سومین کنفرانس بین المللی سالانه کامپیوتر انجمن کامپیوتر ایران
۲ الی ۴ دیماه ۱۳۷۶
دانشکده مهندسی کامپیوتر - دانشگاه علم و صنعت ایران

تنظیم خودکار پارامترهای مدل یادگیری Q با استفاده از اتوماتونهای یادگیر با ساختار ثابت

سیامک حجت
کارشناس ارشد

محمدرضا میبدی
دانشیار دانشکدهٔ مهندسی کامپیوتر

دانشکدهٔ مهندسی کامپیوتر، دانشگاه صنعتی امیرکبیر

چکیده

مدل یادگیری Q [1] پارامترهای متعددی دارد و عملکرد بهینهٔ این مدل به انتخاب مناسب این پارامترها وابسته است. مقادیر این پارامترها معمولاً با سعی و خطا و توسط طراح مدل انتخاب می‌شوند و در طول یادگیری بصورت مقادیر ثابت مورد استفاده قرار می‌گیرند. اما تعیین پارامترها به این روش اولاً بسیار نادقیق و وقتگیر است و ثانیاً انعطاف پذیری لازم را ندارد. در [2] یک راه حل برای تنظیم خودکار پارامترهای مدلهای یادگیر توسط یک یا چند مأمور یادگیر دیگر پیشنهاد شده است و نتایج استفاده از یک *اتوماتون یادگیر با ساختار متغییر* در تنظیم این پارامترها بررسی شده است. در [3] نیز استفاده از چند اتوماتون یادگیر با ساختار متغییر برای تنظیم پارامترهای مدل Q بررسی شده است. هدف از این مقاله بررسی نتایج تنظیم پارامترهای مدل با استفاده از *اتوماتونهای یادگیر با ساختار ثابت* و مقایسهٔ نتایج بدست آمده با آزمایشهای انجام شده در [2] و [3] می‌باشد.

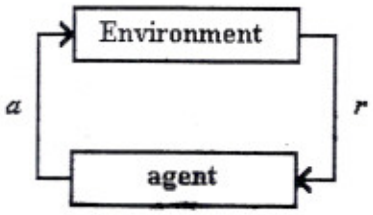
واژه‌های کلیدی: *Reinforcement Learning, Q_Learning, Statistical Clustering, Learning Automata*

۱. مقدمه

در یادگیری تقویتی[1] [4] یک یادگیرنده در هر لحظه از مجموعهٔ فعالیتهای ممکن فعالیتی را انتخاب کرده و در محیط إعمال می‌کند. با إعمال هر فعالیت پسخوری[2] از محیط دریافت می‌شود (شکل ۱). هدف یادگیرنده یافتن یک استراتژی انتخاب فعالیت برای ماکزیمم کردن پسخورهای دریافت شده از محیط در طول زمان است. در چند دههٔ گذشته الگوریتمهای یادگیری تقویتی متنوعی طراحی و پیاده‌سازی شده‌اند. همگی این الگوریتمها شامل پارامترهای متعددی می‌باشند و رفتار آنها به شدت به چگونگی انتخاب مقادیر این پارامترها وابستگی دارد. برای کاهش این

[1] Reinforcement Learning
[2] Feedback

وابستگی در [2] روشی برای تنظیم پارامترهای یک الگوریتم یادگیری تقویتی پیچیده توسط الگوریتمهای یادگیری ساده‌تر پیشنهاد شد. در [2] از مدل Q به عنوان االگوریتم یادگیری تقویتی پیچیده استفاده گشت و از اتوماتونهای یادگیری تقویتی با ساختار ثابت برای تنظیم یک پارامتر این مدل استفاده شد. در اینجا نتایج استفاده از اتوماتونهای ساختار ثابت برای تنظیم چند پارامتر مدل Q بررسی می‌شود و نتایج حاصله با کارهای قبلی مقایسه خواهد شد.



شکل ۱. رابطهٔ مامور با محیط.

۲. مدل یادگیری Q

مدل یادگیری Q یک نوع *یادگیری تقویتی* است. مدل Q یک تکنیک برای انتشار پسخورهای بلافاصله روی توالی فعالیتها می‌دهد. این مدل معمولاً به همراه روشهای دسته *بندی آماری*[3][5] استفاده می‌شود. از مدل Q در کاربردهایی نظیر یادگیری رفتارهای جدید در روباتهای رفتاری[4][6] و کنترل انیماتها[5][7] استفاده شده است. در مدل Q از یک ساختمان داده بنام Q برای تخمین سودمندی اِعمال فعالیت a در وضعیت حس شدهٔ S استفاده می‌شود: $Q(S,a)$. ابتدا $Q(S,a)$ برای تمام فعالیتهای a و وضعیتهای S برابر با صفر فرض می‌شود. سپس با اِعمال هر فعالیت a در وضعیت X و دریافت پسخور بلافاصلهٔ r مقدار $Q(X,a)$ با فرمول زیر بهنگام می‌شود:

$$Q(X,a) \leftarrow Q(X,a) + \lambda\,(r + \gamma\ e(Y) - Q(X,a \qquad (۱)$$

در فرمول بالا Y وضعیت بعدی محیط (پس از اِعمال فعالیت a در وضعیت X است) و $e(Y)$ سودمندی وضعیت Y می‌باشد که با فرمول زیر محاسبه می‌شود: (m تعداد فعالیتهاست)

$$e(Y) \leftarrow \text{maximum } Q(Y,i) \text{ over all actions } i,\ (i = 1,2,\ldots,m) \qquad (۲)$$

پارامتر λ ($0 \le \lambda \le 1$) میزان اصلاح خطا برای Q را تعیین می‌کند و پارامتر γ ($0 \le \gamma \le 1$) میزان صرفنظر کردن از سودمندی وضعیت نتیجه شده را مشخص می‌کند. در مدل Q تابع ارزیابی برای انتخاب بهترین فعالیت در وضعیت S باید فعالیتی را برگزیند که مقدار $Q(S,a)$ را ماکزیمم کند. این سیاست انتخاب فعالیتها هرگز تمام فعالیتهای ممکن را امتحان نمی‌کند و معمولاً منجر به انتخاب غیر بهینهٔ فعالیتها می‌شود. بنا بر این لازم است در درصدی از مواقع (θ) انتخاب فعالیت بطور تصادفی انجام گیرد[6].

زمانی که تعداد وضعیتهای قابل تجربه زیاد باشد بجای ذخیره کردن تمام تجربیات بهتر است آنها را دسته‌بندی کرد. در دسته‌بندی آماری تمام تجربیات مشابه در یک دسته قرار می‌گیرند و بجای ذخیره کردن همهٔ آنها تنها اطلاعاتی آماری از آنها نگهداری می‌شود. در این تکنیک هر تجربهٔ جدید با دسته‌های موجود مقایسه شده و در

---

[3] Statistical Clustering

[4] Behavior Based Robots

[5] Animats

دسته (یا دسته‌های) مشابه ادغام می‌شود. در صورتیکه تجربهٔ جدید مشابه هیچکدام از دسته‌های موجود نباشد یک دستهٔ جدید برای آن تجربه ایجاد خواهد شد.

