# استفاده از مفهوم استیگمرژی دراتوماتاهای یادگیر برای بازی های تصادفی

بهروز معصومی
دانشکده مهندسی برق، رایانه دانشگاه آزاد اسلامی قزوین
دانشگاه آزاد اسلامی علوم و تحقیقات تهران
bmasoumi@Qazviniau.ac.ir

محمد رضا میبدی
دانشکده مهندسی کامپیوتر و فناوری اطلاعات، دانشگاه صنعتی امیرکبیر،تهران، ایران
mmeybodi@aut.ac.ir

**چکیده- استیگمرژی به معنای نوعی همکاری غیرمستقیم بین فعالیت هاست که برای تشریح تاثیر رفتاری که عوامل محیط بر آن به جای می گذارند استفاده می شود. در حال حاضر این مفهوم در سیستمهای چند عامله به کاررفته و چارچوبی رابرای تعامل عامل ها و هماهنگی بین آنها فراهم ساخته است. در این مقاله مکانیزم استیگمرژی در اتوماتاهای یادگیر مورد بهره برداری قرار گرفته است. برای این منظور مدلی مبتنی بر اتوماتاهای یادگیر بر اساس ایده استیگمرژی در حل بازی های تصادفی ارائه گردیده است. در این مدل فرض می شود یکسری عامل ها مجازی در محیط وجود دارند که با توجه به اعمالشان اثری در محیط بر جای می گذارند که این اثر در اتوماتای یادگیر به جای مانده و باعث افزایش وتسریع یادگیری اتوماتای یادگیر می گردد. برای بررسی مدل مذکور از بازی های *GridGame* استفاده شده است. نتایج آزمایشهای انجام گرفته نشان از افزایش سرعت یادگیری را دارد.**

**کلید واژه- اتوماتای یادگیر، استیگمرژی، بازی های تصادفی، سیستمهای چند عامله.**

## 1- مقدمه

مفهوم استیگمرجی اولین بار توسط حشره شناسی به نام پیرپول‌گراسی برای تحلیل رفتار موریانه ها مطرح شد[1]. گراسی به دنبال مکانیزمی می گشت که متضمن ظهور ، هماهنگی و کنترل فعالیتهای جمعی و اجتماعی حشرات باشد. بطور کلی اسیگمرجی بصورت رده ای از سازوکار ها است که به طور غیرمستقیم میانجی ارتباطات بین حیوانات است. ایده اصلی استیگمرجی بر این است که چگونه عاملهای مستقل به طور غیر مستقیم با تاثیر بر محیط به رفتار هماهنگ می رسند[2]. این مفهوم برای توضیح چگونگی رسیدن به رفتار هماهنگ در حشراتی که رفتار جمعی دارند نظیر موریانه ها ،مورچگان و زنبورها مورد استفاده قرار گرفت.

در حال حاضر موضوع استیگمرجی در حوزه های سیستمهای چند عامله و محاسبات مبتنی بر عاملها نیز بسیار مورد توجه قرار گرفته است[3,4,5]. ازجمله سیستم‌هایی که بر اساس این روش عمل پیاده‌سازی شده‌اند، روشهای مبتنی بر کلونی مورچه[1]ها را می‌توان نام برد که در آن جنبه های رفتار اجتماعی حشرات را برای ایجادرفتار هماهنگ و همکاری بین آنها مدل می کند. در این مقاله مدلی مبتنی بر مفهوم استیگمرجی برای ارتباط بین عاملها با استفاده از اتوماتاهای یادگیر برای حل بازی های مارکوفی پیشنهاد می گردد. بازی های مارکوفی یکی از مدل های سیستمهای چند عامله، مبتنی بر مدل مارکوف هستند[6]. این بازی ها توسعه ای از فرآیندهای تصادفی مارکوف با چندین عامل بوده و به عنوان چارچوبی مناسب در تحقیقات یادگیری های چند عامله‌به‌ویژه یادگیری‌تقویتی چندعامله[2] به‌کاررفته‌اند [8] [7]. اتوماتاهای یادگیرنیز در حال حاضر به عنوان ابزاری ارزشمند در طراحی الگوریتمهای یادگیری تقویتی بوده به واسطه ویژگیهایی که دارند در بسیاری از کاربردهای جند عامله و محیط های ناشناخته مناسب هستند [9] [10]. برای حل بازی های مارکوفی نیز الگوریتمهای مختلفی مبتنی بر اتوماتاهای یادگیر ارائه شده است. در[11] روشی مبتنی براتوماتاهای یادگیر برای حل فرآیندهای تصادفی مارکوف چند عامله وبازی های مارکوفی در شرایط ارگودیک[3] مطرح شده اند و نشان داده شده است که شبکه ای از اتوماتاهای یادگیر[4] قادر به رسیدن به استراتژی های تعادل در بازی های مارکوفی می باشند. در[12] یک راه حل کلی برای بازی های

مارکوفی با مجموع کلی با استفاده از اتوماتای یادگیر ارائه شده است که در آن با توجه به بردارهای احتمالات اعمال اتوماتای یادگیر و محاسبه آنتروپی اطلاعاتی به دست آمده در هر حالت، اعمال انتخابی اتوماتای یادگیر پاداش یا جریمه دریافت می دارند. این روش تحت عنوان MLA خوانده می شود. ابطحی و همکاران الگوریتمی دیگر را مطرح می کند که در آن با استفاده از پیداکردن هزینه کوتاهترین مسیر و پاداش دادن به اعمال اتوماتای مسیر استفاده شده است[13].

