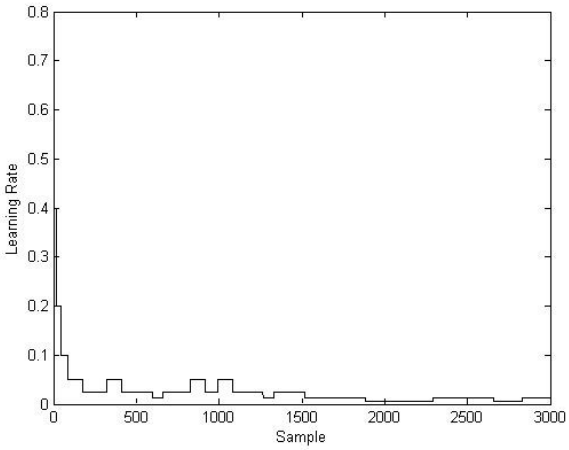
آزمایش ۳: محیط پویایی که برای این بررسی انتخاب شده است. تا حدود ۴۰٪ زمان آموزش از تابع پویای ارزیابی اقدام آتاماتا به صورت $\left|\sin\left(\frac{9i\pi}{n}\right)\right|$ برای ارزیابی اقدام آتاماتا استفاده می‌کند (در i امین نمونه که n تعداد کل نمونه ها است). اما از این مرحله به بعد، تابع به شکل خطی ثابت و بدون تغییر، اقدام آتاماتا را مورد ارزیابی قرار می‌دهد. نتایج مقایسه ای را در شکل ۱۰ و شکل ۱۱مشاهده می‌کنید. در شکل ۱۲ نیز نحوه تغییرات نرخ یادگیری نشان داده شده است. تلاش آتاماتا برای انطباق با محیط در بخش اول نمونه ها که محیط از پویایی بالایی برخوردار است، از طریق افزایش نرخ یادگیری قابل مشاهده است.



شکل ۱۰- خط تیره رنگ درشت تابع ارزیابی اقدام آتاماتا توسط محیط را نشان میدهد. خط تیره رنگ نازک نحوه تغییر بردار احتمال مربوط به اقدام آتاماتای آموزش داده شده به شیوه جدید و خطوط نقطه چین، همان بردار، آموزش داده شده به شیوه معمول را نشان میدهد

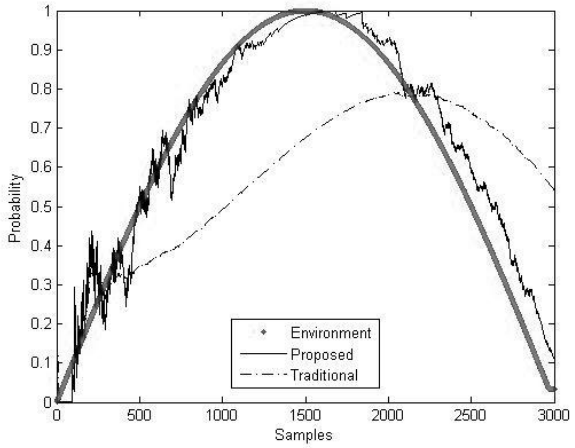


شکل ۱۱-فاصله بردارهای آموزش داده شده به هریک از دو شیوه با بردار واقعی در آزمایش شماره ۳

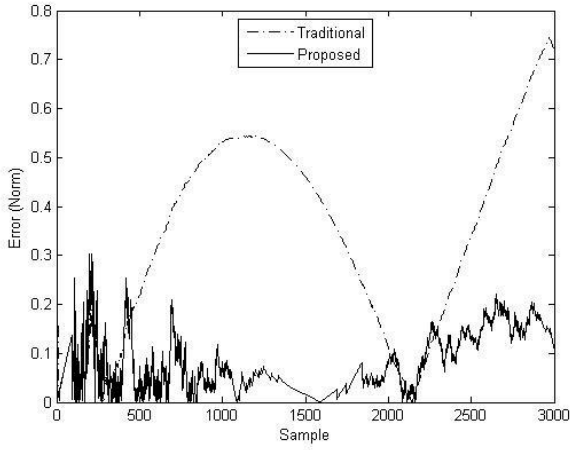


شکل ۱۲- تغییرات نرخ یادگیری در آزمایش ۳

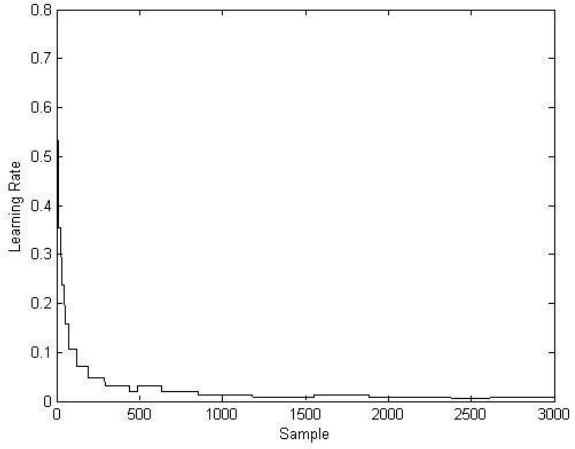
آزمایش ٤: برای این نمونه، محیط دیگری در نظر گرفته ایم که تابع ارزیابی محیط (احتمال پاداش به اقدام) از یک رفتار شبه سینوسی برخوردار است. نتیجه مقایسه ای عملکرد الگوریتم یادگیری L_{R-I} و الگوریتم جدید را در شکل ۱۳ و شکل ۱۴ مشاهده می‌کنید