هر دسته نمایانگر گروهی از وضعیتهای *مشابه* است. یک دسته را میتوان با $n+2$ تایی $C =< (z_1,o_1),(z_2,o_2),\ldots,(z_n,o_n),Q_c,M_c >$ نشان داد که در آن z_i و o_i تعداد دفعاتی است که بیت‌ام از وضعیت S در دستهٔ C، 0 یا 1 بوده است. n تعداد بیتهای یک وضعیت است، Q_c مقدار Q دسته را مشخص می‌کند و M_c نمایانگر تعداد تجربیاتی است که در این دسته قرار گرفته‌اند. حال اگر $p(S \in C)$ احتمال شرطی قرار گرفتن S باشد برای اینکه وضعیت S در دستهٔ C قرار گیرد باید داشته باشیم:

$$p(S \in C | s_1 = v_1, s_2 = v_2, \ldots, s_n = v_n) > \varepsilon \qquad (۳)$$

$$|Q_c - Q_s| < \delta \qquad (۴)$$

نامعادلات (۳) و (۴) تضمین می‌کنند که اولاً مشابهت وضعیت S با دستهٔ C از یک مقدار آستانه‌ای (ε) بیشتر باشد و ثانیاً Q محاسبه شده برای وضعیت S نسبت به مقدار Q ذخیره شده در دستهٔ C از یک مقدار ثابت آستانه‌ای (δ) کمتر باشد. بعد از اینکه مشخص شد که وضعیت S در دستهٔ C قرار می‌گیرد از آن برای بهنگام سازی دسته استفاده خواهد شد. فرض کنید: $C =< (z_1,o_1),(z_2,o_2),\ldots,(z_n,o_n),Q_c,M_c >$، اگر C_u نمایانگر دستهٔ C پس از بهنگام‌سازی باشد و داشته باشیم: $C_u =< (z_{1u},o_{1u}),(z_{2u},o_{2u}),\ldots,(z_{nu},o_{nu}),Q_{cu},M_{cu} >$، برای هر بیت i از وضعیت S که برابر با 1 باشد خواهیم داشت: $(s_i = 1)$ $o_i = 1 + \mu o_i$, $z_{iu} = \mu z_i$ و برای هر بیت i از وضعیت S که برابر با 0 باشد خواهیم داشت: $(s_i = 0)$ $o_i = \mu o_i$, $z_{iu} = 1 + \mu z_i$. در اینجا μ عددی حقیقی و بین صفر و یک است که برای افزایش اهمیت تجارب جدید بکار می‌رود. اگر μ=1 باشد اهمیت تجارب جدید در نظر گرفته نخواهد شد. μ را معمولاً از فرمول $\mu = (2K-1)/2K$ بدست می‌آورند که در آن K عددی صحیح است (از K برای ایجاد دسته های جدید استفاده خواهد شد).

فرض کنید در وضعیت S فعالیت a اِعمال شده باشد و مقدار محاسبه شدهٔ Q برای آن برابر با $Q(S,a)$ باشد آنگاه برای ساختن Q_{cu} می‌توان از مجموع Q_c و $Q(S,a)$ استفاده کرد. معمولاً در این جمع از M_c بعنوان وزن استفاده می‌شود:

$$Q_{cu} = Q_u\left(\frac{M_c}{M_c+1}\right) + Q(S,a)\left(\frac{1}{M_c+1}\right) \qquad (۵)$$

همچنین تعداد تجربیات دستهٔ C_u بصورت $M_{cu} = M_c + 1$ بهنگام می‌شود.

اگر یک وضعیت S مشابه هیچ یک از دسته‌های موجود نباشد باید یک دستهٔ جدید C_{new} برای آن ساخته شود. برای اینکار ابتدا یک دستهٔ خالی به شکل ایجاد می‌گردد: $C =< (z_1,o_1),(z_2,o_2),\ldots,(z_n,o_n),Q_c,M_c >$ که در آن: $M_c = 0$, $Q_c = 0$, $z_i = o_i = K$ و سپس دستهٔ خالی فوق با وضعیت S ادغام می‌شود و دستهٔ C_{new} را می‌سازد.

گاهی دو دسته به‌اندازه‌ای مشابه یکدیگرند که می‌توانند در هم ادغام شوند. برای محاسبهٔ تشابه دو دسته می‌توان از اندازه‌گیری فاصلهٔ بین دو دسته استفاده کرد. دو دستهٔ C_1 و C_2 تنها زمانی با هم ادغام می‌شوند که اولاً فاصلهٔ آنها کمتر از مقدار ثابت ρ باشد و ثانیا مقادیر Q دو دسته اختلافی کمتر از δ داشته باشد. یعنی:

$$distance(C_1,C_2) < \rho \qquad (۶)$$

$$|Q_{c_1} - Q_{c_2}| < \delta \qquad (۷)$$

حال اگر دو دستهٔ C_a و C_b را داشته باشیم و این دو دسته به اندازهٔ کافی مشابه باشند (یعنی روابط ۶ و ۷ برای آنها صادق باشد) آنگاه دو دسته در هم ادغام می‌شوند و دستهٔ جدید C را بوجود می‌آورند. $C =< (z_1, o_1), (z_2, o_2), \ldots, (z_n, o_n), Q_c, M_c >$، که عناصر آن بصورت زیر ساخته می‌شوند:

$$z_{ic} = z_{ia}\left(\frac{M_a}{M_a + M_b}\right) + z_{ib}\left(\frac{M_b}{M_a + M_b}\right) \qquad (۸)$$

$$o_{ic} = o_{ia}\left(\frac{M_a}{M_a + M_b}\right) + o_{ib}\left(\frac{M_b}{M_a + M_b}\right) \qquad (۹)$$

$$Q_c = Q_a\left(\frac{M_a}{M_a + M_b}\right) + Q_b\left(\frac{M_b}{M_a + M_b}\right) \qquad (۱۰)$$

$$M_c = M_a + M_b \qquad (۱۱)$$

در یادگیری Q باید در هر وضعیت S فعالیت a را به گونه‌ای انتخاب کرد که مقدار $Q(S,x)$ را به ازای تمام فعالیتهای ممکن ماکزیمم کند ($x = 1,2,\ldots,m$). برای محاسبهٔ $Q(S,x)$ از روی دسته‌های موجود میتوان از فرمول زیر استفاده کرد:

$$Q(S,x) = \frac{\sum_{C \in C_x} [Q_c \times P(S \in C | S_1 = V_1, \ldots, S_n = V_n)}{\sum_{C \in C_x} [P(S \in C | S_1 = V_1, \ldots, S_n = V_n)} \qquad (۱۲)$$

در عبارت فوق C_x مجموعهٔ دسته‌هایی می‌باشد که در آنها فعالیت x انتخاب شده است. صورت کسر بالا مجموع وزن‌دار مقادیر Q برای عناصر Cx است (از احتمال قرار گرفتن وضعیت S در دستهٔ C بعنوان وزن این جمع استفاده شده است). مخرج کسر نیز برای نرمال کردن عبارت می‌باشد.

مدل Q را می‌توان به صورت زیر خلاصه کرد:

۱. مقادیر ثابتی برای پارامترهای Q $(\theta, \gamma, \lambda)$ و پارامترهای دسته بندی $(\rho, K, \varepsilon, \delta)$ در نظر بگیرید.

۲. برای همیشه:

الف) وضعیت فعلی محیط را مشاهده کنید (X).

ب) در θ درصد از مواقع فعالیتی را به طور تصادفی انتخاب کنید. در مواقع دیگر فعالیتی را انتخاب کنید که مقدار $Q(X,a)$ را ماکزیمم کند.

پ) فعالیت a را در محیط اِعمال کنید. فرض کنید وضعیت جدید Y باشد و پسخور بلافاصلهٔ اِعمال این فعالیت r باشد.

ت) میزان $Q(X,a)$ را با معادلهٔ (۱) بهنگام کنید.

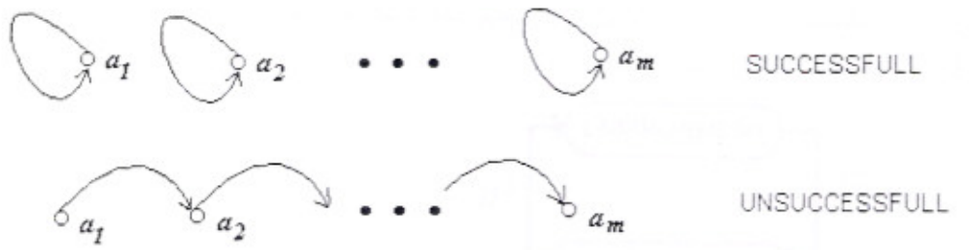
ث) اگر دسته‌ای مانند C وجود داشت که به همراه X در نامعادلات (۳) و (۴) صدق کند، وضعیت X را در دستهٔ C ادغام کنید. در غیر اینصورت دستهٔ جدیدی از روی X ایجاد نمایید.

