

# هماهنگ سازی سیستم‌های چند عامله با استفاده از اتوماتاهای یادگیر و کاربرد آن در خوشه‌بندی داده‌ها

زهره جباری<sup>۱</sup>، مهدی اثنی‌عشری<sup>۲</sup>، محمد رضا میبیدی<sup>۳</sup>

<sup>۱</sup> jabari@aut.ac.ir

<sup>۲</sup> esnaashari@aut.ac.ir

<sup>۳</sup> mmybodi@aut.ac.ir

## چکیده

سیستم‌های چند عامله امروزه به عنوان ابزارهایی مناسب برای پردازش توزیعی و محاسبات غیر متمرکز به شمار می‌آیند. در این سیستم‌ها، ایجاد هماهنگی میان عوامل مستقل به منظور رسیدن به هدف سراسری بسیار ضروری است. همچنین، با بکارگیری ابزارهای یادگیری در عوامل، رفتار سراسری سیستم چند عامله در طول زمان بهبود یافته و به سمت یافتن پاسخ بهینه سراسری همگرا می‌شود. یکی از مسائل مطرح در سیستم‌هایی با چندین عامل پراکنده در یک محیط گسترده، مسأله‌ی گردآوری عوامل همسان و در نتیجه ایجاد دسته‌هایی حاوی عوامل مشابه است. با توجه به ماهیت غیر متمرکز مسأله، یافتن روشهایی که با تکیه بر توانمندی‌های عوامل منفرد و اطلاعات محلی قادر به حل مسأله باشد بسیار اهمیت دارد. ایجاد هماهنگی در این سیستم تعامل میان عوامل جهت یافتن مکانی است که هر عامل به تعداد بیشتری از عوامل مشابه به خود نزدیک شده و در بهترین حالت منجر به خوشه بندی عوامل پراکنده در محیط شود.

در این مقاله، مسأله‌ی گردآوری عوامل همسان و پراکنده در یک محیط گسترده مورد بررسی قرار گرفته و دو راهکار جدید به این منظور پیشنهاد شده است. در راهکار نخست، با تجهیز سلولهای محیط به اتوماتای یادگیر، هدایت عوامل به مکان مناسب بر عهده‌ی اتوماتاهای یادگیر قرار گرفته است. اتوماتاهای یادگیر<sup>۱</sup> در فرایند یادگیری به سمت انتخاب عمل بهتر و در نتیجه مکان مناسب‌تر برای عوامل همگرا می‌شوند. در راهکار دوم، هر سلول به تعدادی بردار پویا مجهز شده‌است که عامل برای یافتن مکان بعدی خود از مقادیر این بردارها استفاده خواهد کرد. بردارهای احتمال هر سلول به منظور بهبود حرکت عوامل در هر گام به روز رسانی خواهند شد. نتایج بدست آمده حاکی از تاثیر قابلیت یادگیری در همگرایی نتایج و بهبود هماهنگی میان عوامل است. در نهایت با استفاده از راهکار پیشنهادی در این مقاله روش جدیدی برای خوشه‌بندی عوامل پراکنده در محیط ارائه شده است.

واژه‌های کلیدی: سیستم‌های چند عامله، هماهنگی، اتوماتای یادگیر، خوشه بندی<sup>۲</sup>

## ۱- مقدمه

رسیدن به اهداف خود و همچنین هدف کل سیستم در تلاش است. بنابراین ایجاد هماهنگی میان عوامل مستقل برای بدست آوردن یک نتیجه‌ی سراسری مطلوب ضروری خواهد بود.

در این مقاله، هدف هماهنگ‌سازی یک سیستم چند عامله است. هدف از ایجاد هماهنگی در این سیستم، آن است که عوامل مختلف پس از جستجو در محیطی سلولی، در کنار عوامل همسان با خود قرار گیرند. به این منظور دو راهکار مختلف پیشنهاد شده است که راهکار اول آن مبتنی بر اتوماتای یادگیر و راهکار دوم بر اساس محاسبه‌ی یک بردار احتمال پویا می‌باشد.

