

تعیین توابع تمایز برای دسته‌بندی الگوها با استفاده از آتاماتونهای یادگیر سلسله مراتبی

غلامرضا رضایی محمدرضا ماییدی

دانشکده مهندسی کامپیوتر

دانشگاه صنعتی امیرکبیر

تهران - ایران

چکیده: یکی از روشهای دسته‌بندی الگوها، استفاده از توابع تمایز^(۱) می‌باشد. در این مقاله روشهای جدیدی براساس بازی آتاماتونهای یادگیر سلسله‌مراتبی با پاسخ یکسان^(۲) جهت تعیین پارامترهای بهینه توابع تمایز ارائه شده است. هر بازیگر بازی، یک واحد تشکیل شده از یک ساختار سلسله‌مراتبی (درختی) از آتاماتونهای یادگیر می‌باشد که وظیفه آن یادگیری یکی از پارامترهای تابع تمایز می‌باشد. در این روشها، تابع تمایز از طریق یک جستجوی احتمالاتی^(۳) در فضای پارامترهای تابع تمایز بدست می‌آید. در [۱۶] الگوریتمهایی براساس بازی آتاماتونهای یادگیر با ساختار ثابت برای تعیین پارامترهای تابع تمایز پیشنهاد گردید. در این روشها هر پارامتر تابع تمایز توسط یک آتاماتون منفرد در بازه‌ای که برای آن تعریف می‌شود، یاد گرفته می‌شود. تعیین دقیق بازه هر پارامتر می‌تواند تأثیر بسیار زیادی در کارایی الگوریتم داشته باشد. انتخاب بازه بزرگ باعث افزایش زمان همگرایی و بازه کوچک باعث پایین آمدن دقت دسته‌بندی می‌گردد. به منظور رفع این مشکل، در این مقاله روشهایی براساس آتاماتونهای یادگیر سلسله‌مراتبی پیشنهاد شده است. به این ترتیب برای هر پارامتر می‌توان بازه بزرگی در نظر گرفت و هر پارامتر را در محدوده وسیعتری جستجو کرد و احتمال تعیین صحیح پارامترها را افزایش داد. در این مقاله از آتاماتونهای یادگیر با ساختار ثابت به عنوان گره‌های ساختارهای سلسله‌مراتبی بازیگران بازی استفاده شده و کارایی این آتاماتونها در یادگیری توابع تمایز بهینه خطی یا غیر خطی مورد بررسی قرار می‌گیرند.

کلمات کلیدی: شناسایی الگو، توابع تمایز، آتاماتونهای یادگیر، ساختارهای سلسله‌مراتبی

۱ - مقدمه

یکی از روشهای دسته‌بندی الگوها، استفاده از توابع تمایز می‌باشد [۱] [۲] [۳] [۱۳]. برخی از روشهای تعیین تابع تمایز، با استفاده از آتاماتونهای یادگیر^(۴) ارقه شده‌اند. Andrew و Barto الگوریتمی به نام $A_{R,p}$ ارائه کرده‌اند که بر اساس آن، با برقراری یک رابطه انجمنی^(۵) بین هر عضو از نمونه ورودی و هر عمل تصمیم‌گیری با استفاده از آتاماتونهای یادگیر با ساختار متغیر^(۶)، توابع تمایز به فرم $g(x) = \theta^T x$ یاد گرفته می‌شود [۳]. Sastry و Thatachar با استفاده از بازی با پاسخ یکسان آتاماتونها و استفاده از آتاماتونهای یادگیر با ساختار متغیر به عنوان بازیگران بازی، روشی را جهت شناسایی پارامترهای بهینه تابع تمایز از طریق انجام یک جستجوی احتمالاتی در فضای پارامترهای آن پیش تعریف شده ارقه کرده‌اند [۴]. Phananskar و Thatachar با استفاده از شبکه‌های پیشخور^(۷) از آتاماتونهای یادگیر با ساختار متغیر با الگوریتم L_R-L ، روشی را جهت پیاده سازی یک سیستم دسته‌بندی سه سطحی خطی قطعه به قطعه^(۸) ارائه کرده و نشان داده‌اند که مجموعه می‌تواند به یک نقطه بهینه محلی همگرا شود [۵] [۶]. Sastry و Thatachar با ارائه گونه جدیدی از آتاماتونهای یادگیر با ساختار متغیر بنام PLA روشی را برای تعیین پارامترهای بهینه توابع تمایز با استفاده از شبکه‌هایی از آتاماتونهای یادگیر ارائه کرده [۸] [۷] و نشان داده‌اند که تحت شرایط ویژه‌ای سیستم می‌تواند به نقطه بهینه سراسری همگرا شود.

در [۱۶] الگوریتمهایی بر اساس بازی آتاماتونهای یادگیر با ساختار ثابت^(۹) پیشنهاد گردیده و نشان داده شده که مجموعه با کارایی بالایی می‌تواند توابع تمایز نزدیک به بهینه را یاد بگیرد. در روشهای ارقه شده در [۱۶] هر جزء بازی، یک آتاماتون منفرد می‌باشد که سعی در یادگیری یکی از پارامترهای تابع تمایز بهینه در فضای پارامترهای آن پیش تعریف شده می‌نماید. بدلیل اینکه مقدار هر پارامتر تابع تمایز از قبل مشخص نمی‌باشد، بازه مناسب برای هر پارامتر براساس حدس و گمان و تا اندازه‌ای با استفاده از اطلاعات موجود در مسئله تعیین می‌شود. انتخاب نامناسب بازه منجر به افزایش زمان همگرایی و یا پایین آمدن دقت دسته‌بندی می‌گردد. انتخاب بازه بزرگ باعث افزایش تعداد عملیات آتاماتون و در نتیجه باعث بالا رفتن زمان همگرایی و بازه کوچک ممکن است به کاهش دقت منجر شود. در این مقاله به منظور رفع مشکل تعیین بازه، از آتاماتونهای یادگیر سلسله‌مراتبی استفاده گردیده است. به این ترتیب می‌توان هر پارامتر را در محدوده وسیعتری جستجو کرد و از این طریق احتمال تعیین صحیح پارامترها افزایش می‌یابد. در این مقاله از آتاماتونهای یادگیر با ساختار ثابت به عنوان اجزای ساختار سلسله‌مراتبی استفاده گردیده است.

سایر بخشهای مقاله بصورت زیر تنظیم شده است. در بخش ۲ آتاماتونهای یادگیر بطور خلاصه تعریف می‌گردد و در بخش ۳ نحوه یادگیری پارامترهای تابع تمایز با استفاده از بازی آتاماتونهای یادگیر سلسله‌مراتبی ارقه خواهد شد. در بخش ۴ یک مسئله نمونه ۳ کلاسه تعریف می‌گردد که روشهای مختلف روی آن پیاده‌سازی خواهند شد. در بخش ۵ کارایی آتاماتونهای یادگیر با ساختار متغیر در حل مسئله نمونه بررسی خواهد شد. در بخشهای ۶ و ۷ دو گروه از آتاماتونهای یادگیر با ساختار ثابت جدید که مناسب برای استفاده در ساختارهای سلسله‌مراتبی می‌باشند پیشنهاد و معرفی می‌گردند. در بخش ۸ نتیجه‌گیری و مقایسه‌ای بین روشها ارقه شده، انجام خواهد شد.