ج) هر دو دستهٔ C_1 و C_2 که در نامعادلات (۶) و (۷) صدق می‌کنند را در هم ادغام کنید.

۲. اتوماتونهای یادگیر [9][8]

در اتوماتونهای یادگیر یک تصمیم گیرنده در یک محیط تصادفی فعالیتهایی را بر اساس پسخورهای دریافت شده از محیط انتخاب می‌کند. تصمیم گیرنده در چنین محیطی بر اساس پسخورهای دریافت شده از محیط و یک استراتژی انتخاب، فعالیتهایی را انتخاب کرده و در محیط اعمال می‌کند. اتوماتونهای یادگیر بر اساس ثابت یا پویا بودن استراتژی انتخاب فعالیتشان به دو دستهٔ اتوماتونهای ساختار ثابت و ساختار متغییر تقسیم می‌شوند. در [2] چند اتوماتون ساختار متغییر بنام LRP، LRI و LRεP (یا بطور کلی اتوماتونهای LA) معرفی شدند. بنا بر این در اینجا تنها اتوماتونهای ساختار ثابت معرفی خواهند شد.

یک استراتژی ساده برای انتخاب فعالیتها این است که اتوماتون تا زمانیکه پسخور موفقیت آمیز دریافت می‌کند به انتخاب فعالیتی که قبلاً انتخاب کرده است ادامه دهد اما هرگاه پسخور ناموفق بود فعالیت دیگری را انتخاب کند. در شکل ۲ این استراتژی انتخاب فعالیت مشخص شده است (در شکل m تعداد فعالیتهاست). اتوماتونی که به این ترتیب عمل می‌کند را اتوماتون ثابت یا FA می‌نامند.



شکل ۲

اتوماتونهای تسلتسین (مدلهای L و G) *[10,11]*، کرینسکی (مدل *K1*) *[12]* و کریلوف (مدل *K2*) *[13]* نیز از انواع دیگر اتوماتونهای ساختار ثابت می‌باشند. این اتوماتونها تعداد پسخورهای موفق و ناموفق حاصل از هر فعالیت را ثبت می‌کنند. برای این منظور اتوماتونها از یک حافظه با عمق ثابت استفاده می‌کنند که عمق این حافظه را با N نمایش می‌دهند. در اتوماتون L پس از انتخاب یک فعالیت تا زمانی که تعداد پسخورهای موفقیت آمیز بیشتر از تعداد پسخورهای ناموفق باشد، فعالیت انتخاب شده در لحظهٔ قبل مجدداً انتخاب خواهد شد. اتوماتون G نیز به همین منوال عمل می‌کند با این تفاوت که این اتوماتون هرگاه پسخور ناموفق دریافت کند بلافاصله فعالیت دیگری را برای اعمال انتخاب خواهد کرد. اتوماتون K1 نیز رفتاری مشابه اتوماتون L دارد اما با این تفاوت که این اتوماتون تنها با دریافت N بار پسخور ناموفق فعالیتی که انتخاب کرده است را تغییر می‌دهد. اتوماتون K2 زمانی که پسخور محیط موفقیت آمیز باشد درست مانند اتوماتون L رفتار می‌کند. اما در قبال دریافت پسخور ناموفق تنها به احتمال ۵۰٪ مانند اتوماتون L عمل می‌کند و در ۵۰٪ دیگر از مواقع با پسخور ناموفق نیز مانند پسخور موفق برخورد می‌کند. برای مطالعهٔ بیشتر اتوماتونهای ساختار ثابت به [8,9,22] مراجعه کنید.

۳. تنظیم پارامترهای مدلهای Q [14][3][2]

برای تنظیم پارامترهای مدل Q با استفاده از اتوماتونهای یادگیری این اتوماتونها باید مجهز به فعالیتهایی جهت تغییر مقادیر پارامترهای مدل Q باشند. شمای کلی روش تنظیم پارامترها توسط اتوماتونها به این صورت است:

۱- اتوماتونها با اِعمال فعالیتهای خود مقدار پارامترهای مدل Q را تغییر می‌دهند.

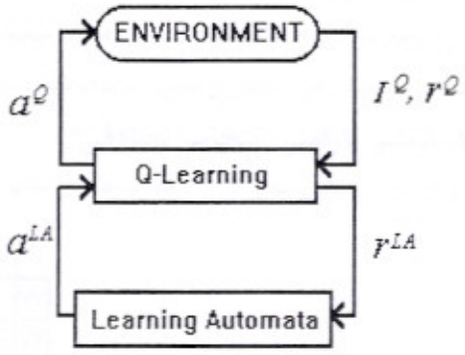
۲- مدل Q در طول یک پریود زمانی (*period*) از این مقادیر استفاده می‌کند.

۳- در انتهای پریود میزان کارا بودن هر یک از پارامترها بر اساس عملکـرد مـدل ارزیـابی می‌شـود و بصـورت پسخورهایی در اختیار اتوماتونهای یادگیر قرار می‌گیرند.

۵- اتوماتونها با استفاده از پسخورهای دریافتی احتمال انتخاب فعالیتهای خود را بهنگام می‌کنند.

۶- تکرار عملیات.

به این ترتیب مدل Q نقش محیط اتوماتون یادگیر را ایفا می‌کند (شکل ۳). به عنوان مثال اگر از یـک اتومـاتون یادگیر برای تنظیم مقدار θ استفاده شود. برای تعیین مقدار مناسب این پارامتر که عددی بین صفر و یک اسـت بـازهٔ *[0,1]* به بازه‌های کوچکتری تقسیم می‌شود. فرض کنید بازه‌ها با فاصلهٔ یک دهم انتخاب شوند (یعنی ده بازه) و هر بازه با کوچکترین عدد آن بازه مشخص شود. وظیفهٔ اتوماتون یادگیر انتخاب بازه‌ای است که عدد مشخص کنندهٔ آن مناسبترین مقدار را برای پارامتر انتخاب تصادفی یادگیری Q داشته باشـد. بـرای ایـن منظـور بـاید از یـک اتومـاتون یادگیر که مجهز به ده فعالیت (برای انتخاب هر کدام از بازه‌ها) است استفاده کرد. اتوماتون یـادگیر پـس از انتخـاب یک مقدار برای این پارامتر آنرا در اختیار مدل Q می‌گذارد و مدل Q در یک پریود زمانی (مثلاً ده واحـد زمـانی) از این مقدار استفاده خواهد کرد. پس از انقضای این پریود میزان کارایی پارامتر تنظیم شده توسط یک تابع ارزیابی بـه اتوماتون یادگیر پسخور می‌گردد و مجدداً این عملیات تکرار می‌شود.



شکل ۳. رابطهٔ مدل Q با اتوماتون یادگیر و محیط.

تابع ارزیابی مورد استفاده برای اتوماتون تنظیم کنندهٔ پارامتر θ به صورت زیر تعریف شده اسـت: (ایـن روابـط پسخور اتوماتون در زمان t را مشخص می‌کنند، فرض شده که اتوماتون در زمان T پسخور بیشینه را دریـافت کـرده باشد)

$$\text{IF } \sum_{i=t-p}^{t} r_i^Q \geq \sum_{i=T-p}^{T} r_i^Q \text{ THEN } r^{LA} = MAX_{i=1}^{t}(r_i^Q) \qquad (۱۳)$$

$$\text{ELSE IF } \sum_{i=t-p}^{t} \frac{r_i^Q}{p} \leq \sum_{i=1}^{t} \frac{r_i^Q}{t} \text{ THEN } r^{LA} = MIN_{i=1}^{t}(r_i^Q) \qquad (۱۴)$$

$$\text{ELSE } r^{LA} = MIN_{i=1}^{t}(r_i^Q) + \frac{\sum_{i=t-p}^{t} \frac{r_i^Q}{p} - \sum_{i=1}^{t} \frac{r_i^Q}{t}}{\sum_{i=T-p}^{T} \frac{r_i^Q}{p} - \sum_{i=1}^{t} \frac{r_i^Q}{t}} \times \left(MAX_{i=1}^{t}(r_i^Q) - MIN_{i=1}^{t}(r_i^Q) \right) \qquad (۱۵)$$

سایر پارامترهای مدل Q علاوه بر عملکرد در تعداد دسته‌های مدل نیز مؤثرنـد. در تـابع ارزیـابی اتوماتونهـای تنظیم کنندهٔ این پارامترها درصورتیکه تعداد دسته‌ها در یک پریود تغییر نکنند پسخور بـا روابـط بـالا محاسـبه شـده است. اما اگر تعداد دسته‌ها در یک پریود زمانی افزایش یابد و میانگین پسخورهای دریافتی در این پریود نسـبت بـه میانگین کل بهتر نشود، آنگاه پسخور کمینه و درصورتیکه تعداد دسته ها در یک پریود زمانی کاهش یابد و میـانگین

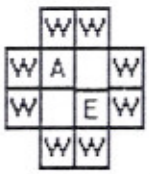
پسخورهای دریافتی در این پریود نسبت به پریود قبلی کاهش نیابد پسخور بیشینه برای اتوماتونها در نظر گرفته شده است.