در روش اول فرض شده‌است که عوامل در یک محیط سلولی پراکنده شده‌اند. هر سلول در این فضا به چند اتوماتای یادگیر مجهز می‌باشد.

در یک سیستم چندعامله، مجموعه‌ای از عوامل برای رسیدن به اهداف خود و نیز هدف سراسری سیستم با یکدیگر همکاری می‌کنند. بطور کلی عامل یک موجودیت مستقل است که قادر به حس کردن محیط اطراف خود و انجام عملی مبتنی بر ادراک خود است. هر عامل دارای ویژگی‌هایی چون خود مختاری، استقلال، هدفمندی، همکاری، انعطاف پذیری، قابلیت تحرک و هوشمندی است [۱]. سیستم‌های چندعامله این امکان را به وجود می‌آورند که زیر مسأله‌های ایجاد شده از یک مسأله‌ی پیچیده به چندین عامل حل کننده‌ی جداگانه با اهداف و ویژگی‌های متفاوت واگذار شود. در یک سیستم چند عامله، عامل برای

اتوماتاهای یادگیر وظیفه‌ی هدایت عوامل برای یافتن جهت حرکت مناسب در محیط را برعهده دارند. به مرور زمان اتوماتاهای یادگیر بر اساس بازخوردی که عوامل پس از جابجایی خود محاسبه کرده و بازمی‌گردانند، رفتار خود را بهبود می‌بخشند. در نتیجه به مرور زمان عوامل را به مکانهایی هدایت می‌کنند که در کنار عوامل همسان با خود قرار گیرند.

در راهکار دوم، هر سلول از محیط به بردارهای احتمال پویا مجهز شده‌است و از شرایط جاری همسایگی هر عامل برای محاسبه‌ی مکان بعدی استفاده می‌شود. مقادیر بردار احتمال هر سلول بر اساس بازخوردی که پس از جابجایی توسط عوامل محاسبه می‌شود، تغییر می‌کند. با بهبود مقادیر بردار احتمال هر سلول در هر گام، به مرور زمان احتمال حرکت عوامل به مکان مناسب‌تر بیشتر می‌شود.

در نهایت، با استفاده از روش مبتنی بر اتوماتاهای یادگیر، روشی برای خوشه‌بندی عوامل پراکنده در محیط ارائه شده‌است. در این روش، با تعیین نقاطی از محیط تحت عنوان مراکز تجمع، حرکت هر عامل به سمت مرکز تجمع عوامل همسان خود افزایش می‌یابد. به عبارتی با افزایش اطلاعاتی که عوامل از محیط دریافت می‌کنند، مکان این مرکز برای عامل قابل تشخیص بوده و حرکت عامل در جهت نزدیک شدن به این مرکز بهبود می‌یابد.

ادامه این مقاله به صورت زیر سازماندهی شده است. ابتدا در بخش ۲ مروری بر فعالیتهای انجام شده در این زمینه صورت می‌پذیرد. در بخش ۳ اتوماتای یادگیر به عنوان ابزار یادگیری مورد استفاده در این مقاله معرفی می‌گردد. در بخش ۴، دو روش جدید به منظور حل مسأله‌ی هماهنگی و یافتن عوامل همسان در سیستم چند عامله ارائه خواهد شد. در نهایت در بخش ۵ به ارزیابی و مقایسه‌ی دو روش ارائه شده و نیز نتایج حاصل از استفاده از روش مبتنی بر اتوماتاهای یادگیر برای خوشه‌بندی عوامل پرداخته می‌شود. بخش ۶ حاوی نتیجه‌گیری مقاله خواهد بود.

## ۲- مرور فعالیت های گذشته

با توجه به این که مسأله‌ی مطرح در این مقاله دسته‌بندی عوامل همسان و در نهایت خوشه‌بندی عوامل در محیط است در این بخش به مرور سیستم‌های چند عامله‌ای خواهیم پرداخت که به حل مسائلی از این دست پرداخته‌اند.