Probabilistic Search (۳)

Game of Hierarchical Learning Automata with Common Pay-Off (۲)

Discriminant Function (۱)

Variable Structure LA (۶)

Associative Relation (۵)

Learning Automata (LA) (۴)

Fixed Structure LA (۹)

3-Layer Linear Piecewise Classification (۸)

Feedforward Network (۷)

۲ - آتاماتونهای یادگیر

یک آتاماتون یادگیر، یک آتاماتون احتمالی^(۱) است که با محیط خود بصورت پسجور^(۲) تراکنش دارد. مطابق شکل ۱، خروجی آتاماتون همی است که به عنوان ورودی به محیط داده می شود و خروجی محیط پاسخ محیط به عمل انجام شده است که به عنوان ورودی به آتاماتون ارسال می گردد. آتاماتون یادگیر بوسیله α تابعی $\{\Phi, \alpha, \beta, F(\cdot, \cdot), G(\cdot, \cdot)\}$ تعریف می شود که در آن $\Phi = \{\Phi_1, \dots, \Phi_S\}$ مجموعه حالات داخلی آتاماتون، $\alpha = \{\alpha_1, \dots, \alpha_P\}$ مجموعه عملها یا خروجی آتاماتون، β مجموعه ای محدود یا نامحدود است که بیانگر مجموعه ورودی آتاماتون می باشد. تابع F تابع تغییر حالت آتاماتون بر اساس حالت و ورودی فعلی و تابع G تابع خروجی آتاماتون بر اساس حالت فعلی است. محیط بوسیله β تابعی $\{\alpha, c, \beta\}$ تعریف می شود که در آن $\alpha = \{\alpha_1, \dots, \alpha_P\}$ مجموعه ورودی ها، β مجموعه خروجی ها و $c = \{c_1, \dots, c_P\}$ مجموعه مشخصات داخلی محیط است بطوریکه هر c_i با α_i به صورت $c_i = \Pr\{\beta(k) = \text{پاسخ نامطلوب} | \alpha(k) = \alpha_i\}$ رابطه دارد. به این ترتیب c_i احتمال دریافت پاسخ نامطلوب از جانب محیط است اگر عمل α_i در محیط انجام گیرد.

از نظر ساختاری آتاماتونهای یادگیر به دو دسته تقسیم می شوند: آتاماتونهای یادگیر با ساختار ثابت و با ساختار متغیر. در آتاماتونهای با ساختار ثابت، نحوه تغییر حالت آتاماتون بوسیله تابع F از قبل مشخص می شود. بر اساس حالت آتاماتون در مرحله k ام، عمل $\alpha(k)$ با استفاده از تابع $G(\cdot)$ در محیط انجام می شود و دریافت پاسخ $\beta(k)$ از جانب محیط حالت بعدی آتاماتون تعیین می گردد. در آتاماتونهای با ساختار متغیر، نحوه تغییر حالت از قبل مشخص نیست. در این مدل، انتخاب هر عمل بر اساس بردار احتمال عمل^(۳) که احتمال انتخاب هر عمل را در لحظه k ام معین می کند و به صورت $P(k) = \{P_1(k), \dots, P_P(k)\}$ ، $i=1, \dots, P$ ، $P_i(k) = \Pr\{\alpha(k) = \alpha_i | \beta(1), \dots, \beta(k-1)\}$ ، توزیع اولیه این بردار معمولاً بصورت یکنواخت در نظر گرفته می شود و در هر مرحله بر اساس پاسخ دریافت شده، این بردار به هنگام می شود. این به هنگام سازی با هدف یافتن عمل بهینه به منظور دریافت بیشترین احتمال پاسخ مطلوب از جانب محیط انجام می شود. برای اطلاعات بیشتر درباره آتاماتونهای یادگیر، می توانید به [۹] [۱۰] [۱۱] [۱۲] [۱۳] [۱۴] [۱۵] مراجعه نمایید.

۳ - نحوه یادگیری تابع تمایز

فرض کنیم $h(x) = g(\theta_1, \dots, \theta_N, x)$ تابع تمایز مورد نظر باشد بطوریکه $\theta_1, \dots, \theta_N$ پارامترهای تابع g هستند که باید یاد گرفته شوند و x بردار نمونه ویژگی است که باید دسته بندی شود. با استفاده از N آتاماتون یادگیر در قالب یک بازی یا پاسخ یکسان، مطابق شکل ۲، می توان پارامترهای فوق را تخمین زد. هر آتاماتون A^i دارای $\alpha_1^i, \dots, \alpha_{P_i}^i$ عمل می باشد و هر عمل آتاماتون A^i یک مقدار ممکن برای پارامتر θ_i از تابع تمایز در نظر گرفته می شود. مقادیر ممکن برای پارامتر θ_i در نتیجه مجموعه عملهای آتاماتون A^i از قبل تعیین شده اند. هدف از این بازی پیدا کردن مقادیری برای پارامترهای θ_i می باشد بطوریکه تعداد دسته بندی های غلط مینیمم گردد. در هر مرحله آتاماتون A^i با توجه به شیوه تصمیم گیری خود یک عمل (مقدار) از مجموعه عملهای خود (مجموعه مقادیر ممکن برای پارامتر θ_i) را انتخاب و به محیط اعلام می نماید. یک (یا چند) نمونه از بردارهای ویژگی موجود در محیط توسط تابع تمایز بدست آمده از پارامترهای انتخاب شده، دسته بندی می شود و نتیجه بصورت درست یا غلط (و یا نسبت تعداد دسته بندی صحیح به کل نمونه های آزمایش شده) به تیم ارسال می گردد. بر اساس نتیجه بدست آمده، آتاماتونها شیوه تصمیم گیری خود را در انتخاب عمل بعدی با هدف بدست آوردن پارامترهای بهینه برای تابع تمایز، به هنگام می نمایند.



شکل ۱: رابطه آتاماتون یادگیر با محیط

شکل ۲: بازی آتاماتونهای یادگیر با پاسخ یکسان

در این روش مقادیر ممکن برای پارامتر θ_i توسط مجموعه $\{\alpha_1^i, \dots, \alpha_{P_i}^i\}$ (که مجموعه عملهای آتاماتون A^i می باشد) با P_i مقدار گسسته در نظر گرفته می شود. در صورتیکه اطلاعات کمی در مورد حدود پارامتر θ_i در دست باشد، باید مجموعه بزرگی برای هر پارامتر در نظر گرفته شود، به این ترتیب تعداد عملهای آتاماتون A^i (که همان P_i است) می تواند به طور قابل ملاحظه ای افزایش یابد که این امر می تواند منجر به کند شدن سرعت فرایند یادگیری توسط هر آتاماتون گردد. برای حل این مشکل، می توان از بازی آتاماتونها به جای استفاده از آتاماتون منفرد، از واحدی از آتاماتونهای یادگیر (U^1) استفاده کرد که هر واحد شبکه ای درختی از آتاماتونهای یادگیر می باشد. برگهای این درخت حاوی مقادیر ممکن در نظر گرفته شده برای پارامتر θ_i می باشد. شکل ۳ ساختار یک واحد U^1 با دو سطح را نشان می دهد. در این ساختار انتخاب یک مقدار برای θ_i در دو قدم انجام می شود. در ریشه آتاماتون A^1 با P_1 عمل قرار دارد. هر عمل متناظر با یکی از آتاماتونهای سطح دوم می باشد. با انتخاب هر عمل، یکی از آتاماتونهای $A^2, \dots, A_{P_1}^2$ انتخاب می شود. در قدم بعد یکی از عملهای آتاماتون A^2 انتخاب می شود که این عمل یک مقدار ممکن برای پارامتر θ_i محسوب می گردد. به این ترتیب یک مسیر فعال در درخت ایجاد می شود که شامل آتاماتونهای A^1 و A^2 می باشد. با انجام عمل انتخاب شده در محیط، پاسخ محیط به واحد U^1 داده شده و نحوه تصمیم گیری آتاماتونهای فعال شده با توجه به پاسخ محیط به هنگام می شود و نحوه تصمیم گیری سایر آتاماتونها تغییر نمی کند. در انتهای فرایند یادگیری، هر واحد U^1 مسیری را یاد می گیرد که بیانگر مقدار بهینه برای پارامتر θ_i می باشد. در ساختارهای با عمق بیشتر از ۲ عمل می کنند. در هر سطح آتاماتون انتخاب شده، یک آتاماتون یادگیر در سطح بعدی را انتخاب می کند. با افزایش تعداد سطوح (عمق درخت) در هر واحد U^1 ، می توان تعداد بیشتری انتخاب

برای هر پارامتر β_i در نظر گرفت و به این ترتیب از محدوده بزرگتری برای هر پارامتر استفاده کرد.