۴. آزمایشها و نتایج

در آزمایشهایی که شرح آنها خواهد آمد از مدلهای Q، Q_FA، Q_L، Q_G، Q_K1 و Q_K2 برای کنترل یک مأمور یادگیرنده در محیطهایی مصنوعی استفاده شده است. در این آزمایشها محیط یک صفحهٔ شطرنجی شکل است که در هر خانهٔ آن ممکن است یک شیء وجود داشته باشد. مأمور یادگیرنده مجهز به چهار حس برای مشاهدهٔ اجسام خانه‌های مجاور خود (بالا، راست، پایین و چپ) و چهار فعالیت برای حرکت به این خانه‌های مجاور است. این فعالیتها تنها زمانی موجب حرکت به یک خانهٔ مجاور می‌شوند که در آن خانه شیئی وجود نداشته باشد. در این صورت حرکت به خانهٔ مجاور انجام می‌گردد و مأمور پسخور صفر (r=0) را از محیط دریافت می‌کند. در هر خانه از محیط یکی از اشیاء دیوار، غذا، زهر یا دشمن می‌تواند وجود داشته باشد. اِعمال یک فعالیت برای حرکت به خانه‌ای که در آن دیوار، غذا، زهر یا دشمن وجود دارد به ترتیب پسخورهای 40-، 100+، 100- و 100- را ایجاد می‌کند. اشیاء دیوار، غذا و زهر نمی‌توانند در محیط حرکت کنند اما دشمن می‌تواند در هر واحد زمانی به یکی از چهار خانهٔ مجاور (بالا، راست، پایین یا چپ) حرکت کند (حرکت دشمن به خانه‌های مجاور بطور تصادفی انجام می‌گیرد). برای ساده کردن توصیف محیط در آزمایشها از حروف W، F، P، E و A به ترتیب برای نمایش دیوار، غذا، زهر، دشمن و مامور استفاده شده است.

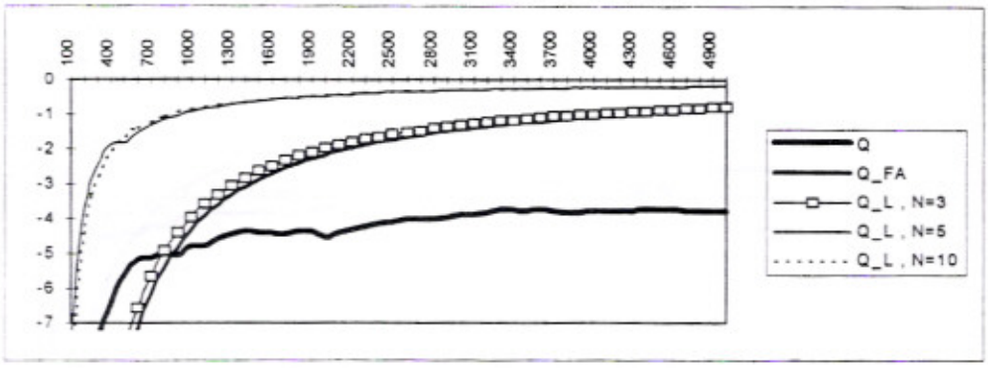
۴-۱. تنظیم پارامتر 'انتخاب تصادفی'

برای بررسی تاثیر تنظیم θ در انعطاف پذیری مدلها از یک محیط ساده به شکل زیر استفاده شده است:

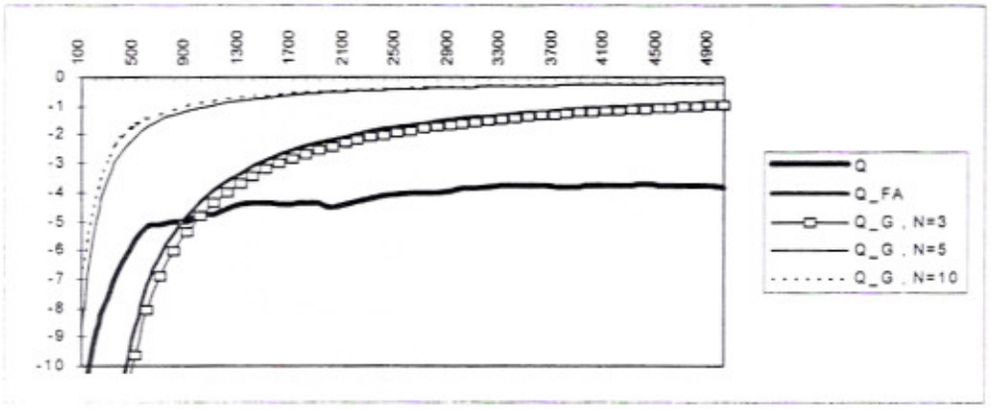


شکل ۴. محیط آزمایش انعطاف پذیری.

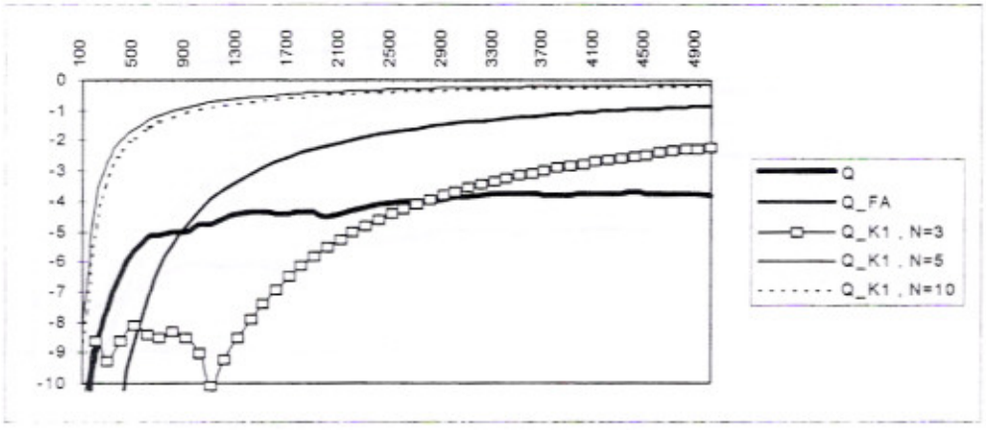
در این آزمایشها مقادیر پارامترهای ثابت مدلهای Q با پارامتر θ "ثابت" و "تنظیم شده" بدین صورت انتخاب شده‌اند: $\gamma = 0.9$، $\lambda = 0.5$، $\rho = 2$، $\varepsilon = 0.000001$، $\delta = 10$، $k = 5$، $\theta = 0.1$ (فقط برای مدلهای Q با پارامترهای ثابت) و $period = 1$ (فقط برای مدلهای Q با پارامتر تنظیم شده). در نمودارهای ۱ تا ۴ به ترتیب عملکرد هر کدام از مدلهای *Q_L*، *Q_G*، *Q_K1* و *Q_K2* با عمقهای حافظهٔ مختلف با مدلهای *Q* و *Q_FA* مقایسه شده‌اند. مطابق نمودارها مشخص است که اولاً تنظیم پارامترها باعث افزایش عملکرد شده است (دلایل افزایش عملکرد با تنظیم پارامترها در [2] و [3] بررسی شده است) و ثانیاً افزایش عمق حافظه در مدلها موجب عملکرد بهتر آنها شده است. در نمودار ۵ نحوهٔ تغییر عملکرد با افزایش عمق حافظه آورده شده است. چنانچه ملاحظه می‌شود برای این محیط عمق حافظه بین ۵ تا ۱۰ بهترین نتیجه را در بر داشته است. نمودار ۶ عملکرد مدلهای *Q_L*، *Q_G*، *Q_K1* و *Q_K2* با عمق حافظهٔ ۵ را با یکدیگر و با مدلهای *Q*، *Q_FA* و مدل *Q_LRP* (مدلی که پارامترهای آن با اتوماتون ساختار متغییری بنام *LRP* تنظیم می‌شود) [2] مقایسه کرده است. چنانچه ملاحظه می‌شود مدلهای *Q_L*، *Q_G*، *Q_K1* و *Q_K2* عملکرد بهتری نسبت به سایر مدلها نشان داده‌اند و مطابق نمودارهای ۶ و ۷ عملکرد این مدلها در این محیط نسبتاً مشابه بوده است.