در[14] روشی با استفاده از اتوماتای یادگیر برای کنترل فرآیندهای تصادفی چند عامله با توجه به مدل کلونی مورچگان ارائه شده است که در آن مشابهت استفاده از اتوماتاهای یادگیر وکولونی مورچه ها نشان داده شده است و نشان داده شده که شبکه اتوماتاهای یادگیر مستقل قادر به کنترل فرآیندهای تصادفی چند عامله با پاداش و احتمالات ناشناخته هستند.

در این مقاله، مفهوم استیگمرجی چارچوبی ساده برای ارتباطات و هماهنگی عاملها را بیان می کند. برای این منظور از مفهومی به نام آنتروپی بردار احتمالات اتوماتای یادگیر در هر حالت به عنوان نقشی مشابه فرمون هااستفاده شده است که در واقع، مدل ارتباطی بین عاملها مبتنی بر استیگمرجی با استفاده از اتوماتهای یادگیر برای حل بازی های مارکوفی است. برای بررسی و ارزیابی روش پیشنهادی از محیط بازی های Grid Game برای شبیه سازی آزمایش ها استفاده شده است. درادامه مقاله، در بخش 2 مفهوم استیگمرجی توضیح داده شده است. در بخش 3بازی های مارکوفی ارائه می گردند. در بخش 4 اتوماتاهای یادگیر و در بخش 5 الگوریتم پیشنهادی مطرح و در بخش 6 آزمایش های انجام شده در محیط بازی های GridGame و نتایج آزمایشها ارائه گردیده اند.

## 2- الگوریتمهای استیگمرجی

استیگمرجی فرآیندی خود کار از هماهنگی غیر مستقیم بین واحد ها با فعالیتهاست که اثر آن برروی محیط با استفاده از رفتاری است که کارایی عمل بعدی آن فعالیت یا واحد های دیگر را تحت تاثیر قرار می دهد. این مفهوم یک فرم خود سازمانده است و ساختارهای پیچیده و هوشمندی را تولید میکند که به هیچ نقشه، کنترل و یا ارتباطی بین واحد ها نیاز ندارد. تا به حال الگوریتمهای مختلفی برای به کارگیری استیگمرجی در سیستمهای چند عامله ارائه شده است. یکی از روشهای معمول ایجاد ارتباط بین عامل ها از طریق فرمون است*[15]* . عاملها می توانند فرمون ها را در محیط حس کرده و برای راهنمایی انتخاب عمل خود تغییر دهند. الگوریتمهای دیگر موجود بر پایه لانه سازی موریانه ها و زنبورهای وحشی و یا تفکیک نوزادان مورچه است *[16][17]*. در این سیستمها عملکرد یکی از افراد برروی محیط محلی تاثیر گذاشته و باعث تغییر در رفتار دیگران می گردد. در اکثر سیستمهای نام برده شده مجموعه ای از عناصر مجزا از هم وجود دارند. محیط عامل به چند منطقه محلی تقسیم می گرددو هر محل دارای یک حالت محلی است که عامله می توانند به آن دسترسی داشته و آن را تغییر دهند. عاملها می توانند در محلی که سرکشی می نمایند با توجه به حالت محلی عملی را انجام دهند. بین محل ها یک نقشه ارتباط داخلی تعریف شده که امکان حرکت بین آنها را به عاملها می دهد. هدف ما گنجاندن این مطالب در چارچوب اتوماتاهای یادگیر است.