۴ - یک مسئله نمونه

مسئله: پیدا کردن تابع تمایز برای سه کلاس W_1, W_2, W_3 با توزیعهای نرمال با مشخصات زیر:

$$W_1: M = (20, 10)^T, \Sigma = \begin{pmatrix} 4 & 0 \\ 0 & 1 \end{pmatrix}, W_2: M = (14, 4)^T, \Sigma = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}, W_3: M = (26, 4)^T, \Sigma = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}$$

بردار میانگین و ماتریس کوواریانس توزیعها می باشد. در شکل ۴، شکل توزیع کلاسهها و تابع تمایز هر کلاس نشان داده شده است.

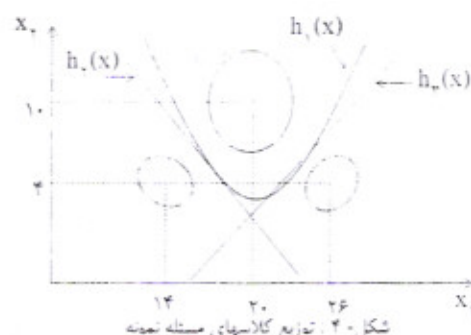
تابع تمایز $h_1(x)$ و $h_2(x)$ و $h_3(x)$ تابع تمایزی هستند که باید یاد گرفته شوند بطوریکه $\text{if } h_i(x) \geq 0 \text{ then } x \in W_i \text{ else } x \notin W_i$ با توجه به شکل

توزیع. تابع $h_1(x) = x_1 - 0.19x_2 + 0.6x_3 - 80 \geq 0 : x \in W_1$ $h_2(x) = x_1 - 0.19x_2 + 0.6x_3 - 80 \geq 0 : x \in W_1$ $h_3(x) = -x_1 + x_2 - 17 \geq 0 : x \in W_3$

می توانست تقریبهای خوبی از توابع تمایز کلاسههای مسئله باشند. همانطور که مشاهده می شود $h_1(x)$ دارای ۳ پارامتر و $h_2(x)$ و $h_3(x)$ دارای ۲ پارامتر هستند که باید یاد گرفته شوند. بنابراین از سه تیم آتاماتون مجزا شامل ۲ و ۲ و ۲ واحد آتاماتون که به طور مستقل کار می کنند، به

صورت $h_1: U_{11}, U_{12}, U_{13} \rightarrow h_1 = x_1 - \alpha_{11}x_2 + \alpha_{13}x_3 - \alpha_{10}$ $h_2: U_{21}, U_{22} \rightarrow h_2 = -x_1 + \alpha_{21}x_2 - \alpha_{20}$ $h_3: U_{31}, U_{32} \rightarrow h_3 = -x_1 + \alpha_{31}x_2 - \alpha_{30}$

برای یادگیری تابع تمایز استفاده می کنیم. U_{ij} ها واحدهای آتاماتون یادگیری هستند که پارامترهای تابع تمایز $h_i(x)$ را تعیین می کنند. α_{ij} پارامترهای تابع تمایز $h_i(x)$ هستند. مجموعه مقادیر ممکن برای پارامترهای این سه تابع تمایز بصورت رابطه (۱) در نظر گرفته شده است. هر مقدار توسط یک واحد دوسطحی از آتاماتونها که هر آتاماتون ۵ عمل دارد یاد گرفته می شود (۲۵ مقدار برای هر پارامتر). مجموعه مقادیر ممکن برای هر واحد در جلوی نام آن واحد نوشته شده است:



شکل ۴ - توزیع کلاسههای مسئله نمونه

$$\begin{aligned} U_{11} : \{ \alpha_{11}(t) | t \in \text{اعداد طبیعی}, 0 \leq \alpha_{11}(t) \leq 24, \alpha_{11}(t) = -0.05 + 0.02t \} \\ U_{12} : \{ \alpha_{12}(t) | t \in \text{اعداد طبیعی}, 0 \leq \alpha_{12}(t) \leq 24, \alpha_{12}(t) = 1/6 + 0.05t \} \\ U_{13} : \{ \alpha_{13}(t) | t \in \text{اعداد طبیعی}, 0 \leq \alpha_{13}(t) \leq 24, \alpha_{13}(t) = 20 + 0.5t \} \\ U_{21} : \{ \alpha_{21}(t) | t \in \text{اعداد طبیعی}, 0 \leq \alpha_{21}(t) \leq 24, \alpha_{21}(t) = -2 + 0.05t \} \\ U_{22} : \{ \alpha_{22}(t) | t \in \text{اعداد طبیعی}, 0 \leq \alpha_{22}(t) \leq 24, \alpha_{22}(t) = -1 + 0.1t \} \\ U_{31} : \{ \alpha_{31}(t) | t \in \text{اعداد طبیعی}, 0 \leq \alpha_{31}(t) \leq 24, \alpha_{31}(t) = -2 + 0.05t \} \\ U_{32} : \{ \alpha_{32}(t) | t \in \text{اعداد طبیعی}, 0 \leq \alpha_{32}(t) \leq 24, \alpha_{32}(t) = -7 + 0.1t \} \end{aligned} \quad (1)$$

در هر واحد U_{ij} ، در رشته یک آتاماتون با ۵ عمل وجود دارد که هر عمل آن یکی از آتاماتونهای سطح دوم را فعال می کند. هر آتاماتون سطح دوم ۵ عمل دارد که هر عمل متناظر با یکی از مقادیر ذکر شده در (۱) برای پارامتر α_{ij} می باشد. مقادیر ذکر شده برای هر U_{ij} در (۱) به ترتیب و به صورت ۵ عمل برای هر آتاماتون. بین آتاماتونهای سطح دوم توزیع می شوند.

در قسمت یادگیری، برای هر کلاس ۵۰۰ نمونه تصادفی آموزشی در نظر گرفته می شود. پس از انجام عمل یادگیری، تست و گزارش کارایی روی مجموعههای ۵۰۰ تایی که متفاوت با مجموعههای آموزشی هستند، انجام می گیرد. در هر آزمایش، نتایج اعلام شده، میانگین ۵۰ شبیه سازی می باشد. مسئله نمونه فوق در [۱۶] بوسیله مجموعه ای از آتاماتونهای یادگیر منفرد با استفاده از روشهای مختلف مبتنی بر آتاماتونهای یادگیر با ساختار متغیر و آتاماتونهای با ساختار ثابت حل شده است. در این مقاله هدف، استفاده از آتاماتونهای یادگیر سلسله مراتبی برای تعیین پارامترهای تابع تمایز می باشد.

۵ - آتاماتونهای یادگیر سلسله مراتبی با ساختار متغیر

در این بخش به بررسی کارایی آتاماتونهای یادگیر سلسله مراتبی با ساختار متغیر که دارای شیوه به هنگام سازی خطی احتمالات می باشند می پردازیم. اگر در مرحله k ام عمل $\alpha_i(k)$ توسط آتاماتون A^i انتخاب شده باشد، بردار احتمال عمل با استفاده از روابط ۲ و ۳ به هنگام می شود:

$$\begin{aligned} P_i(k+1) &= P_i(k) + a[1 - P_i(k)] \\ P_j(k+1) &= (1-a)P_j(k), \quad j = i \end{aligned} \quad (2) \quad \text{الف - با دریافت پاسخ مطلوب:}$$

$$\begin{aligned} P_i(k+1) &= (1-b)P_i(k) \\ P_j(k+1) &= \frac{b}{r-1} + (1-b)P_j(k), \quad j = i \end{aligned} \quad (3) \quad \text{ب - با دریافت پاسخ نامطلوب:}$$

و a و b پارامترهای یادگیری هستند [۹]. با توجه به روابط (۲) و (۳) با دریافت پاسخ مطلوب از محیط احتمال عمل انتخاب شده افزایش و احتمال سایر عملها کاهش می یابد و با دریافت پاسخ نامطلوب، احتمال عمل انتخاب شده کاهش و احتمال سایر عملها افزایش می یابد. باید توجه داشت که به هنگام سازی به گونه ای انجام می شود

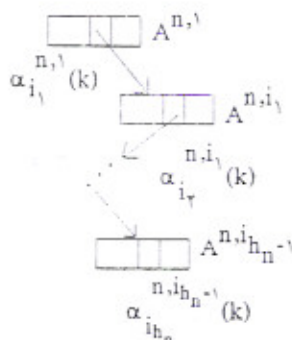
که در هر مرحله k داشته باشیم $\forall i \in [1 \dots r]: P_i(k) \geq 0, \sum_{i=1}^r P_i(k) = 1$. شرط پایان الگوریتم رسیدن احتمال عمل بهینه به بیش از ۰/۹۹ در نظر گرفته شده است.