در [۳] یکی از کاربردهای سیستم‌های چندعامله در زمینه‌ی داده کاوی و یادگیری مبتنی بر دانش اشتراکی برای بهبود طبقه‌بندی<sup>۳</sup> اطلاعات ارائه شده است. در روش ارائه شده، از ترکیب واسطه‌ها<sup>۴</sup> در یک سیستم متشکل از عوامل خودمختار برای ارتقاء داده‌کاوی استفاده شده است. هر عامل طی دو مرحله یادگیری یکبار به صورت انفرادی و بار دیگر به صورت مشارکتی داده‌ها را بررسی می‌کند. هر یک از عوامل در این سیستم دارای یک واسطه متفاوت با سایر عوامل است. در مرحله‌ی نخست، هر عامل به طور جداگانه اقدام به بررسی و تحلیل

اطلاعات می‌کند. این مرحله منجر به کشف و استخراج اطلاعات توسط هر عامل به‌طور مجزا و در نتیجه بررسی اطلاعات از چند دیدگاه با واسطه‌های متنوع می‌گردد. در مرحله‌ی بعد، نتایج به دست آمده از تمام عوامل با یکدیگر ترکیب می‌شود. بنابراین در فاز **هماهنگی** عوامل، اطلاعات بدست آمده از تک تک عوامل با هم تلفیق شده و نتیجه‌ی نهایی که طبقه‌بندی اطلاعات پراکنده است حاصل می‌شود. از مزایای این روش آن است که با در نظر گرفتن چندین واسطه مختلف، داده از دیدگاه‌های متفاوتی بررسی شده و با ترکیب توانمندی‌های روشهای گوناگون توسط عوامل مستقل، سیستم قدرتمندی برای اکتشاف دانش ارائه شده‌است.

از دیگر روشهای ایجاد هماهنگی در یک سیستم چندعامله، استفاده از روشهای مبتنی رفتار حشرات اجتماعی مانند روش کلونی مورچه‌ها<sup>۵</sup> ست [۴] [۵]. این روشها که تحت عنوان روشهای استیگمرجی<sup>۶</sup> شناخته شده‌اند، برای حل مسائل توزیعی راهکارهای مناسبی ارائه می‌کنند. با استفاده از عواملی مجهز به چندین مورچه، مسأله‌ی استخراج و کشف منابع اطلاعاتی از یک محیط گسترده در [۵] ارائه شده است. هر عامل وظیفه‌ی کنترل یک منبع اطلاعاتی را برعهده دارد. عامل با ارسال مورچه‌های خود به مسیرهای موجود در محیط، علاوه بر این که سایر منابع و عوامل را از وجود خود آگاه می‌کند، می‌تواند در جستجوی منابع دیگر مورد نیاز خود باشد. زمانی که مورچه به تقاطعی در مسیر رسیده و نیاز به تصمیم‌گیری برای ادامه‌ی مسیر دارد، در صورتی که در تقاطع مورد نظر اطلاعاتی راجع به منابع درخواستی عامل خود یافت، اطلاعات آن را به عامل خود می‌رساند. در غیر این صورت با بر جای گذاشتن اطلاعات خود در تقاطع، مسیری تصادفی را برای ادامه‌ی جستجو انتخاب می‌کند. در این سیستم، علاوه بر امکان کشف تمامی منابع محیط به مرور زمان، مورچه‌ها در هر تقاطع با برجای گذاشتن اطلاعات مربوط به عامل خود، سایر عوال را از حضور خود مطلع می‌کنند.