### 2-1- یک الگوریتم ساده استیگمرجی

یکی از ساده ترین الکوریتمهای استیگمرجی روش بهینه سازی مبتنی بر کلونی مورچگان است که در آن تمامی عامل ها دارای هدف مشترک بوده و محیط را برای برقراری ارتباط و هماهنگی کارها تغییر می دهند*[15]*. راه حل مورچه ها برای پیدا کردن کوتاهترین مسیر بین لانه و منبع غذا به صورت زیر است: مورچه‌ها هنگام راه رفتن از خود ردی از فرومون به جا می‌گذارند. فرومون با گذشت زمان تبخیر می‌شود اما در کوتاه‌مدت به عنوان رد مورچه بر سطح زمین باقی می‌ماند. یک رفتار بسیار ساده پایه‌ای در مورچه‌ها وجود دارد: آنها یک مسیر را از بین چندین مسیر بر اساس احتمال انتخاب می کنند. مسیری که فرومون بیشتری داشته باشد یا به عبارت دیگر مورچه‌های بیشتری قبلا از آن عبور کرده‌باشند احتمال بیشتری برای انتخاب شدن خواهد داشت. مورچه‌ها برای انتخاب گرهٔ بعدی از مقادیر فرمون‌ها و فاصلهٔ بین گره‌ها استفاده می‌کنند. بعد از اینکه هر مورچه یک دور تولید کرد مقدار دنبالهٔ فرمون روی یال‌ها بهنگام می‌شود. در کلونی مورچه‌ها این عمل ابتدا با کاهش مقدار فرمون با

یک فاکتور ثابت (ضریب تبخیر) و سپس قرار دادن فرمون توسط هر مورچه روی یال‌هایی که در دور خود پیموده است انجام می‌شود. ضریب تبخیر که مقداری بین صفر و یک دارد برای اجتناب از انباشتن بی‌حد دنبالهٔ فرومون است و الگوریتم را قادر می‌سازد تا تصمیم‌های بدی را که قبلاً گرفته شده است فراموش کند. برای اطلاعات بیشتر در مورد الگوریتم های کولونی مورچه ها می توان به *[5,6]* مراجعه نمود.

## 3- اتوماتاهای یادگیر

اتوماتای یادگیر، ماشینی است که می‌تواند تعدادی متناهی عمل را انجام دهد. هر عمل انتخاب شده توسط یک محیط احتمالی ارزیابی می‌شود و نتیجه ارزیابی در قالب سیگنالی مثبت یا منفی به اتوماتا داده می‌شود و اتوماتا از این پاسخ در انتخاب عمل بعدی تأثیر می‌گیرد. هدف نهایی این است که اتوماتا یاد بگیرد تا از بین اعمال خود، بهترین عمل را انتخاب کند. بهترین عمل، عملی است که احتمال دریافت پاداش از محیط را به حداکثر برساند. کارکرد اتوماتای یادگیر در تعامل با محیط، در شکل1 مشاهده می‌شود.

محیط را می‌توان توسط سه‌تایی $E \equiv \{\alpha, \beta, c\}$ نشان داد که در آن $\alpha \equiv \{\alpha_1, \alpha_2, ..., \alpha_r\}$ مجموعه ورودی‌ها، $\beta \equiv \{\beta_1, \beta_2, ..., \beta_m\}$ مجموعه خروجی‌ها و $c \equiv \{c_1, c_2, ..., c_r\}$ مجموعه احتمال‌های جریمه می‌باشد. هرگاه $\beta$ مجموعه‌ای دو عضوی باشد، محیط از نوع P است. در چنین محیطی $\beta_1 = 1$ به عنوان جریمه و $\beta_2 = 0$ به عنوان پاداش در نظر گرفته می‌شود. در محیط از نوع Q، $\beta(n)$ می‌تواند به طور گسسته یک مقدار از مقادیر محدود در فاصله $[0.1]$ را اختیار کند و در محیط از نوع S، $\beta(n)$ متغیر تصادفی در فاصله $[0.1]$ است. $c_i$ احتمال اینکه عمل $\alpha_i$ نتیجه نامطلوب داشته باشد می‌باشد. در محیط ایستا، مقادیر $c_i$ بدون تغییر می‌مانند، حال آن‌که در محیط غیرایستا این مقادیر در طی زمان تغییر می‌کنند. اتوماتاهای یادگیر به دو دسته اتوماتای یادگیر با ساختار ثابت اتوماتای یادگیر با ساختار متغیر[5] (VSLA) دسته بندی می شوند.

اتوماتای یادگیر با ساختارمتغیر را می‌توان توسط چهارتایی $\{\alpha, \beta, p, T\}$ نشان‌دادکه $\alpha \equiv \{\alpha_1, \alpha_2, ..., \alpha_r\}$ مجموعه عمل‌های اتوماتا، $\beta \equiv \{\beta_1, \beta_2, ..., \beta_r\}$ ورودی‌های اتوماتا، $p = \{p_1, .., p_r\}$ بردار احتمال انتخاب هریک از عمل‌ها و $p(n+1) = T[\alpha(n), \beta(n), p(n)]$ الگوریتم یادگیری می‌باشد.