برحسب اینکه مقادیر a و b رابطه آنها نسبت به هم چگونه باشند سه نوع الگوریتم به هنگام سازی حاصل می شود که عبارتند از: $L_R - p: 1 > a = b = 0$ ، $L_R - I: 1 > a > 0, b = 0$ ، $L_R - EP: 1 > a > 0, b = 0$.

آتاماتونهای یادگیر فوق به عنوان گرههای ساختار سلسله‌مراتبی ارائه شده در بخش ۲ در نظر گرفته شد و بردار احتمال عمل هر آتاماتون فعال شده، به‌وسیله رابطه ۲ و ۳ با توجه به پاسخ دریافت شده از جانب محیط به‌صورت گام‌گام گردید. نتایج نشان می‌دهد که الگوریتم L_{R-P} کارایی خوبی در حل مسئله از خود نشان می‌دهد به این معنی که سیستم به سمت انتخاب مجموعه پارامترهای بهینه همگرا نمی‌گردد. نمودارهای (۱-الف) و (۱-ب) نمودار دقت جداسازی و تعداد مراحل مورد نیاز برای همگراندن را برای الگوریتم L_{R-I} و نمودارهای (۲-الف) و (۲-ب) دقت و تعداد مراحل را برای الگوریتم L_{R-E-P} نشان می‌دهد. همانطور که ملاحظه می‌شود با کوچکتر شدن پارامترهای یادگیری (a و b)، دقت جداسازی و تعداد مراحل مورد نیاز افزایش می‌یابد. الگوریتم L_{R-E-P} با شرط $b = \frac{a}{\gamma}$ اجرا شد و مشاهده گردید که به ازای $a \leq 0.01$ سیستم همگرا نمی‌شود (این پدیده در [۱۶] نیز مشاهده گردیده است). بنابراین به ازای $a \leq 0.01$ ، الگوریتم با شرط $b = \frac{a}{\gamma}$ اجرا گردید. حداکثر دقت بدست آمد برای L_{R-I} به ازای $a = 0.005$ برابر $67/725$ درصد با تعداد متوسط مراحل ۱۹۱۵۵۱ و برای L_{R-E-P} به ازای $a = 0.0005$ ، $b = 0.005$ برابر $92/720$ با تعداد مراحل ۳۷۶۵۷۲ می‌باشد. با توجه به نتایج بدست آمده، می‌توان چنین نتیجه گرفت که برای دقتهای بالا، الگوریتم L_{R-I} برار کند عمل می‌کند.

۶- آتاماتونهای یادگیر سلسله‌مراتبی با ساختار ثابت (گروه اول)

در [۱۶] کارایی آتاماتونهای یادگیر با ساختار ثابت مورد بررسی قرار گرفت و نتایج بدست آمده نشان داد که آتاماتونهای ساختار ثابت با حفظ حافظه ۱ بهترین نتیجه را بدست می‌دهند. با اصلاح آتاماتونهای ساختار ثابت و ارائه آتاماتونهای جدید که آتاماتونهای نوع ۱ تا نوع ۶ نامیده شدند، آتاماتونهای با ساختار ثابت جدیدی با کارایی بالاتر ارائه شدند. در این بخش با توسعه آتاماتونهای پیشنهاد شده در [۱۶] و ارائه الگوریتمهای یادگیری مناسب برای آتاماتونهای یادگیر سلسله‌مراتبی با ساختار ثابت، سعی در حل مسئله نمونه می‌نماییم.



شکل ۵- مسیر فعال شده در واحد U^n



شکل ۵: تغییر حالت آتاماتون پیشنهادی

اگر $\Phi(k)$ حالت آتاماتون در مرحله k ام باشد، خروجی $\alpha(k)$ توسط آتاماتون در لحظه k انتخاب و به محیط اعلام می‌گردد و پاسخ $\beta(k)$ از جانب محیط به آتاماتون ارسال می‌گردد و حالت بعدی ($\Phi(k+1)$) آتاماتون با توجه به حالت فعلی و پاسخ دریافت شده، مشخص می‌گردد. شکل ۵ آتاماتونی با حالت Φ_1, \dots, Φ_r و عمل $\alpha_1, \dots, \alpha_r$ را نشان می‌دهد. تابع خروجی آتاماتون بصورت $G(\Phi_i) = \alpha_i$ تعریف می‌شود. آتاماتون در حالت $\Phi_i(k)$ عمل $\alpha_i(k)$ را انتخاب و به محیط اعلام می‌کند و پاسخ آن ($\beta(k)$) را که به یکی از دو صورت پاسخ مطلوب یا پاسخ نامطلوب می‌باشد را از محیط دریافت می‌نماید. در صورت دریافت پاسخ مطلوب، آتاماتون در حالت فعلی باقی‌مانده و با احتمال p تشویق می‌شود و در صورت دریافت پاسخ نامطلوب با احتمال q تنبیه می‌شود. به هر عمل آتاماتون وزن $w_i(k)$ با مقدار اولیه $\frac{1}{r}$ که r تعداد عملهای آتاماتون است، نسبت داده می‌شود. در صورت انجام عمل تشویق، وزن عمل انجام گرفته افزایش یافته و وزن سایر عملها کاهش می‌یابد. در صورت انجام تنبیه، آتاماتون تغییر حالت می‌دهد. این تغییر حالت می‌تواند بصورت تصادفی انجام گیرد [۹]. در الگوریتمهایی که پیشنهاد می‌شود وزن عمل $\alpha_i(k)$ با استفاده از رابطه ۴ به‌صورت گام‌گام می‌شود:

$$p_i(k+1) = p_i(k) + \frac{1 - p_i(k)}{W}, \quad p_j(k+1) = p_j(k) - \frac{p_j(k)}{W}, \quad j \neq i, \quad j \in [1, \dots, r] \quad (2)$$

W عددی صحیح است و پارامتر یادگیری نامیده می‌شود. اگر عمل انتخاب شده در مرحله k ام، $\alpha_i(k)$ باشد، با توجه به رابطه ۲ می‌توان بسادگی نشان داد:

$$\Delta w_i(k) = w_i(k+1) - w_i(k) = \frac{1 - w_i(k)}{W} \quad (5-1) \quad , \quad \sum_{i=1}^r w_i(k) = 1 \quad (5-2)$$

رابطه ۵-۱ نشان می‌دهد که با بزرگتر شدن W سرعت همگرایی کاهش می‌یابد.

احتمالهای p و q به حالت داخلی آتاماتون و حالت آتاماتونهای دیگری که در تیم با آتاماتون همکاری می‌کنند و همچنین به حالت آتاماتونهای دیگری که در سایر سطوح ساختار درختی از آتاماتونها قرار دارند، ارتباط دارد. انتخاب نوع ارتباط تاثیر بسیار زیادی در کارایی الگوریتم خواهد داشت.

فرض کنیم در بازی آتاماتونها، N واحد آتاماتون U^1, \dots, U^N شرکت دارند که هر واحد وظیفه پیدا کردن پارامتر θ_1 از تابع نمایان مربوطه را بر عهده دارد. هر واحد

U^n از یک ساختار درختی از آتاماتونهای یادگیر با h_n سطح تشکیل شده است. شکل ۶ مسیر فعال شده در این واحد را در لحظه k ام نشان می‌دهد. مطابق شکل ۶

آتاماتون $A^{n,1}$ که آتاماتون ریشه واحد U^n می‌باشد، عمل $\alpha_{i_1}^{n,1}(k)$ را انتخاب می‌کند که منجر به فعال شدن آتاماتون A^{n,i_1} در سطح دوم می‌شود و این آتاماتون

نیز عمل $\alpha_{i_2}^{n,i_1}(k)$ را انتخاب می‌کند که باعث فعال شدن آتاماتون A^{n,i_2} در سطح سوم می‌شود و به همین ترتیب یک مسیر فعال تا سطح h_n ام ایجاد می‌شود.