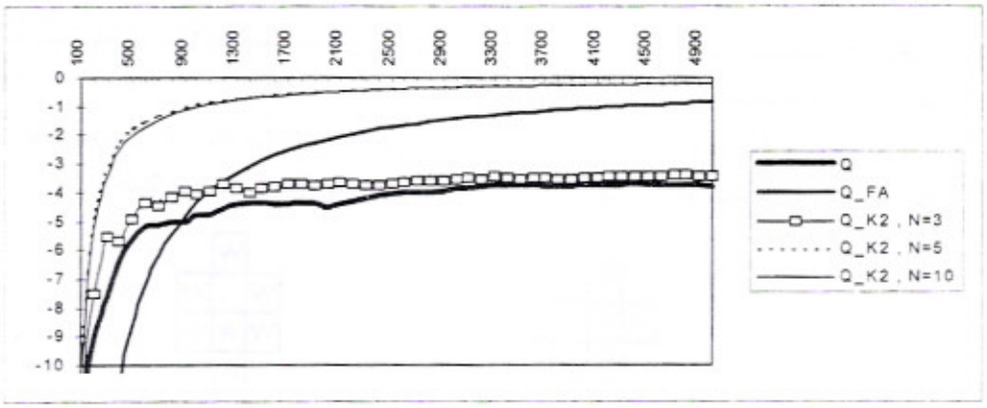
نمودار ۱. عملکرد مدل Q_L با عمقهای حافظهٔ مختلف.



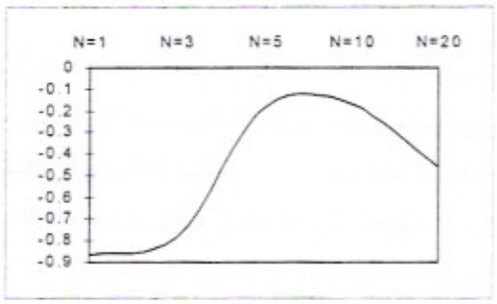
نمودار ۲. عملکرد مدل Q_G با عمقهای حافظهٔ مختلف.



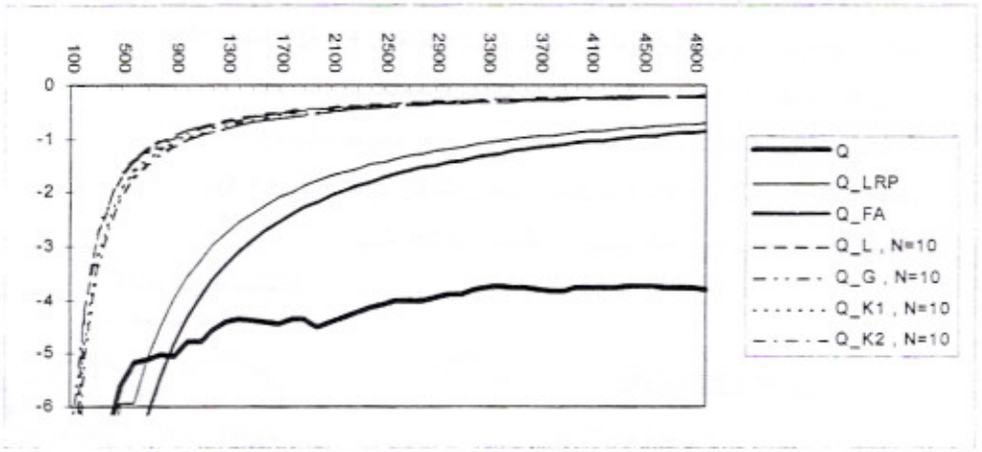
نمودار ۳. عملکرد مدل Q_K1 با عمقهای حافظهٔ مختلف.



نمودار ۴. عملکرد مدل Q_K2 با عمقهای حافظهٔ مختلف.



نمودار ۵. تغییرات عملکرد مدل Q_L با تغییر عمق حافظه.



نمودار ۶. مقایسهٔ عملکرد مدلهای مختلف با یکدیگر.

۴-۱-۱. مقایسهٔ انعطاف پذیری

منظور از *انعطاف پذیری* مدلهای یادگیری قابلیت انطباق آنها با محیط است. درجهٔ انعطاف پذیری مدل یادگیری را می‌توان متناسب با زمان لازم برای یافتن بهترین فعالیت برای اِعمال در یک وضعیت تجربه نشده دانست. برای اندازه‌گیری درجهٔ انعطاف پذیری مدل در برخورد با وضعیتهای جدید از محیط پایه با پیکربندی‌های شکل ۵ استفاده شده است. در این آزمایش ابتدا از محیط شکل ۵-الف برای خبره کردن یادگیرنده استفاده گشته است. برای این منظور یادگیرنده به مدت ۲۰۰۰ واحد زمانی در این محیط قرار گرفته است. سپس یادگیرنده در محیط شکل ۵-ب قرار داده شده و در این وضعیت جدید زمان لازم برای انتخاب بهترین فعالیت (حرکت به بالا)

اندازه‌گیری شده است. برای آزمایش انجام گرفته مقادیر پارامترهای ثابت مدلهای Q با "پارامتر ثابت" و "تنظیم شده" بدین صورت انتخاب شده‌اند: $\gamma = 0.9$، $\lambda = 0.5$ ، $\rho = 2$، $\varepsilon = 0.000001$، $\delta = 10$، $k = 5$، $\theta = 0.1$ (فقط برای مدل Q با پارامتر ثابت) و $period = 1$ (برای مدلهای Q با پارامتر تنظیم شده). در آزمایشهای این بخش زمان متوسط یادگیری وضعیت جدید با معدل گیری از پنج بار اجرای هر آزمایش محاسبه گشته است.

محیط ب　　محیط الف

شکل ۵. پیکربندی‌های محیط پایه در آزمایش انعطاف پذیری

در این آزمایش از مدلها بعنوان مأمور یادگیر محیط شکل ۵ استفاده گشته است. در اینجا مأمور یادگیرنده در محیط ۵-الف یاد می‌گیرد که انتخاب فعالیتهای "حرکت به بالا" و "حرکت به پایین" در هر شرایط پسخور ناموفق دریافت خواهد کرد اما هرگاه خانهٔ سمت راست (یا چپ) مأمور خالی باشد انتخاب فعالیت "حرکت به راست" (یا چپ) پسخور موفق دریافت خواهد نمود. اما وقتی مأمور در محیط ۵-ب قرار می‌گیرد باید یاد بگیرد که در وضعیت جدید خود فعالیت "حرکت به بالا" را انتخاب کند. بنابر این آنچه که مأمور در محیط ۵-الف یادگرفته است (دسته‌های کشف شده در محیط ۵-الف) برای یادگیری وضعیت جدید در محیط ۵-ب کافی نمی‌باشد و درنتیجه یادگیری بهترین فعالیت برای اِعمال در محیط ۵-ب فقط با انتخاب تصادفی فعالیتها امکان پذیر خواهد بود. شکل ۶ وضعیت درونی اولیهٔ اتوماتون تنظیم کنندهٔ θ در مدل Q_FA در محیط ۵-ب را با دایرهٔ توپر نشان می‌دهد (این وضعیت پس از ۲۰۰۰ واحد زمانی شبیه‌سازی در محیط ۵-الف ایجاد شده است). با توجه به این شکل نرخ تغییر پارامتر θ با دریافت پسخورهای ناموفق متوالی در محیط جدید در مدل Q_FA بیشتر از سایر مدلهاست. مطابق شکل ۶ اتوماتون FA با دریافت هر پسخور ناموفق پارامتر θ در Q_FA را تغییر خواهد داد و همانطور که در جدول ۱ هم پیداست این مدل بیشترین انعطاف را داشته است.