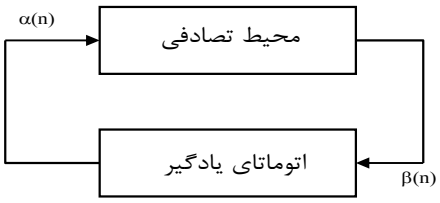
الگوریتم‌زیربراساس‌روابط (2) و (3)یک‌نمونه‌ازالگوریتم‌های یادگیری خطی است. فرض می‌کنیم‌عمل $\alpha_i$در مرحله nام انتخاب شود. در اینصورت پاسخ مطلوب از محیط بصورت:

$$p_i(n+1) = p_i(n) + a[1 - p_i(n)]$$
$$p_j(n+1) = (1-a)p_j(n) \qquad \forall j \quad j \neq i \qquad (2)$$

و پاسخ نامطلوب از محیط بصورت زیر می باشد.

$$p_i(n+1) = (1-b)p_i(n)$$
$$p_j(n+1) = (b/r - 1) + (1-b)p_j(n)$$
$$\forall j \quad j \neq i \qquad (3)$$

در روابط (2) و (3)، a پارامتر پاداش وb پارامتر جریمه می‌باشند. با توجه به مقادیرa وb سه حالت را می‌توان درنظر گرفت: اگرa وb باهم برابر باشند، الگوریتم را $L_{RP}$، هنگامی‌کهb ازa خیلی کوچکتر باشد، الگوریتم را $L_{R\varepsilon P}$ و اگر b مساوی صفر باشد آن را $L_{RI}$ می‌نامیم[18]. شمای S-$L_{RP}$ برای مدلهای Q و S براساس رابطه (4) بیان‌می شود:




شکل1- ارتباط بین اتوماتای یادگیر و محیط

اگر عمل $\alpha_i$ در مرحله $n$ام انتخاب شود در این صورت طبق معادله (4) داریم :

$$p_i(n+1) = p_i(n) + a(1-\beta_i(n))(1-p_i(n)) - a\beta_i(n)p_i(n) \qquad (4)$$
$$pj(n+1) = pj(n) + a(1-\beta_i(n))(pj(n)) + a\beta_i(n)\left[\frac{1}{r-1} - p_j(n)\right] - a.(1-\beta_i(n))p_j(n) \; if \; j <> i$$

r تعداد اعمال ممکن، aپارامتر پاداش وb پارامتر جریمه می‌باشند. برای اطلاعات بیشتر در باره اتوماتاهای یادگیر می توان به[19] مراجعه نمود.

## 3-1- بازی اتوماتاهای یادگیر

در بازی اتوماتاها **Error! Reference source not found.**[18] و[20] تعدادی اتوماتا بدون آنکه دانش کاملی از یکدیگر داشته باشند در محیط عمل می کنند.از نقطه نظر بازی ها بهتر است اعمال اتوماتا را به عنوان استراتژی آنها و ورودی شان را به عنوان نتایج در نظر بگیریم . یک بازی در برگیرنده بازی های تکراری درهر بازی هر اتوماتا یک عمل را بر پایه توزیع احتمال جاری روی مجموعه اعمال انتخاب می کند. با انجام نتایج هر اتوماتا یک ورودی را از محیط دریافت می کند.

به دلیل طبیعت تصادفی بودن محیط ها ،انتخاب اعمال توسط هر اتوماتا در یک بازی تعیین کننده احتمال نتایجشان است. همچنین فرض شده اتوماتای شرکت کننده در بازی، احتمالات نتایج رابه عنوان دانش اولیه دراختیار ندارد. علاوه براین ،فقط اطلاعاتی که هر اتوماتا در جریان بازی از ورودی اش بدست می‌آورد به عنوان تابعی از اعمالش در بازی های موفق محسوب می شود. بنابر این در قبال تئوری بازی ها ، در بازی های اتوماتا بازیکنان اطلاعات اولیه در ارتباط با بازی نظیر تعداد بازیکنان ، اعمال احتمالی ایشان و عنصر ماتریس های بازی را در اختیار ندارند.

در بازی های با مجموع صفر الگوریتم LR-I به نقطه تعادل همگرا می شود اگر در استراتژیهای محض باشد در حالیکه در الگوریتم LR-εP به استراتژی های تعادل مخلوط همگرایی دارد.در **Error! Reference source not found.**نشان داده شده است در بازی های با مجموع غیر صفر وقتیکه اتوماتاهای یادگیر از الگوریتم LR-I استفاده می نمایند و بازی دارای یک نقطه تعادل محض باشد همگرایی تضمین شده است.

## 4- معرفی بازی های مارکوفی

### 4-1- تعریف فرآیندهای تصادفی مارکوف[۶]

مساله کنترل کردن یک زنجیره مارکوفی محدود به نام مساله تصمیم گیری مارکوفی خوانده می شودکه در آن احتمالات گذارحالت وپاداش‌ها ناشناخته اندو به صورت زیر تعریف می شود.