مقدار انتخاب شده برای پارامتر θ_n عمل $\alpha_{i_{h_n}}^{n,i_{h_n}-1}(k)$ که توسط آتامتون $A^{n,i_{h_n}-1}$ در سطح h_n انتخاب شده است، می‌باشد. به این ترتیب در هر لحظه k در واحد U^n و در سطح m ، آتامتون A^{n,i_m-1} فعال شده و عمل $\alpha_{i_m}^{n,i_m-1}(k)$ با وزن $w_{i_m}^{n,i_m-1}(k)$ انتخاب می‌شود (برای سازگار شدن نگارش رابطه ذکر شده با نام آتامتون ریشه در شکل ۵، i_0 برابر ۱ فرض می‌شود). برای این آتامتون احتمالی p و q بصورت مطابق روابط ۱۲ تا ۱۴ تعریف می‌شوند:

$$p = \begin{cases} 1 & \text{if } PI \geq P_{Max} \\ \frac{PI}{P_{Max}} & \text{otherwise} \end{cases} \quad (6) \quad , \quad p = \begin{cases} 1 & \text{if } PI \leq P_{Max} \\ \frac{P_{Max}}{PI} & \text{otherwise} \end{cases} \quad (7)$$

طوری‌که:

$$PI = \sqrt[N]{\prod_{n=1}^N PW_n} \quad (8) \quad , \quad P_{Max} = \sqrt[N]{\prod_{n=1}^N P_{Max_n}} \quad (9)$$

$$PW_n = \left(\prod_{j=m}^{h_n} w_{i_j}^{n,i_{j-1}}(k) \right)^{\frac{1}{h_n-m+1}} \quad (10) \quad , \quad P_{Max_n} = \left(\prod_{j=m}^{h_n} w_{i_j}^{n,i_{j-1}}(k) \right)^{\frac{1}{h_n-m+1}} \quad (11)$$

$$w_{i_j}^{n,i_{j-1}}(k) = \text{Max} \left\{ w_{i_j}^{n,i_{j-1}}(k) | j=1, \dots, r_{n,i_j}, j=i_{j-1} \right\} \quad (12)$$

رابطه ۱۲ حداکثر وزن عملی انتخاب شده توسط آتامتون $A^{n,i_{j-1}}$ را تعریف می‌کند. رابطه ۱۰ و ۱۱ میانگین هندسی وزن عملی انتخاب شده و حداکثر وزن عملی انتخاب نشده آتامتونهای قرار گرفته در مسیر فعال شده در واحد U^n را از سطح m به بعد محاسبه می‌کند. رابطه ۸ و ۹ میانگین هندسی حاصل روابط ۱۰ و ۱۱ را بین واحدهای مختلف شرکت کننده در بازی محاسبه می‌کند. توسط روابط ۱۰ و ۱۱ وزن هر آتامتون در ساختار درختی توسط آتامتونهای زیرین تحت تاثیر قرار می‌گیرد. به این ترتیب، تنبیه و تشویق آتامتونهای موجود در مسیر با یکدیگر هماهنگ می‌شود. رابطه ۸ و ۹ بین آتامتونهای فعال شده همسطح در واحدهای مختلف ارتباط برقرار می‌کند که این ارتباط باعث همگون شدن پادگیری پارامترهای مختلف تابع تمایز در واحدهای مختلف شرکت کننده در بازی می‌شود. روابط ۶ و ۷ احتمالی تنبیه و تشویق را در هر لحظه تعریف می‌کنند. طبق رابطه ۶ اگر پاسخ مثبت (مطلوب بودن عمل انجام شده) در حالی دریافت شود که وزن عملی انتخاب شده در مسیرهای فعال بیشتر از وزن سایر عملها باشد، احتمالاً عملی انتخاب شده، عملی بهینه بوده و آتامتونهای فعال شده قطعاً تشویق می‌شوند. در غیر اینصورت احتمالاً عملی انتخاب شده، عملی بهینه نبوده و بلکه در مقطعی باعث تولید پاسخ مثبت شده‌اند. بنابراین آتامتونهای شرکت کننده در بازی با احتمال کمتری تشویق می‌شوند. طبق رابطه ۷ اگر پاسخ منفی (نامطلوب بودن عمل انجام شده) در حالی دریافت گردد که وزن عملی انتخاب شده در مسیرهای فعال کمتر از وزن سایر عملها باشد، احتمالاً عملی انتخاب شده، عملی بهینه نبوده و آتامتونهای فعال قطعاً تنبیه می‌شوند. در غیر اینصورت احتمالاً عملی انتخاب شده، عملی بهینه نبوده‌اند اما در مقطعی باعث تولید پاسخ منفی شده‌اند. بنابراین آتامتونهای شرکت کننده در بازی با احتمال کمتری تنبیه می‌شوند. روابط ۱۲ تا ۱۴ از سولات اضافی آتامتونهای شرکت کننده در بازی جلوگیری کرده و باعث سریع‌تر شدن همگرایی آتامتونهای شرکت کننده در بازی به سمت انتخاب مجموعه عملی بهینه (پارامترهای بهینه برای تابع تمایز) می‌شوند. الگوریتم تعریف شده در روابط ۶ تا ۱۴ در بازی آتامتونها را **الگوریتم نوع ۱** می‌نامیم.

الگوریتم نوع ۱ برای مسئله نمونه پیدسازی و اجرا شده و نمودارهای درصد دقت دستبندی صحیح و متوسط تعداد مراحل مورد نیاز برای همگرایی بر حسب پارامتر W در نمودارهای (۳-الف) و (۳-ب) آمده است. همانطور که انتظار می‌رفت، با افزایش W همگرایی کندتر و دقت دستبندی بیشتر شده است. حداکثر دقت بدست آمده به ازای $W = 200$ برابر $97/810$ با تعداد مراحل 150953 حاصل شده است.

با ایجاد تغییراتی در الگوریتم نوع ۱ می‌توان الگوریتمهای دیگری بدست آورد. طبق روابط ۱۰ و ۱۱، PW_n و P_{Max_n} بصورت میانگین هندسی وزن عملی آتامتونهای فعال شده در مسیر فعال از پایین به بالا محاسبه شده است. این ترتیب به این علت در نظر گرفته شده است که آتامتونهای موجود در درختی آتامتونها، مستقیماً با محیط در ارتباطند و حالیکه عملی انتخاب شده توسط آتامتونهای بالایی از طریق آتامتونهای پایینی با محیط ارتباط پیدا می‌کنند. با تعویض ترتیب حاصلضرب مطابق روابط ۱۳ و ۱۴ می‌توان الگوریتم دیگری بدست آورد. در روابط ۱۳ و ۱۴ میانگین هندسی در مسیر فعال از ریشه به سمت پایین محاسبه شده است. الگوریتم جدید را **الگوریتم نوع ۲** می‌نامیم. نتیجه اجرای این الگوریتم در نمودارهای (۴-الف) و (۴-ب) آمده است. حداکثر دقت جداسازی به ازای $W = 200$ برابر $97/211$ با تعداد مراحل 248760 بدست آمده است.

$$PW_n = \left(\prod_{j=1}^m w_{i_j}^{n,i_{j-1}}(k) \right)^{\frac{1}{m}} \quad (13) \quad , \quad P_{Max_n} = \left(\prod_{j=1}^m w_{i_j}^{n,i_{j-1}}(k) \right)^{\frac{1}{m}} \quad (14)$$

در الگوریتم نوع ۲ به جای استفاده از میانگین هندسی وزنهای آتامتونهای شرکت کننده در بازی استفاده کرد. به این ترتیب **الگوریتم نوع ۳** حاصل می‌شود. در این الگوریتم PW_n و P_{Max_n} مطابق روابط ۱۵ و ۱۶ تعریف می‌شوند.