شکل ۶. وضعیت درونی اولیهٔ اتوماتون FA از مدل Q_FA در محیط ۱۰-ب

پس از مدل Q_FA مدل Q_L بیشترین نرخ تغییر پارامتر θ را خواهد داشت. اتوماتون L در N واحد زمانی اولیه با دریافت پسخورهای ناموفق تغییری در پارامتر نخواهد داد ولی از آن پس با دریافت هر پسخور ناموفق پارامتر θ در مدل Q_L را تغییر خواهد داد. اتوماتون G برای هر تغییر در پارامتر θ نیاز به دریافت N پسخور ناموفق دارد و بنا بر این مدل Q_G کمترین انعطاف را نشان می‌دهد. از آنجایی که رفتار مدل Q_K1 در هنگام دریافت پسخورهای ناموفق معادل رفتار مدل Q_L است، نتایج آزمایشها برای آن نیز مشابه مدل Q_L بوده است. با استدلالهای مشابهی می‌توان انتظار داشت که انعطاف پذیری مدل Q_K2 بسیار پایین باشد. همچنین واضح است که افزایش N در اتوماتونهای ساختار ثابت باعث افزایش زمان یادگیری وضعیت جدید در مدلهای Q_L، Q_G، Q_K1

و *Q_K2* خواهد شد. باید توجه داشت که هرچند انتخاب مقادیر بزرگتر برای θ احتمال بیشتری برای یادگیری وضعیت جدید ایجاد می‌کند اما انتخاب هر مقدار غیر صفر برای آن می‌تواند باعث کشف وضعیت جدید شود و بنا بر این نسبت زمان لازم برای یادگیری یک وضعیت جدید توسط مدل *Q_G* با افزایش N بطور خطی افزایش پیدا نخواهد کرد. لازم به ذکر است که مدل Q (با پارامتر ثابت) بطور متوسط به *32.1* و مدل *Q_LRP* به *14.3* واحد زمانی برای یادگیری وضعیت جدید در محیط ۵-ب نیاز داشته‌اند.

مدل	N	متوسط زمان لازم برای یادگیری وضعیت جدید
Q_FA	1	9.6
Q_L	5	12
	10	18.2
	20	27.2
Q_G	5	12.8
	10	27
	20	54.4
Q_K1	5	10.6
	10	22.3
	20	31.6
Q_K2	5	39.3
	10	67.3
	20	446

جدول ۱. مقایسهٔ زمان یادگیری وضعیت جدید.

۴-۱-۲. مقایسهٔ قدرت اکتشاف

در مدلهای یادگیری از پارامتر انتخاب تصادفی (θ) به منظور کشف وضعیتهای جدید استفاده می‌شود. تکنیک اکتشاف بوسیلهٔ انتخاب تصادفی فعالیتها به تکنیک اکتشافی غیر مستقیم [16][15] معروف است. پیاده سازی این تکنیک اکتشافی، ساده، اما غیر کارا است[17] (در [18] ثابت شده است که در این تکنیک با افزایش تعداد وضعیتها زمان یادگیری به طور نمایی افزایش پیدا می‌کند). بنا بر این معمولاً هنگامی که تعداد وضعیتها زیاد است از تکنیکهای اکتشافی مستقیم[17,19,20,21] استفاده می‌گردد. در این تکنیکها برای اکتشاف محیط از اطلاعات آماری کسب شده در تجربیات استفاده می‌شود (ثابت شده است که با افزایش تعداد وضعیتها زمان یادگیری با این تکنیکها به صورت یک چند جمله‌ای درجه پایین رشد می‌کند). در آزمایشهای انجام شده در این قسمت از یک نمونه از این مدلها بنام *DQ* نیز استفاده شده است [2][14].

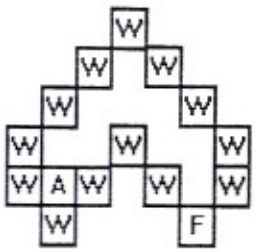
برای گنجاندن قدرت اکتشاف بیشتر در مدلهای Q با پارامترهای تنظیم شونده روش محاسبهٔ پسخور مأمورهای یادگیری تنظیم کنندهٔ پارامتر θ در این مدل به صورت زیرتغییر داده شده است:

(۱۶) $$\text{IF } \sum_{i=t-p}^{t} r_i^Q > \sum_{i=T-p}^{T} r_i^Q \text{ THEN } r^{LA} = MAX_{i=1}^{t}(r_i^Q)$$

(۱۷) $$\text{ELSE } r^{LA} = MIN_{i=1}^{t}(r_i^Q)$$

شرایط بالا تضمین می‌کنند تا در صورت عدم افزایش بیشترین پسخور دریافتی در پریودها مقدار پارامتر تصادفی مدل (و در نتیجه میزان اکتشاف آن) پسخور کمینه دریافت کند. برای مقایسهٔ قدرت اکتشاف در محیط

شکل ۷ استفاده شده است. در این آزمایش قدرت اکتشاف مدلها بر حسب زمان لازم برای یافتن غذا در محیط اندازه گیری شده است.



شکل ۷. آزمایش قدرت اکتشاف.

همانطور که جدول ۲ نشان می‌دهد، مدلهای "Q با پارامتر تنظیم شده توسط اتوماتونهای ساختار ثابت" (Q_L، Q_G، Q_K1 و Q_K2) بیشترین قدرت اکتشاف را در محیط داشته‌اند اما متوسط پسخور دریافتی در طی زمان اکتشاف برای این مدلها بسیار پایین بوده است (عملکرد این مدلها حتی از مأمور تصادفی هم بدتر بوده است). در بین مدلهای آزمایش شده عملکرد مدل Q در محیط بهترین بوده است اما این مدل ضعیفترین قدرت اکتشاف را داشته است. در مقابل مدلهایی که از اتوماتونهای ساختار ثابت برای تنظیم پارامتر استفاده کرده‌اند بهترین اکتشاف را داشته‌اند اما عملکرد مناسبی را در زمان اکتشاف نمایش نداده‌اند (همانطور که پیداست قدرت اکتشاف این مدلها با عمق حافظهٔ آنها نسبت عکس داشته است). در این آزمایش مدلهای Q_LRP و DQ مناسبترین جفت قدرت اکتشاف و عملکرد را نشان داده‌اند.