*تعریف1.* فرآیند تصادفی مارکوف بصورت چندتایی $\langle S,A,R,T\rangle$ نشان داده می شود که در آن $S$ مجموعه متناهی از وضعیت‌ها ؛ $A$ مجموعه عملیات قابل دسترس برای عامل ،$\gamma$ ضریب کاهش و $T : S\times A\times S \to [0,1]$ احتمال انتقال از وضعیت جاری به وضعیت بعدی با انجام عمل $a$ است و $R: S\times A\to \Re$ تابع پاداش است که یک مقدارحقیقی را بر می گرداند. هدف کلی در فرآیند های تصادفی مارکوف، پیدا کردن سیاستی مانند $\alpha$ است بطوریکه امید ریاضی مجموع کاهش‌یافته پاداشها $J(\alpha)$ را بیشینه نماید که در رابطه (5) دیده می شود. سیاستهای در نظر گرفته شده بصورت ایستا بوده و غیر تصادفی هستند.

$$J(\alpha)=\lim_{l\to\infty}\frac{1}{l}E\left[\sum_{t=0}^{l-1}R^{x(t)x(t+1)}(\alpha)\right] \qquad (5)$$

### 4-2- تعریف بازی های مارکوفی

بازی های مارکوف تعمیم فرآیندهای تصادفی مارکوف به‌حالت چندعامله است و بصورت زیر تعریف میشوند :

تعریف2. بازی‌مارکوف بصورت‌چندتایی $\langle N,S,A_{1..n},T,R_{1..n}\rangle$ بیان میشود که‌درآن$N$ تعداد عامل ها، S مجموعه حالات، $Ai$ مجموعه اعمال هر عامل $i$ (در فضای اعمال گروهی $(A_1 \times A_2 \times \ldots \times A_n$ ، $T$ تابع انتقال $S\times A\times S \to [0,1]$ و $R$ تابع پاداش برای عامل $i$ام با توجه به اعمال انتخابی در هر حالت است.

هر بازی مارکوفی با یک حالت بصورت یک بازی نرمال تکراری در تئوری بازی ها شناخته شده و هربازی مارکوفی با یک عامل بصورت یک فرآیند تصمیم گیری مارکوفی است. علاوه بر این هر عامل تابع پاداش خاص خودش را داراست. در حالتی که هر عامل پاداش مختلفی را داراست پیدا کردن سیاست بهینه برای تمام عاملها بسیار مشکل بوده لذا بجا ی آن نقاط تعادل بازی جستجو می شوند، وضعیتی که هیچ عاملی به تنهایی نمی تواند تا زمانیکه تمام عاملهای دیگر سیاستشان را ثابت نگه می دارند برای بهبود پاداش سیاستش را تغییر دهد.

## 5- روش پیشنهادی

در این بخش روش پیشنهادی برای حل بازی های مارکوف کلی با توجه به ایده استیگمرژی به کمک

اتوماتای یادگیر ارائه میگردد. در این روش، برای ارتباطات استیگمرژی از اتوماتای یادگیر استفاده می شود. فرض می شود در هر حالت از محیط بازی $S_i$ (i=1..m) و m تعداد حالات) بازای هر عامل k در بازی یک اتوماتای یادگیر نظیر $LA_k^i$ با ساختار متغیر قرار داده می‌شود. با توجه به تعداد حالت های مجاور با هر حالت از محیط، تعداد اعمال اتوماتای یادگیرمشخص می‌گردند. هر عامل با توجه به عمل انتخاب شده توسط اتوماتاهای یادگیر آن حالت و ترکیب گروهی اعمال آنها به حالت بعدی می رود.

در این روش، علاوه بر عاملهای درگیر در بازی که از یک حالت شروع، آغازنموده و به حالت نهایی می رسند، یکسری عامهای مجازی نیز برای هر حالت از محیط در نظر گرفته شده اند. در هر لحظه هر یک از عوامل با توجه به اتوماتاهای در آن حالت یادگیر تغییر حالت داشته و اتوماتاهای یادگیر حالت قبلی با توجه به آنتروپی بردار احتمالات اعمال‌اتوماتای‌یادگیر حالت جدید و یا به هدف رسیدن(حالت نهایی) پاداش یا جریمه دریافت می دارند. الگوریتم موقعی پایان می یابد که عاملهای واقعی به هدف (حالت نهایی) مورد نظر برسند. آنتروپی بردار احتمالات اعمال اتوماتاها در اینجا همان نقش فرمونها را ایفا می کنند. و عاملهای مجازی نیز نقش مورچه ها را ایفا می نمایند.