$$PW_n = \prod_{j=m}^{h_n} w_{i_j}^{n,i_{j-1}}(k) \quad (15) \quad , \quad P_{Max_n} = \prod_{j=m}^{h_n} w_{i_j}^{n,i_{j-1}}(k) \quad (16)$$

با انجام این تغییر تاثیر آتامتونهای قرار گرفته در سطوح بالایی نسبت به الگوریتم نوع ۱ تضعیف می‌شوند. چون هر آتامتون قرار گرفته در سطوح بالاتر نماینده کلیه آتامتونهای موجود در سطوح پایین‌تر که مرتبط با آن هستند، می‌باشد، این ستراتی همانطور که نتایج آزمایشها نشان می‌دهد، باعث افزایش دقت دستبندی و تعداد

مراحل شده است. نتیجه اجرای الگوریتم نوع ۳ در نمودارهای (الف-۵) و (ب-۵) آمده است. ملاحظه می‌شود که حداکثر دقت بدست آمده به ازای $W = 250$ برابر $97/83\%$ با تعداد مراحل ۱۳۳۸۸۱ می‌باشد با ایجاد تغییرات دیگری در PW_n و PWM_{\max} می‌توان نتایج بهتری بدست آورد. برای آتاماتون A با ۲ عمل. حداکثر وزن یک عمل در طول فرآیند یادگیری طبق رابطه ۱۷ تعریف می‌شود:

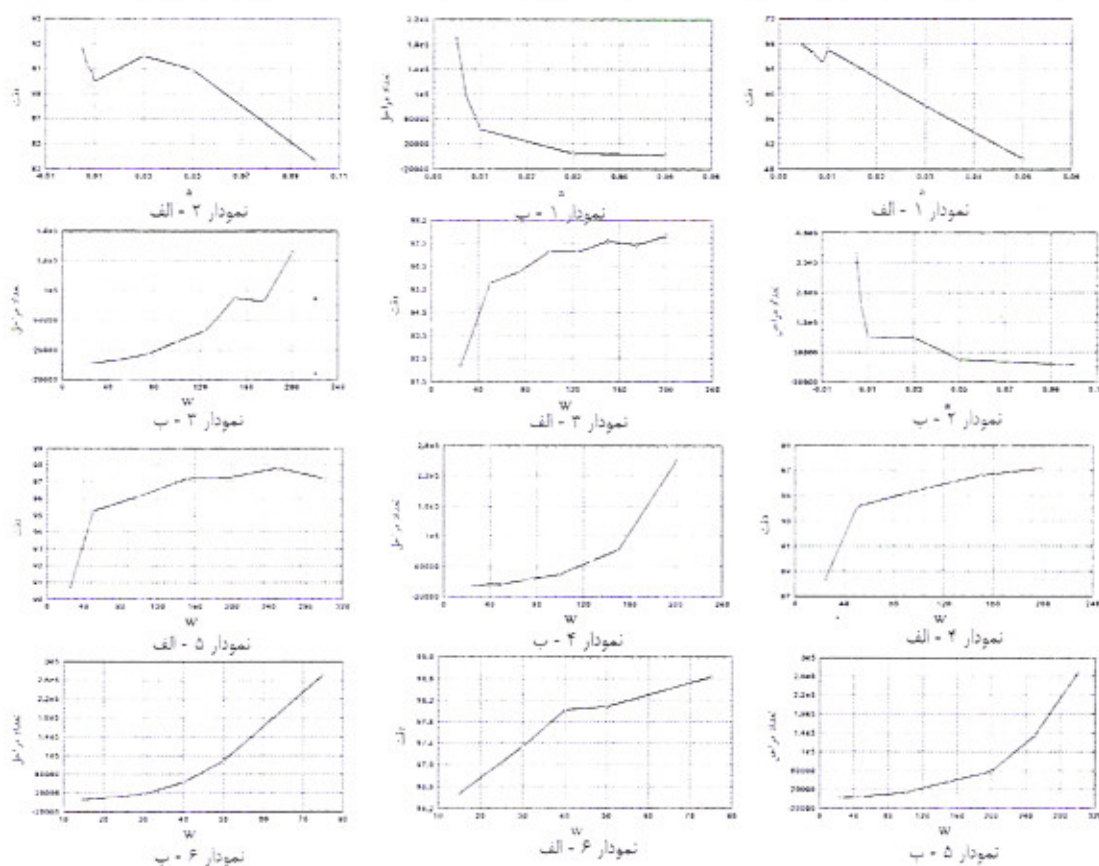
$$w_{\max_j}(k) = \text{Max} \{w_j(k') | 1 \leq j \leq r, 0 \leq k' \leq k\} \quad (17)$$

با قرار دادن $w_{\max_j}(k)$ در روابط ۱۰ تا ۱۱ روابط ۱۸ تا ۲۰ بدست می‌آید:

$$PW_n = \left(\prod_{j=m}^{h_u} w_{\max_{i_j}^{n,i_{j-1}}}(k) \right)^{\frac{1}{h_n - m + 1}} \quad (18) \quad , \quad PWM_{\max_n} = \left(\prod_{j=m}^{h_u} w_{\max_{i_j}^{n,i_{j-1}}}(k) \right)^{\frac{1}{h_n - m + 1}} \quad (19)$$

$$w_{i_j, \max}^{n,i_{j-1}}(k) = \text{Max} \left\{ w_{i_j}^{n,i_{j-1}}(k) | j=1, \dots, r_{n,i_j} = i_{n,i_j} \right\} \quad (20)$$

با قرار دادن روابط ۱۸ تا ۲۰ بجای روابط ۱۰ تا ۱۳ در الگوریتم شماره ۱. و با توضیح آتاماتون به محض دریافت پاسخ مطلوب از جانب محیط. الگوریتم نوع ۴ حاصل می‌شود. شرط پایان الگوریتم رسیدن وزن کلیه آتاماتونهای فعال به مقدار ثابتی (۰/۹۹) در نظر گرفته شده است. با استفاده از روابط ۲۰ تا ۲۲ و همچنین آتاماتون با دیدن پاسخ مطلوب. آتاماتون می‌تواند سریع‌تر به پاسخ مطلوب برسد. الگوریتم ۴ با در نظر گرفتن مقادیر ۱۵,۳۰,۴۰,۵۰,۷۵ برای W در هریک از آتاماتونهای شرکت کننده در واحدهای U_{11}, U_{12}, U_{13} و مقدارهای ۱۵۰,۲۰۰,۱۰۰,۵۰,۲۵ برای W در هریک از آتاماتونهای شرکت کننده در سایر واحدها. پیاده‌سازی و اجرا شد که نتایج آن در نمودارهای (الف-۶) و (ب-۶) آمده است. همانطور که مشاهده می‌شود، به ازای $W = 40$ دقت $98/02\%$ با تعداد مراحل ۲۲۲۲۲ و به ازای $W = 50$ دقت $98/08\%$ با تعداد مراحل ۸۹۰۳۹ و به ازای $W = 75$ دقت $98/63\%$ با تعداد مراحل ۲۷۱۹۱۲ بدست آمده است که بهتر از نتایج قبلی می‌باشد.