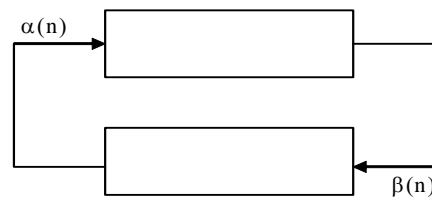
در یک بررسی دیگر که در مرجع [۶] ارائه شده‌است، روشی تحت عنوان مورچه‌های سلولی<sup>۷</sup> برای نمایش پایگاه داده‌های پیچیده و چند بعدی ارائه شده‌است. در این روش ابتدا با استفاده از تکنیک‌های استخراج ویژگی داده، سعی در استخراج ویژگیهای بصری یک داده مانند مکان، شکل، رنگ و اندازه‌ی آن شده است. سپس با استفاده از دو تکنیک ساده‌ی کلونی مورچه‌ها و اتوماتای سلولی، به معرفی یک سیستم ترکیبی جدید پرداخته شده است. مورچه‌های سلولی به نوعی یک روش دسته‌بندی داده‌ها است که داری قابلیت نمایش نتایج بدست آمده به صورت بصری نیز می‌باشد. هر داده به عنوان یک عامل در نظر گرفته شده است. تمامی عوامل از قوانین رفتاری ساده و یکسانی تبعیت کرده و با سایر عوامل در همسایگی خود در ارتباط هستند. هر عامل با بررسی محیط اطراف، در صورتی که فاصله‌ی اقلیدسی داده‌ی نرمال شده‌ی خود با داده‌ی عوامل همسایه‌اش زیاد باشد مکان خود را

ترک کرده و در غیر این صورت در مکان خود باقی می ماند. در نهایت اطلاعات همسان در کنار هم قرار گرفته و دسته بندی می شوند.

در بررسی دیگری که در مرجع [۷] انجام شده است، یک سیستم چند عامله برای خوشه بندی عوامل ارائه شده است. در این بررسی، اتوماتای سلولی تصادفی به عنوان یک سیستم چند عامله در نظر گرفته شده است. در هر گام در هر سلول از سیستم، قوانین انتخاب حالت بعدی برای اتوماتاهای آن سلول با استفاده از یک تابع توزیع احتمال از روی شرایط فعلی سلول و همسایگانش ایجاد می شود. سپس با استفاده از پارامتری تنظیم پذیر، انتخاب حالت بعدی برای هر اتوماتا به سمت ایجاد بالاترین میزان هماهنگی در سطح سراسری هدایت می شود. بالاترین میزان هماهنگی ایجاد شده در سیستم زمانی است که تعداد دسته هایی متشکل از سلولهایی با حالت یکسان، حداقل باشد و به عبارت بهتر، خوشه های تشکیل شده از عوامل همسان بزرگتر شده و تعداد خوشه های محلی و پراکنده به حداقل رسیده باشد.

### ۳- اتوماتای یادگیر

اتوماتای یادگیر یک مدل انتزاعی است که به طور تصادفی یک عمل از مجموعه ی متناهی عمل های خود را انتخاب کرده و بر محیط اعمال می کند. محیط عمل انتخاب شده توسط اتوماتا را ارزیابی کرده و نتیجه ارزیابی خود را توسط سیگنال تقویتی به اتوماتای یادگیر اعلام می کند. اتوماتا با استفاده از عمل انتخاب شده و سیگنال تقویتی وضعیت داخلی خود را تغییر داده و سپس عمل بعدی خود را انتخاب می کند [۸]. در این مقاله از اتوماتای یادگیر به عنوان ابزار استفاده شده جهت بهبود هماهنگ سازی سیستم های چند عامله استفاده شده است. هدف نهایی آن است که اتوماتا یاد بگیرد تا از بین عمل های 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 شکل ۱ مشاهده می شود.



شکل ۱ ارتباط بین اتوماتای یادگیر و محیط

محیط را می توان توسط سه تایی  $E \equiv \{\alpha, \beta, c\}$  نشان داد که در آن  $\alpha \equiv \{\alpha_1, \alpha_2, \dots, \alpha_r\}$  مجموعه ی ورودیها،  $\beta \equiv \{\beta_1, \beta_2, \dots, \beta_m\}$  مجموعه ی خروجیها و  $c \equiv \{c_1, c_2, \dots, c_r\}$  مجموعه ی احتمالهای جریمه می باشد. هرگاه  $\beta$  مجموعه ی دو عضوی باشد، محیط از نوع P می باشد. در چنین محیطی  $\beta_1 = 1$  به عنوان جریمه و  $\beta_2 = 0$  به عنوان پاداش در نظر گرفته می شود. در محیط از نوع Q،  $\beta(n)$  می تواند به طور