در ابتدا فرض می شود اتوماتاهای یادگیر تمام عملهای خود را با احتمالی یکسان انتخاب می‌کنند. در صورتیکه تغییر حالت ناشی از اعمال اتوماتای هر حالت منجر به ورود عامل به حالت نهایی (هدف) شود اتوماتای یادگیر پاداش می گیرد و در صورتیکه حالت جدید حالت نهایی(هدف) نیست هر اتوماتا با توجه به بردارهای احتمال اعمال اتوماتای یادگیر و محاسبه آنتروپی اطلاعاتی بردارهای احتمالات بصورت زیر تعریف می شود پاداش یا جریمه می گیرند.

$$H(S_i) = -\sum_{j=1}^{N} P_j(S_i)\log(P_j(S_i)) \qquad (6)$$

$N$ تعداد اعمال اتوماتا در حالت $S_i$ و $P_j(s_i)$ احتمال انجام عمل $j$ در حالت $S_i$ است. آنتروپی بردار احتمال، میزان عدم قطعیت اتوماتای یادگیر حالت بعد را در انتخاب عمل خود نشان می‌دهد. هر چه آنتروپی بیشتر باشد میزان عدم قطعیت بیشتر است. در ابتدای الگوریتم که بردارهای احتمال اعمال اتوماتای یادگیر دارای مقادیر یکسان هستند یعنی $P_1=P_2=\ldots P_r=1/r$ میزان آنتروپی بیشترین مقدار است. لذا اتوماتای یادگیر دارای اطلاعات مفیدی برای رسیدن به هدف نبوده و عملهای خود را به صورت تصادفی انتخاب می‌کند (جستجو[۷]). با ادامه الگوریتم با تغیر احتمالات اعمال اتوماتای یادگیر میزان آنتروپی کم می شود و در حالت مینیم به صفر می رسد. کاهش آنتروپی به این معنی است که اتوماتای یادگیر با احتمال بالایی یکی از اعمال خود را انتخاب می‌کند و دارای اطلاعات مفیدی برای رسیدن به هدف بوده و از این اطلاعات بهره برداری[۸]می نماید. برای اینکه مقدار آنتروپی به مقداری بین 0 و 1 تبدیل شده تا به عنوان پاداش یعنی$\beta$در اتوماتای یادگیر در محیطهای مدل$S$ قابل استفاده باشد، آنتروپی آن حالت را به آنتروپی ماکزیمم تقسیم کرده و به آن آنتروپی نسبی گویند. الگوریتم نهایی در شکل 2 نشان داده شده است.

## 6- آزمایشهای انجام گرفته در محیط بازی GridGame

یکی از انواع بازی‌های‌اتفاقی‌غیر رقابتی بازی های Grid Game است که توسط Hu,Wellman ارائه شده است[**Error! Reference source not found.**]. این بازی یک بازی مارکوف دو نفری از نوع جمع کلی است. درشکل3 بازی GridGame نشان داده شده است. در این بازی دوعامل وجود دارندکه هردو می خواهند به یک هدف مشترک برسند. .فرض بر این است که دوعامل از دو گوشه یک صفحه شروع کرده و سعی دارندتا باکمترین تعداد حرکت به هدف برسند.

اعمال بازیکنان بطور همزمان انجام گرفته و هر یک از بازیکنان می توانند یکی از اعمال شمال، جنوب، شرق و یا غرب را انتخاب نمایند. مجموعه فضای‌حالات بصورت{(0 1), (0 2), … , (8 7)}=S تعریف می شوند که هرحالت $S=(l_1,l_2)$ مختصات عاملهای 1و2 را نشان می دهد. عاملها همزمان نمی توانند در یک مختصات یکسان قرار گیرند. اگر دو عامل سعی در حرکت به یک مربع یکسان داشته باشند حرکت هردو باشکست مواجه شده و هردو یک واحد جریمه گردیده و در موقعیت قبلی باقی می مانند. اگر عاملها به دو مربع مختلف غیر هدف بروند

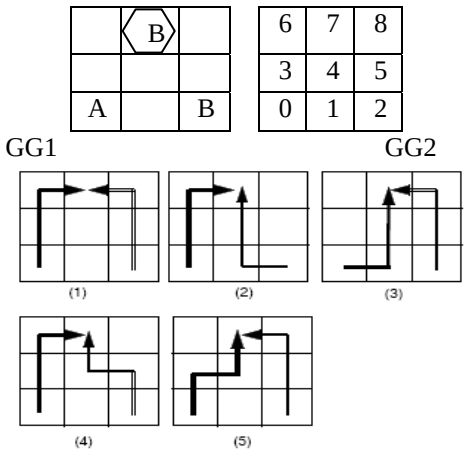
هر دو پاداش صفر را دریافت می کنند و اگر یکی به هدف برسد 100 واحد پاداش می گیرد.

یک مسیر دنباله ای از اعمال انتخاب شده از نقطه شروع تا پایان را نشان می دهد. به کوتاهترین مسیری که یک عامل بدون تداخل با عاملهای دیگرطی می نماید مسیر بهینه گفته می شود.