۷ - آتاماتونهای یادگیر سلسله‌مراتبی با ساختار ثابت (گروه دوم)

در این قسمت آتاماتونی پیشنهاد می‌دهیم که وزنیهای w_j^n در ساختار آتاماتون گنجانده شده است. آتاماتون پیشنهادی دارای دو عمل است که برای بیش از دو عمل نیز براحتی قابل گسترش است. آتاماتونی با دو عمل α_1, α_2 و Φ_N حافظه $2N$ و $\Phi_{N+1}, \Phi_N, \dots, \Phi_1$ را در نظر بگیرد (مفهوم از حافظه تعداد حالت‌های آتاماتون می‌باشد [۹]). Φ_1 تا Φ_N مجموعه حالتی است که اگر آتاماتون در آن قرار بگیرد، عمل α_1 و Φ_{N+1} تا Φ_N مجموعه حالتی است که اگر آتاماتون در آن قرار بگیرد عمل α_2 را انتخاب می‌کند. گراف تغییر حالت آتاماتون با آمدن پاسخ مطلوب و پاسخ نامطلوب از جانب محیط در شکل (الف-۶) و (ب-۶) نشان داده شده است. همانطور که در شکل (الف-۶) و (ب-۶) نشان داده شده است. آتاماتون در حالت Φ_{α_1} قرار دارد و عمل α_1 را انتخاب می‌کند. با دریافت پاسخ مطلوب از جانب محیط. آتاماتون با احتمال p تشویق می‌شود که در این صورت به عمل α_1 یک واحد اضافه می‌شود (آتاماتون به حالت Φ_{α_1+1} می‌رود) و در صورتیکه

آتانئون در حالت Φ_N باشد در همان حالت باقی می‌ماند. در صورت انجام عمل تشویق، فرض می‌کنیم حالت فنی آتانئون در انتخاب عمل α_p باشد. در این صورت با احتمال q عمل α_p تشویق می‌شود که در این صورت: (اگر عمل قبلی α_{p-1} باشد) یک واحد از عمق آن کم می‌شود (عمق قبلی عمل α_{p-1} در نظر گرفته می‌شود). با دریافت پاسخ نامطلوب از جانب محیط آتانئون با احتمال q' تشویق می‌شود که در این صورت، حالت فعلی آن به عنوان حالت قبلی عمل α_p ذخیره شده و آتانئون به حالت قبلی عمل α_p (که در شکل Φ_{α_p} است) تغییر حالت می‌دهد. عمق اولیه برای کلیه عملها در شروع کار آتانئون $\frac{M}{T}$ در نظر گرفته می‌شود که M عمق حافظه آتانئون و T تعداد عملهای آن است.

برای ارتباط دادن پاسخ محیط با q, q', p ، q' ، میانگین‌های مختلفی در نظر گرفت. اگر آتانئون A تا T عمل معروض باشد، وزن هر عمل را می‌توان مطابق رابطه ۲۱ تعریف نمود:

$$w_i(k) = \frac{m_i(k)}{M} \quad (21)$$

رابطه ۲۱ برای هر عمل وزنی برابر عمق حافظه عمل $(m_i(k))$ به برابر حداکثر تعداد حالتهای در نظر گرفته شده برای هر عمل است.

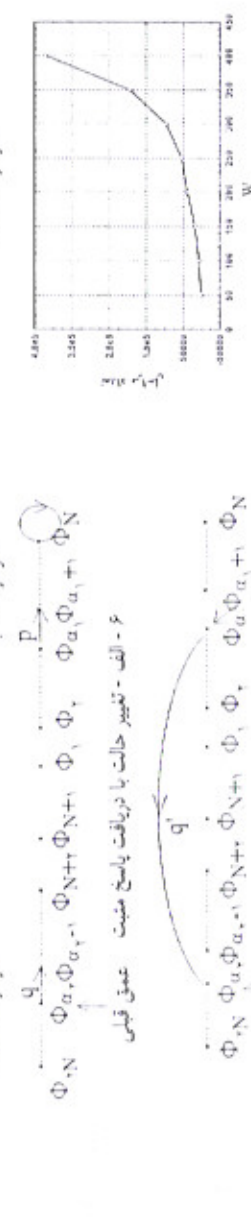
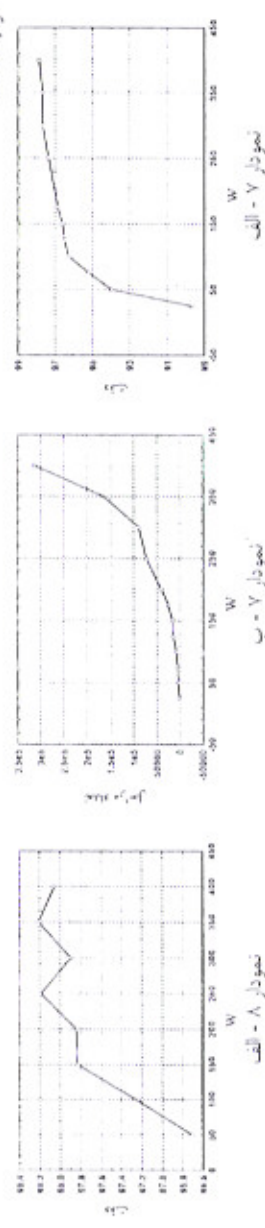
تعریف می‌نماییم: با تعریف p, q و q' مطابق روابط ۲۲ تا ۲۴. برای آتانئون فرار گرفته در واحد I^{th} و در سطح m م الگوریتم نوع ۵ به صورت می‌آید:

$$p = \begin{cases} \frac{PI}{PMax} & \text{if } PI \approx PMax \\ \frac{n \cdot j_{i,j} \cdot w_i(k)}{I_j} & \text{otherwise} \end{cases} \quad (22) \quad q = w_i \quad (23) \quad q' = \begin{cases} \frac{PMax}{PI} & \text{if } PI \leq PMax \\ \text{otherwise} \end{cases} \quad (24)$$

روابط ۲۲ و ۲۴ مانند روابط ۹ و ۸ عمل می‌کنند و رابطه ۲۳ احتمال q را برابر وزن عمل تعریف می‌کند. مقادیر PI و $PMax$ مطابق روابط ۱۲ تا ۸ محاسبه می‌شوند.

الگوریتم نوع ۵ برای مسئله نمونه پیاده‌سازی و اجرا شده و نتایج به دست آمده در نمودارهای (الف-۷) و (ب-۷) آمده است. همانطور که مشاهده می‌شود به ازای $W = 200$ دقت $99/87\%$ با متوسط تعداد مراحل 31200 به دست آمده است.

با تعریف حداکثر وزن به دست آمده، مانند رابطه ۲۰ (که در این گروه از آتانئونها پیگیر حداکثر عمق یک عمل در طول فرایند یادگیری می‌باشد) و استفاده از آنها در محاسبه PI و $PMax$ مشابه روابط ۱۸ و ۱۹ و تشویق نمودن آتانئون به محض دریافت پاسخ مطلوب از جانب محیط (در نظر گرفتن احتمال تشویق برابر ۱). الگوریتم دیگری به دست می‌آید که آن را الگوریتم نوع ۶ می‌نامیم. در این الگوریتم با 99% بودن احتمال تشویق، سعی در همگرا نمودن سریع تر آتانئونها به سمت انتخاب عملهای بهینه می‌شود. به الگوریتم نوع ۶ برای مسئله نمونه پیاده‌سازی و اجرا شده و نتایج آن در نمودارهای (الف-۸) و (ب-۸) آمده است. با توجه به نمودارها مشاهده می‌شود که حداکثر دقتی به دست آمده به ازای $W = 250$ برابر $99/90\%$ با تعداد متوسط مراحل 53355 و به ازای $W = 350$ برابر $99/91\%$ با تعداد مراحل 194912 حاصل گردیده است. همانطور که مشاهده می‌شود به ازای $W > 250$ مقداری آشفتگی در محض دقت بوجود آمده که باعث لغت میزان افزایش دقت بر حسب گردیده است.