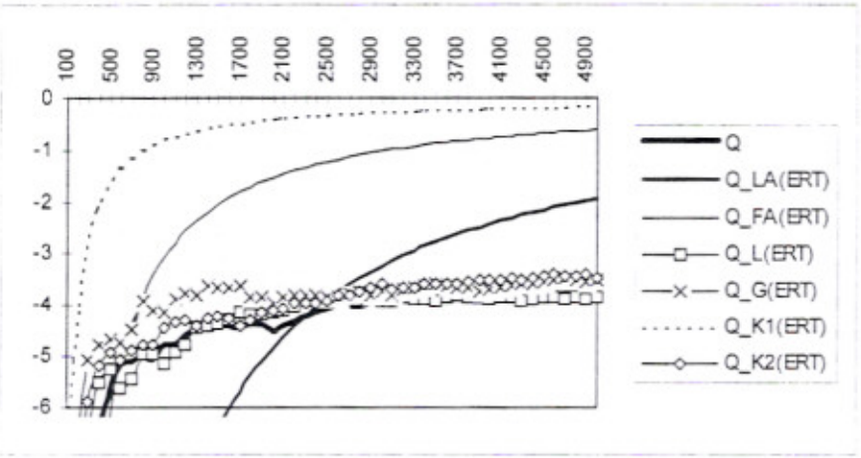
نام مدل	زمان متوسط برای یافتن غذا (قدرت اکتشاف)	متوسط پسخور طی زمان اکتشاف (عملکرد)
R	147	-22
Q	10357	-2.2
Q_L , N=5	147	-22.6
Q_L , N=10	109	-23
Q_L , N=20	85	-24
Q_G , N=5	119	-24.3
Q_G , N=10	112	-24
Q_G , N=20	113	-22.6
Q_K1 , N=5	39	-25.2
Q_K1 , N=10	72	-20.8
Q_K1 , N=20	151	-22.4
Q_K2 , N=5	63	-18.0
Q_K2 , N=10	113	-22
Q_K2 , N=20	147	-22.4
Q_LRP	537	-8.4
DQ	361	-12.8

جدول ۲. مقایسهٔ قدرت اکتشاف مدلها.

۴-۲. تنظیم سایر پارامترهای یادگیری با اتوماتونهای یادگیر

در اینجا رفتار 'مدلهای Q با پارامترهای تنظیم شده توسط چند اتوماتون یادگیر' بررسی شده است. پیکربندی محیط پایه در این آزمایشها همان پیکربندی شکل ۴ است. مشخصات اتوماتونهای تنظیم کنندهٔ پارامتر(های) مدلها در این آزمایشها از این قرار می‌باشد: اتوماتون تنظیم کنندهٔ θ شامل ۱۰ فعالیت است که یکی از مقادیر 0، 0.1، 0.2،

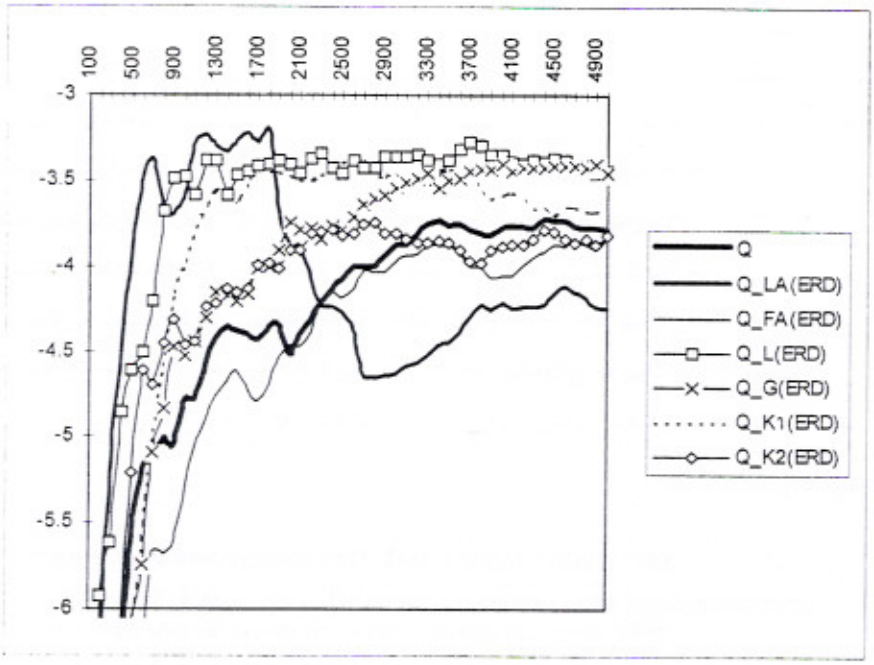
... و 0.9 را برای θ انتخاب می‌کنند. اتوماتون تنظیم کنندهٔ ε دارای ۱۰ فعالیت برای انتخاب یکی از مقادیر 0، 0.00000001، 0.000001، 0.0001، 0.001، 0، 0.2، 0.4، 0.6 و 0.8 برای ε است، اتوماتون تنظیم کنندهٔ ρ یکی از ۵ مقدار 0، 1، 2، 3 و 4 را برای ρ و اتوماتون تنظیم کنندهٔ σ یکی از ۱۰ مقدار 1، 5، 10، 15، ... و 45 را برای σ انتخاب می‌کند. در این آزمایشها پارامترهای K، λ و γ تنظیم نشده‌اند و مقدار این پارامترها برای تمام مدلها ثابت و به ترتیب برابر با 5، 1 و 0 در نظر گرفته شده است. همچنین هرجا پارامترهای θ، ε، ρ و σ تنظیم نشده باشند مقدار آنها به ترتیب برابر با 10، 0.000001، 2 و 20 قرار داده شده است (این مقادیر برای آزمایش مشابهی در [Mahedavan92] پیشنهاد شده‌اند). مقدار *period* برای تمام اتوماتونها برابر با 10 واحد زمانی در نظر گرفته شده است. نتایج آزمایشها در نمودارهای ۸ تا ۱۰ آورده شده است. نتایج آزمایشها نشان دادند که در بعضی موارد مدلها نمی‌توانند بهترین مقادیر را برای پارامترهای تنظیم کننده پیدا کنند و در بهینه‌های محلی قرار می‌گیرند. بعنوان مثال در جدول ۳ مقدار پارامتر θ در انتهای شبیه‌سازی نمودار ۸ آورده شده است همانطور که در این جدول ملاحظه می‌شود پس از خبره شدن مدل در محیط پارامتر θ در بعضی موارد بجای آنکه به مقدار *0* همگرا به مقدار *0.1* همگرا شده است. اما با این وجود عملکرد مدلهای تنظیم کنندهٔ پارامترها در اکثر موارد معادل و یا بهتر از مدل Q با پارامتر ثابت بوده است. بنظر می‌رسد که با در نظر گرفتن توابع بهتری برای تعیین پسخور اتوماتونهای تنظیم کنندهٔ پارامترها و تنظیم عمق حافظهٔ این اتوماتونها بتوان به نتایج بهتری نیز دست یافت.



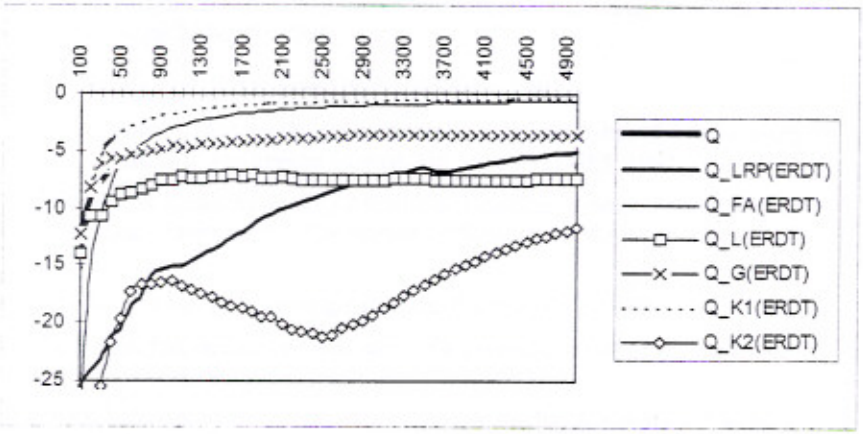
نمودار ۸. مقایسهٔ عملکرد مدلهای مختلف با تنظیم همزمان پارامترهای ε، ρ و θ.

نام مدل:	Q_LRP	Q_FA	Q_L	Q_G	Q_K1	Q_K2
مقدار θ:	۰	۰	۱۰	۱۰	۰	۱۰

جدول ۳. مقدار نهایی θ پس از شبیه‌سازی نمودار ۸.



نمودار ۹. مقایسهٔ عملکرد مدلهای مختلف با تنظیم همزمان پارامترهای ε، ρ و σ.