The Proposed algorithm :

**StigmergyMLA(SG,a,b,M)**
**Inputs: a ,b**: reward and penalty parameter for each LA **k**: exploration Parameter , **M** : total training time
**Initialize** :
for all states s, agents k do
P(s,k) =1/number of permissible actions
end for
// Main Loop
for episode =1 to M
s = startState; //random or fixed
while not done
**For** each ps in states **do in parallel**
jointAction =∅;
for all real agent k or virutal agent do
cuncurrently
Activate $LA_k^s$
Action = SelectAction($LA_k^{ps}$)
Observe Rewards $r_k^{ps}$
jointAction = jointAction ∪ action;
ps' = getNextState(s, jointAction)
Compute $\beta_k^{ps'}$signal base on EQ (6)
Train $LA_k^s$ according $\beta_k^{s'}$
end state
end while
end for episode

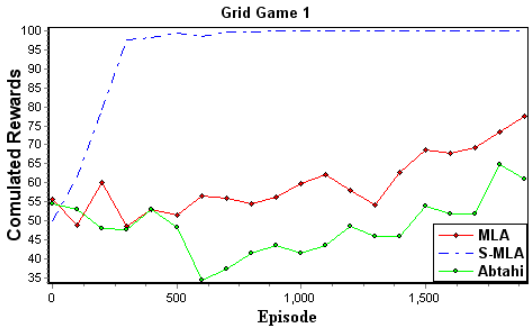
شکل 2. الگوریتم پیشنهادی




شکل 3. بازی Grid World و نمایش مختصات بازی به همراه راه حلهای بهینه آن

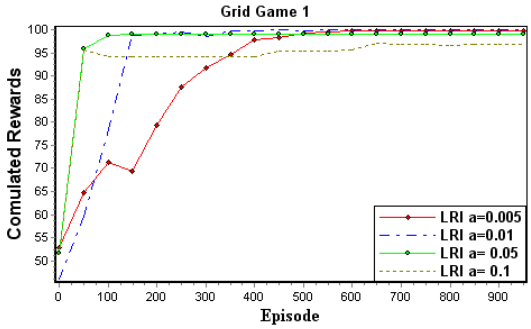
در این بازی فرض می شود عاملها از موقعیت هدف در ابتدای بازی آگاهی نداشته و همچنین از پاداش سراسری یکدیگر اطلاع ندارند. عاملها اعمالشان را همزمان انتخاب نموده و فقط می توانندازاعمال قبلی عاملهای دیگروحالت فعلی (موقعیت مشترک هر دو عامل) آگاهی داشته باشند.

## 6-1- مقایسه الگوریتم پیشنهادی با دیگر روشها

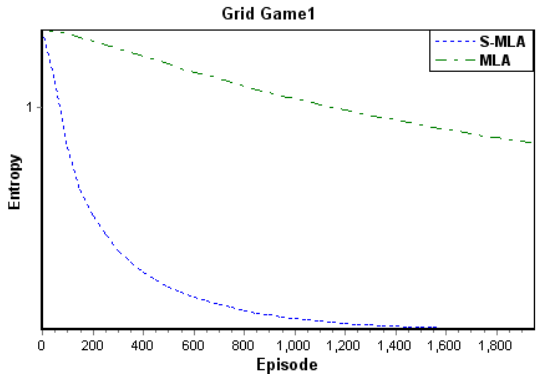



شکل مقایسه الگوریتم S-MLA و MLA در بازی GG1

A=0.005, K=3

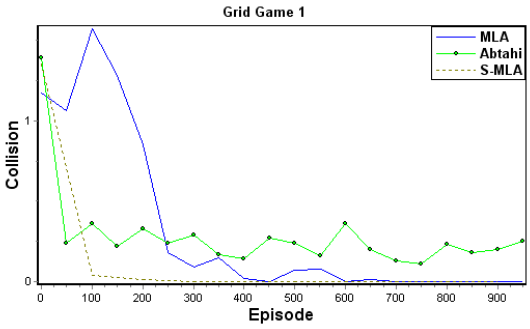



شکل نقش پارامتر یادگیری در همگرایی




## 7- نتیجه گیری

در این مقاله روشی مبتنی بر اتوماتاهای یادگیر برای حل بازی های مارکوفی ارائه گردید. در این روش با توجه به هزینه مسیر طی شده تا رسیدن به هدف، اعمال انتخابی اتوماتاهای یادگیر در طول مسیر پاداش یا جریمه می گیرند. در طول مسیرنیز با توجه به آنتروپی به دست آمده از بردارهای احتمال اتوماتای یادگیرحالت جدید به عنوان پاداش های کمکی جهت پاداش یا جریمه اتوماتا ها استفاده می شود و این کار باعث افزایش کارایی می گردد. با توجه به الگوریتم ارائه شده ونتایج بدست آمده می بینیم روش مورد نظر رفتار مناسبتری را نسبت به روشهای دیگربرای حل بازی های مارکوفی نشان می دهد. تعداد تکرارها، پارامترهای یادگیری و سرعت رسیدن به نقاط تعادل را تعیین می نمایند. تنظیم پارامترهای پاداش و جریمه اتوماتاها می تواند کارایی رسیدن به راه حل بهینه را افزایش دهد. با توجه به نتایج به دست آمده اتوماتاهای یادگیر مدل مناسب یادگیری و هماهنگی بین عاملها در سیستمهای چندعامله بوده و می تواند به عنوان راه حلی مناسب و کارا در بازی های مارکوف به کار روند.