نمودار ۷ - الف

نمودار ۷ - ب

۸ - نتیجه گیری

در این مقاله، روشهای مختلفی بر اساس بازی آتانئونهای یادگیر سلسله‌ممراتی با پاسخ یکسان برای تعیین پارامترهای تابع تمایز برای شناسایی الگو ارائه شد. هر واحد بازیگر بازی. از یک ساختار سلسله‌ممراتی از آتانئونهای یادگیر تشکیل می‌شود و وظیفه آن یادگرفتن یکی از پارامترهای تابع تمایز در محدوده ارزشی تعریف شده برای آن واحد، می‌باشد. استفاده از ساختارهای سلسله‌ممراتی می‌توان محدود و وسیع‌تر نسبت به ساختار تک‌سختی برای هر پارامتر در نظر گرفت و به دقت و سرعت بالاتری دست یافت. این روشها با هدف بررسی کارایی آتانئونهای ساختار ثابت در ساختارهای سلسله‌ممراتی برای یافتن توابع تمایز بهینه در فضای پارامترهای تابع تمایز از طریق انجام یک جستجوی احتمالاتی ارائه شدند. در بخش ۵ آتانئونهای با ساختار ثابت جدیدی با عمق حافظه ۱۰ مقدار وزن W برای هر عمل آتانئون، ارائه شدند و نشان داده شد که استفاده از این آتانئونها به عنوان اجزای ساختار درختی هر واحد بازیگر، می‌تواند با کارایی خوبی مسئله تعریف شده را حل کند. در بخش ۷ با ارائه آتانئونهای دیگری با عمق حافظه بیش از یک و گنجاندن وزن هر عمل در ساختار آتانئون، آتانئونهای مناسب دیگری پیشنهاد شدند. با توجه به نتایج به دست آمده، می‌توان ادعا کرد که با ارائه الگوریتمهای مناسب بر اساس آتانئونهای سلسله‌ممراتی با ساختار ثابت می‌توان پارامترهای توابع تمایز بهینه را با دقت خوبی تعیین زد.

با توجه به اینکه مسئله حل شده از سه توزیع نرمال تشکیل شده است، می‌توان با استفاده از قانون بیز، دقت تابع تمایز بهینه را که دستبندی خط را می‌توانیم می‌کند

اندازه‌گیری کرد. نتیجه ۵۰ بار آزمایش قانون بیز روی ۵۰ مجموعه نمونه مختلف، دقت متوسط ۹۸/۷۱۷ را بدست می‌دهد. جدول شماره ۱ خلاصه نتایج بدست آمده از کارایی روشهای مختلف ارائه شده در این مقاله در حل مسئله نمونه را نشان می‌دهد.

جدول ۱ علاوه بر خلاصه نتایج بدست آمده برای حل مسئله نمونه ارائه شده در بخش ۴، نتایج بدست برای حل مسئله نمونه شماره ۲ را نیز نشان می‌دهد. در مسئله شماره ۲، میزان روی هم افتادگی بیشتری برای کلاسه‌های مسئله نمونه در نظر گرفته شد. به این ترتیب دقت جداسازی کمتری امکان‌پذیر است. حتمهای مشخص شده با --- به معنی عدم همگرایی سیستم می‌باشد. همانطور که مشاهده می‌شود، به ازای برخی پارامترها، الگوریتمهای نوع ۲ و نوع ۴ قادر به همگرایی شدن نمی‌باشند. همچنین در هر دو مورد می‌توان مشاهده نمود که آتاماتونهای یادگیر با ساختار ثابت کارایی خوبی در حل مسئله‌ها از خود نشان می‌دهند.

جدول ۱

| مورد اول | | | مورد دوم | | |
|------------------|-------------|--------|----------|-------------|--------|
| روش | پارامترها | دقت | مرحله | پارامترها | دقت |
| L_{max} | $a = 0.005$ | ۹۷/۷۴۵ | ۱۹/۱۵۵۱ | $a = 0.005$ | ۶۱/۴۵۳ |
| L_{max} | $a = 0.005$ | ۹۲/۸۴۰ | ۳۷/۶۵۷۱ | $a = 0.009$ | ۸۰/۷۴۶ |
| الگوریتم نوع ۱ | $W = 300$ | ۹۷/۸۱۰ | ۱۵۰/۹۵۳ | $W = 300$ | ۸۶/۷۶۷ |
| الگوریتم نوع ۲ | $W = 300$ | ۹۷/۲۱۱ | ۲۴/۸۶۰ | $W = 300$ | ۷۳/۰۷۲ |
| | | | | $W = 50$ | ----- |
| الگوریتم نوع ۳ | $W = 250$ | ۹۷/۸۳۴ | ۱۳۳/۸۸۱ | $W = 300$ | ۷۵/۵۴۰ |
| | | | | $W = 60$ | ۸۸/۲۲۰ |
| الگوریتم نوع ۴ | $W = 75$ | ۹۸/۶۴۰ | ۲۷/۱۱۹۲ | $W = 15$ | ۸۹/۰۳۳ |
| | | | | $W = 35$ | ----- |
| الگوریتم نوع ۵ | $M = 300$ | ۹۸/۸۴۷ | ۳۱۲۰/۵۱ | $M = 60$ | ۸۸/۸۱۴ |
| الگوریتم نوع ۶ | $M = 350$ | ۹۸/۱۹۰ | ۱۹۴/۹۱۲ | $M = 60$ | ۸۷/۰۷۱ |
| | | | | $M = 100$ | ۸۸/۰۰۶ |
| قانون بیز | | ۹۸/۷۱۷ | | | ۹۲/۷۵ |

۹ - مراجع

- [1] Fukunaga, An Introduction to Statistical Pattern Recognition, New York, Academic Press, 1972.
- [2] Pierre A. Devijver and Joseph Kittler, Pattern Recognition Theory and Applications, NATO ASI Series, Series F: Vol. 30, 1986.
- [3] Andrew G. Barto and P. Anandan, "Pattern Recognition Stochastic Learning Automata", IEEE Trans. Sys. Man and Cybern., Vol. SMC-15, No. 3, May/January 1985.
- [4] Mandayam A. L. Thatachar and P. S. Sastry, "Learning Optimal Discriminant Functions Through a Cooperative Game of Automata", IEEE Trans. Sys. Man and Cybern., Vol. SMC - 17, No. 1, January/February 1987.
- [5] M. A. L. Thatachar and V. V. Phananskar, "A feedforward Network of Learning Automata for Pattern Recognition", in Proc. Int. Joint. Conf. Neural Networks, Singapore, Nov. 1991.
- [6] M. A. L. Thatachar and V. V. Phananskar, "Convergence of Teams and Hierarchies of Learning Automata in Connectionist Systems", IEEE Trans. Sys. Man and Cybern., Vol. 25, No 11, Nov. 1995.
- [7] V.V. Phananskar and M.A.L. Thatachar, "Global Convergence of Teams of Learning Automata", in Symp. Intell. Syst., Bangalore, Dec. 1991.
- [8] M. A. L. Thatachar and V. V. Phananskar, "Learning Global Maximum with Parameterized Learning Automata", IEEE Trans. Neural Networks, Vol. 6, No. 2, March 1995.
- [9] Kumpan S. Narendra and M. A. L. Thatachar, Learning Automata: An Introduction, New Jersey, Prentice Hall, 1989.
- [10] K. Najim and A. S. Poznyak, Learning Automata Theory and Applications, Elsevier Science Ltd, 1994.
- [11] P. Mars, J.R. Chen and R. Nombiar, Learning Algorithms Theory and Applications in Signal Processing, Control and Communications, CRC Press, New York, 1998.
- [12] S. Lakshimavarhan, Learning Algorithms Theory and Applications, New York, Springer - Verlag, 1981.
- [13] J. T. Tou and R. C. Gonzalez, Pattern Recognition Principals, Reading, MA: Addison-Welsey, 1974.
- [14] M. R. Meybodi and S. Lakshimavarhan, "On a class of Learning Algorithms Which Have a Symmetric Behavior Under Success and Failure", Springer - Verlag Lecture Notes in Statistics, PP. 145-155, 1984.
- [15] M.R. Meybodi, "Results on Strongly Absolutely Expedient Learning Automata", Proceedings of Inference Conference 86, ed. D.R. Moates and R. Butrick (Athens, Ohio: Ohio University Press, 1987), pp. 197-209.
- [16] G.R. Rezaei and M.R. Meybodi, "Automatic Determination of Discriminant Functions for Pattern Recognition", Proceedings of Fourth International Conference of Computer Society of Iran, 1998.
- [17] N. Baba and H. Handa, "Utilization of Hierarchical Structure Stochastic Automata for the Back Propagation Method with Momentum", proc. of IEEE ICNN-95, pp. 389-393, 1995.



The 7th Iranian Conference On Electrical Engineering

Iran Telecommunication Research Center 17-19 May 1999

Proceedings Computer