نمودار ۱۰. مقایسهٔ عملکرد مدلهای مختلف با تنظیم همزمان پارامترهای θ، ε، ρ و σ.

در جدول ۴ تعداد دسته‌ها در انتهای شبیه‌سازیهای نمودارهای ۸ تا ۱۰ آورده شده است. می‌توان نشان داد که حداقل تعداد دسته‌هایی که مأمور در محیط شکل ۴ احتیاج دارد ۱۲ است (این دسته‌ها معنی‌دارترین دسته‌ها برای یادگیری این محیط را نیز تشکیل می‌دهند). اما همانطور که مشاهده می‌شود مدلهایی که پارامترهای آنها توسط اتوماتونهای ساختار ثابت تنظیم می‌شده نتوانسته‌اند به تعداد دسته‌های بهینه همگرا شوند و در این مورد تنظیم پارامترها با اتوماتونهای ساختار متغییر نتایج بهتری را در برداشته است.

Q_K2	Q_K1	Q_G	Q_L	Q_FA	Q_LRP	Q	
۳۲	۱۰	۳۲	۱۲	۳۲	۱۷	۲۴	ε, ρ, θ
۲۳	۱۶	۲۰	۲۱	۱۲	۱۲	۲۴	ε, ρ, σ
۱۵	۱۱	۴۶	۲۹	۳۲	۱۲	۲۴	ε, ρ, σ ,θ

جدول ۴. تعداد دسته‌ها پس از شبیه‌سازی

۶.جمعبندی

با تنظیم خودکار پارامترهای مدل یادگیری Q مشکلات استفاده از روشهای تجربی و سعی و خطا در یافتن مقادیر بهینهٔ این پارامترها از بین می‌رود. پیش از این در [2] و [3] استفاده از اتوماتونهای یادگیر با ساختار متغییر برای تنظیم خودکار پارامترها پیشنهاد گشته است و در این مقاله نیز نتایج استفاده از اتوماتونهایی با ساختار ثابت برای این منظور مورد بررسی قرار گرفته شد. نتایج آزمایشها نشان می‌دهند که اتوماتونهای ساختار ثابت تقریباً در تمام آزمایشها نتایج بهتری نسبت به مدل کلاسیک (بدون تنظیم پارامترها) داشته‌اند. در بسیاری از آزمایشها نتایج این مدلها از مدلهای تنظیم شده با اتوماتونهای ساختار متغییر نیز بهتر بوده است. هرچند این اتوماتونها بیش از اتوماتونهای ساختار متغییر در بهینه‌های محلی قرار می‌گیرند اما در عوض درای ساختار نسبتاً ساده‌تری می‌باشند و بنظر می‌رسد که برای تنظیم پارامترهای مدل Q در بسیاری از کاربردها مناسب باشند.

مراجع

[1] C. Watkins, Learning from delayed rewards, PhD. Thesis, Kings College, 1989.

[2] S. Hodjat and M.R. Meybodi, Fine tuning of Q-learning parameters using learning automata, The Second Annual Conference of Computer Society of Iran, pp. 33-44, 1996.

[3] S. Hodjat and M.R. Meybodi, Fine tuning of Q-learning parameters using games of automata, Computer Science Technical Report, Amirkabir University, 1997.

[4] L. P. Kaelbling and M. L. Littman and A.W. Moore, Reinforcement learning, Artificial Intelligence Journal, vol 4, pp 237-285 , 1996.

[5] R. Schalkoff, Pattern recognition, Wiley International Editions, 1991.

[6] S. Mahadevan and J. Connel, Automatic programming of behavior-based robots using reinforcement learning, Artificial Intelligence Journal, vol 55, pp 311-365, 1992.

[7] M. Dorigo and H. Bersini, A comparision of Q learning & classifier systems, Proceedings of from Animats to Animals, International Conference on Simulation of Adaptive Behavior, SAB 1994.

[8] K. S. Narendra and A. L. Thathachar, Learning automata, Prentice Hall, 1989.

[9] M. R. Meybodi and S. Lakshmivarahan, ε-Optimality of a general class of absorbing barrier learning algorithms", Information Sciences, vol 28, pp. 1-20, 1982.

[10] M. L. Tsetlin, On the behavior of finite automata in random media, Automata and Remote Control, vol 22, pp 1345-54, 1962.

[11] M. L. Tsetltn, Automata theory and modeling of biological systems, Academic Press, NY, 1973.

[12] V. I. Krinsky, An Asymptotically optimal automaton with exponential convergance, Biofizica, vol 9 , pp 99-105, 1964.

[13] V. Yu. Krylov, One stochastic automaton which is asymptotically optimal in random medium, Automata and Remote Control, vol 24 , pp 1114-16, 1964.

[14] M. C. Mozar and J. R. Bachrach, Discovering the structure of a reactive environment by exploration, Technical Report. CU-CS-451-8. Dept of Computer Science, University of Colorado, Boulder, November 1989.

[15] A. G. Barto, S. J. Bradthe and S. P. Singh, Learning to act using real-time dynamic programming, Artificial Intelligence, vol 72(1), pp. 81-138, 1995.

[16] A. K. McCallum, Efficient Exploration in reinforcement learning with hidden state, University of Rochester, 1996.

[17] S. D. Whitehead, Complexity and cooperation in Q learning, In L.A. Birnbauman G.C. Collins, editors, proceedings of the eighth international workshop on Machine learning, San Mateo, CA, Morgan Kaufmann , pp 363-367, 1991.

[18] L. Kaelbling, Learning in embedded systems, PhD. Thesis, Stanford University, Stanford. CA, 1990.

[19] S. Koenig and R. G. Simmons. Complexity analysis of real time reinforcement learning, In Proceedings of Eleventh Conference on Artificial Intelligence, AAAI-93, Menlo-park CA, pp 99-105, 1993.

[20] M. Sato, K. Abe and H. Taheda, Learning control of finite Markov chains with an explicit trade off between estimation and control, IEEE Transactions on Systems, Man and Cybernetics, vol 18(5), September 1988.

[21] S. B. Thrun, The role of exploration in learning control, In handbook of intelligent control: Neuro, Fuzzy and Adaptive Approaches, Nostrand Reinhold, 1992.

[22] S. Hodjat and M.R. Meybodi, Fine tuning of Q-learning parameters using fixed structured automata, Computer Science Technical Report, Amirkabir University, 1997.

PROCEEDING OF THE

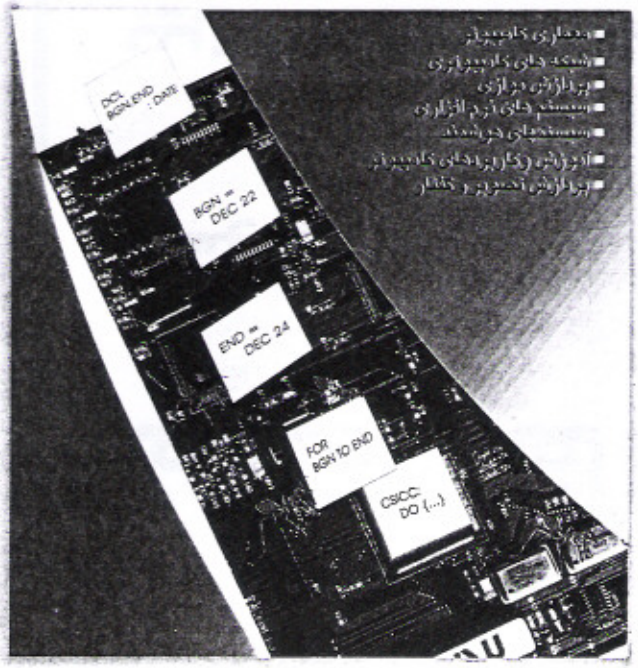
THIRD INTERNATIONAL

ANNUAL COMPUTER

SOCIETY OF IRAN

COMPUTER CONFERENCE

CSICC'97



COMPUTER CONFERENCE

Iran University Of Science And Technology

Computer Engineering Department

I.R.IRAN - I.R.IRAN

23-25 Dec. 1997