شکل مقایسه الگوریتم S-MLA و MLA, و Abtahi از نظر برخوردها در هر اپیزود
A=0.05, K=3

## 8- مراجع

[1] P. P. Grass´e, "La reconstruction du nid et les coordinations inter-individuelles chez Bellicositermes natalensis et Cubitermes sp. la th´eoriede la stigmergie: Essai d'interpr´etation du comportement des termites constructeurs. Insectes Sociaux, 6(1):pp.41–80, 1959.

[2] G. Theraulaz and E. Bonabeau. *A brief history of stigmergy*. Artificial Life, 5(2):97–116, 1999.

[3] K. Verbeeck and A. Nowe. Colonies of Learning Automata,*Systems, Man and Cybernetics, Part B, IEEE Transactions on*, 32(6):pp.772–780, 2002.

[4] P. Valckenears and M. Kollingbaum. Multi-agent Coordination and Control using Stigmergy applied to Manufacturing Control. Mutli-agents systems and applications, pp. 317–334, 2001.

[5] L. Panait and S. Luke. A Pheromone-based utility Model for Collaborative Foraging. *Autonomous Agents and Multiagent Systems*, AAMAS 2004. Proceedings of the Third International Joint Conference on, pp. 36–43, 2004.

[6] Y. Shoham, *Multiagent Systems: Algorithmic Game Theoretic and Logical Foundations*, Cambridge University Press, 2009.

[7] M. L. Littman. "Markov Games as a Framework for Multi-agent Reinforcement Learning", *In Proceedings of the 11th International Conference on Machine Learning*, pp. 322 – 328, 1994.

[8] L. Busniu, R. Babuska, B. Schutter "A Coprehensive Survey of Multiagent Reinforcement Learning ", *IEEE Transaction on System, Man, Cybern*, vol. 38, no.2, pp. 156–171, 2008.

[9] M. R. Khojasteh, M. R. Meybodi, "Evaluating Learning Automata as a Model for Cooperation in Complex Multi-Agent Domains", *Lecture Notes in Artificial Intelligence, Springer Verlag, LNAI 4434*, pp. 409-416, 2007

[10] A. Now´e, K. Verbeeck, M. Peeters, "Learning Automata as a basis for Multi-agent Reinforcement Learning", *Lecture Notes in Computer Science*, vol. 3898, pp. 71–85, 2006.

[11] P. Vrancx, K. Verbeeck, A. Nowe, "Decentralized Learning in Markov Games", *IEEE Transactions on Systems, Man and Cybernetics (Part B: Cybernetics)*, vol. 38, iss. 4, pp. 976-81, 2008.

[12] B. Masoumi , M. R. Meybodi, B. Jafarpour, "Solving General Sum Stochastic Games using Learning Automata", *Proceedings of the second Joint Congress on Fuzzy and Intelligent Systems, Malek Ashtar University of Technology*, Tehran, Iran, pp. 28-30, 2008.

[13] F. Abtahi, M. R. Meybodi, "Solving Multi-Agent Markov Decision Processes Using Learning Automata", *Proceedings of the 6th International Symposium on Intelligent Systems (SISY2008)*, Subotica, Serbia, September 26-27, 2008.

[14] K. Verbeeck and A. Nowe. Colonies of learning automata. *Systems, Man and Cybernetics, Part B, IEEE Transactions on*, 32(6):pp772–780, 2002.

[15] M. Dorigo, E. Bonabeau, and G. Theraulaz. Ant algorithms and stigmergy. FUTURE GENERATION COMPUTER SYSTEM, 16(8):pp. 851–871, 2000.

[16] G. Theraulaz and E. Bonabeau. Modelling the collective building of complex architectures in social insects with lattice swarms. *Journal of Theoretical Biology*, 177(4):381–400, 1995.

[17] O. Holland and C. Melhuish. Stigmergy, self-organization, and sorting in collective robotics. *Artificial Life*, 5(2):173–202, 1999.

[18] K. Narendra, M. Thathachar. *Learning Automata: An Introduction*, Prentice-Hall International, Inc, 1989.

[19] M. A. L. Thathachar, P. Sastry, "Varieties of Learning Automata: An Overview", *IEEE Transaction on Systems, Man, and Cybernetics-Part B: Cybernetics*, Vol. 32, No. 6, pp. 711-722, 2002.

[20] M.A.L. Thathachar and P.S. Sastry. *Networks of Learning Automata: Techniques for Online Stochastic Optimization*. Kluwer Academic Publishers, 2004.

## زیر نویس ها

[1]:Ant Colony
2 Multi Agent Reinforcement Learning
3 Ergodic
4 Network of Learning Automata
5 Variable Sturcture Learning Automata
6 Markov Decision Process
7 Exploration
8 Exploitation