شکل ۱۳- مقایسه روش معمول (خطوط نازک بریده) در مقایسه با روش پیشنهادی(خطوط تیره پیوسته) در یک محیط پویا (شبه سینوسی) در آزمایش ٤



شکل ١٤- فاصله بردارهای آموزش داده شده به هریک از دو شیوه با بردار واقعی در آزمایش شماره ٤



شکل ۱۵- تغییرات نرخ یادگیری در آزمایش ٤

۵- نتیجه گیری:

در این مقاله، به کمک شاخص های آماری حاصل از توزیع احتمال بردار انتخاب اقدام های آتاماتا، و با کمک نابرابری چبیشف روش‌های جدیدی پیشنهاد شد که به کمک آن آتاماتای یادگیر ضمن تظیم خودکار نرخ یادگیری، قادر به یادگیری در محیط‌های با پویایی بالا است

این الگوریتم‌های جدید قادر به تنظیم اتوماتیک نرخ یادگیری بوده و میتوانند بر حسب میزان تغییرات در محیط و به منظور انطباق با آن، اقدام به کاهش یا افزایش نرخ یادگیری نماید. استفاده از این الگوریتم جدید در محیط های تصادفی پویا که پاسخ محیط به اقدام انجام شده توسط آتاماتا غیر ابت و تابعی از زمان است مورد بررسی قرار گرفت و نشان داده شد که الگوریتم‌های جدید مبتنی بر نامساوی چبیشف از عملکرد بهتری نسبت به روش های یادگیری معمول برخوردارند.

٦- منابع

[1] H.Beigy, M. R. Meybodi, and M. B. Menhaj, “Adaptation of Learning Rate in Back Propagation Algorithm using Fixed Structure Learning Automata.,” in *6th Iranian Conference on Electrical Engineering*, 1998.

[2] S. Shah-Hosseini, “Automatic adjustment of learning rates of the self-organizing feature map,” *Scientia Iranica*, vol. 8, 2001.

[3] carlo H. sequin. chedsada chinrungrueng, “Optimal Adaptive K-Means Algorithm with Dynamic Adjustment of Learnig Rate.” 1991.

[4] J. Akbari Torkestani and M. R. Meybodi, “Learning automata-based algorithms for solving stochastic minimum spanning tree problem,” *Applied Soft Computing*, vol. 11, no. 6, pp. 4064–4077, Sep. 2011.

[5] J. Akbari Torkestani and M. R. Meybodi, “Learning Automata-Based Algorithms for Finding Minimum Weakly Connected Dominating Set in Stochastic Graphs,” *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 18, no. 06, pp. 721–758, Dec. 2010.

[6] H.Beigy and M.R.Meybodi, “Utilizing distributed learning automata to solve stochastic shortest path problems,” *International Journal of Uncertainty, …*, vol. 14, no. 5, pp. 591–615, 2006.

[7] M. R. MollakhaliliMeybodi and M.R.Meybodi, “A New Distributed Learning Automata Based Algorithm for Solving Stochastic Shortest Path,” in *6th Conference on Intelligent Systems*, 2004.

[8] H. Beigy and M. R. Meybodi, “Solving stochastic shortest path problem using Distributed Learning Automata,” in *6th Annual CSI Computer Conference (CSICC 2001)*.

[9] H. Beigy and M. . Meybodi, “Intelligent Channel Assignment in Cellular Networks: A Learning Automata Approach,” Amirkabir University of Technology, 2004.

[10] M. L. Thathachar and P. S. Sastry, “Varieties of learning automata: an overview.,” *IEEE transactions on systems, man, and cybernetics. Part B, Cybernetics : a publication of the IEEE Systems, Man, and Cybernetics Society*, vol. 32, no. 6, pp. 711–22, Jan. 2002.

[11] K. S. Narenda and M. A. L. Thatacher, *Learning Automata*. Prentice-Hall, 1989.

[12] M. L. Tsetlin, “on the behaviour of finite automata in random media,” *Automata., Telemekh.*, vol. 22, pp. 1345–1354, 1961.

[13] B. J. OOmen and H. MAsum, “Switching Models for Nonstationary Random Environments,” *IEEE transactions on systems, man, and cybernetics.*, vol. 25, no. 9, pp. 1334–1347.

[14] K. S. Narendra and M. A. L. Thathachar, “On the Behavior of a Learning Automaton in a Changing Environment with Application to Telephone Traffic Routing,” *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 10, no. 5, pp. 262–269.

[15] S. M. Ross, *Introduction to Probability and Statistics dor Engineers and Scientists*, Third. Elsevier Academic Press, 2004.

[16] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, 3rd ed. New York, USA: McGrawHill, 1991.