گسسته یک مقدار از مقادیر محدود در فاصله ی [۰,۱] و در محیط از نوع S،  $\beta(n)$  متغیر تصادفی در فاصله [۰,۱] است.  $c_i$  احتمال اینکه عمل  $\alpha_i$  نتیجه نامطلوب داشته باشد می باشد. در محیط ایستا مقادیر  $c_i$  بدون تغییر می مانند، حال آنکه در محیط غیر ایستا مقادیر در طی زمان تغییر می کنند. اتوماتاهای یادگیر به دو گروه با ساختار ثابت و با ساختار متغیر تقسیم بندی می گردند. در ادامه اتوماتای یادگیر با ساختار متغیر معرفی می شود.

اتوماتای یادگیر با ساختار متغیر را می توان توسط چهار تایی  $\{\alpha, \beta, p, T\}$  نشان داد که  $\alpha = \{\alpha_1, \dots, \alpha_r\}$  مجموعه عملهای اتوماتا،  $\beta = \{\beta_1, \dots, \beta_m\}$  مجموعه ورودیهای اتوماتا،  $p = \{p_1, \dots, p_r\}$  بردار احتمال انتخاب هریک از عملها و  $p(n+1) = T[\alpha(n), \beta(n), p(n)]$  الگوریتم یادگیری می باشد. نحوه فعالیت این اتوماتا به صورت زیر است. اتوماتا یک عمل از مجموعه عمل های خود را به صورت تصادفی و مطابق بردارهای احتمال  $p_i$  انتخاب کرده و بر محیط اعمال می کند. اگر عمل انتخاب شده  $\alpha_i$  باشد، پس از دریافت پاسخ محیط، اتوماتا بردار احتمال عمل های خود را در صورت دریافت پاسخ مطلوب بر اساس رابطه (۱) و در صورت دریافت پاسخ نامطلوب طبق رابطه (۲) بروز می کند.

$$\begin{aligned} p_i(n+1) &= p_i(n) + a[1 - p_i(n)] \\ p_j(n+1) &= (1-a)p_j(n) \quad \forall j \quad j \neq i \end{aligned} \quad (1)$$

$$\begin{aligned} p_i(n+1) &= (1-b)p_i(n) \\ p_j(n+1) &= (b/r-1) + (1-b)p_j(n) \quad \forall j \quad j \neq i \end{aligned} \quad (2)$$

در روابط فوق  $a$  پارامتر پاداش و  $b$  پارامتر جریمه می باشد. با توجه به مقادیر  $a$  و  $b$  سه حالت زیر را می توان در نظر گرفت. زمانی که  $a$  و  $b$  با هم برابر باشند، الگوریتم را  $L_{RP}$  می نامیم، زمانی که  $b$  از  $a$  خیلی کوچکتر باشد، تحت عنوان الگوریتم یادگیری  $L_{R\&P}$  شناخته می شود و زمانی که  $b$  مساوی صفر باشد الگوریتم  $L_{RI}$  نامیده می شود [۹].

### ۴- دسته بندی عوامل پراکنده در محیط

#### ۴-۱- تعریف مسأله

در این قسمت به تعریف مسأله ی مد نظر در این مقاله خواهیم پرداخت. در این مسأله هدف از ایجاد هماهنگی میان عوامل پراکنده در محیط در حقیقت هدایت آنها به سمت مکانی است که در کنار عوامل همسان با خود قرار گیرند. به عبارتی، در این هماهنگی، هدف هدایت عوامل پراکنده در محیط به سمت محل تجمع عوامل همسان است. در ادامه هر یک از اجزای مسأله به طور مختصر تعریف می شود:

اتوماتای یادگیری که به عنوان *فعال/کننده*ی سایر اتوماتاهای یادگیر به کار برده می‌شود دارای دو عمل است: *فعال* و *غیرفعال*. در ابتدا احتمال انتخاب هر دو عمل برای این اتوماتا یکسان است. در صورتی که عمل انتخابی این اتوماتا *غیرفعال* باشد سایر اتوماتاهای یادگیر موجود در سلول غیر فعال می‌گردند. سایر اتوماتاهای یادگیر در هر سلول هر یک دارای ۸ عمل هستند. هر یک از این عمل‌ها متناظر با انتقال عامل به یکی از ۸ همسایگی سلول جاری می‌باشد.

```

For Each Active Cell (i, j)
{
- ComputeDensity( Agent_ID ) // By Each Agent
- Active = Activation_LA ( A_LA i,j) // By Activation-LA In Cell i,j
- If ( Active )
- {
- ComputeNextPlace( G_LA i,j) //By Goal-LA In Cell i,j
- MoveToNewPlace( Agent_ID) //By Agent
- }
- Else
- Stay_Put // By Agent
- FeedBack = Compute_Result ( Agent_ID) // By Agent

- If ( FeedBack) // By Goal-LA and Activation-LA In Cell i,j
- Award
- Else
- Penalty
}

```

شکل ۲ الگوریتم هماهنگ‌سازی با استفاده از اتوماتای یادگیر

شکل ۲ الگوریتم اجرا شده در هر سلول را در صورت ورود یک عامل به آن سلول نشان می‌دهد. در این الگوریتم، عامل پس از ورود به یک سلول، ابتدا به بررسی عوامل موجود در همسایگی خود و محاسبه‌ی میزان همانندی خود با سلولهای اطراف می‌پردازد. به عبارت بهتر عامل با این کار تعداد عوامل همسان موجود در ۸ سلول همسایه را محاسبه می‌کند. مجموع این عوامل همسان **وضعیت فعلی** عامل در مکان جاری را نشان می‌دهد. سپس اتوماتای یادگیر **فعال‌کننده** اقدام به انتخاب عمل در گام جدید می‌کند. این عمل یکی از دو عمل **فعال** یا **غیر فعال** است. در صورتی که عمل غیر فعال انتخاب شود به این معنی است که عامل در این گام در مکان فعلی خود باقی مانده و جابجا نخواهد شد. در غیر این صورت، با توجه به نوع عاملی که در سلول قرار دارد اتوماتای یادگیر متناظر با آن فعال شده و با انتخاب یکی از ۸ عمل خود عامل را به یکی از سلولهای همسایه هدایت می‌کند. سپس عامل به مکان جدید منتقل می‌شود. پس از استقرار تمامی عوامل در مکان جدید، هر عامل مجدداً به محاسبه‌ی وضعیت خود پرداخته و آن را با وضعیت قبلی که در ابتدای ورود به سلول قبلی محاسبه نموده بود مقایسه می‌نماید. نتیجه‌ی این مقایسه به عنوان بازخورد برای اتوماتاهای یادگیر سلول قبلی ارسال می‌شود. اتوماتای یادگیری که از عامل بازخوردی دریافت کند عمل خود را متناسب با آن بازخورد پاداش یا جریمه می‌نماید. اعمال پاداش برای عمل هر اتوماتا بر اساس رابطه‌ی ۱ و همچنین جریمه بر اساس رابطه‌ی ۲ صورت می‌گیرد. این روند در هر گام بصورت همزمان در تمامی سلولهای فعال اجرا می‌شود.

- محیط مسأله یک محیط سلولی دو بعدی است. در این محیط قانون همسایگی به این صورت است که هر سلول از محیط دارای ۸ همسایه است.

- عوامل در این سیستم موجودیتهای مستقلی هستند که دارای قدرت حرکت، ارزیابی محیط و بررسی وضعیت همسایگان خود هستند. همچنین، هر عامل دارای یک پارامتر هدف است که به صورت یک بردار تعریف شده و بیان کننده‌ی ویژگی‌ها و یا تمایلات عامل است.

- هر عامل قادر است در هر گام به اندازه‌ی یک سلول از مکان خود جابجا شود. به عبارتی در هر گام عامل می‌تواند به یکی از سلولهای همسایگی خود نقل مکان کند.

- برای تعیین میزان همانندی عوامل از پارامتر هدف هر عامل به عنوان معیاری جهت مقایسه‌ی میزان همانندی عوامل استفاده می‌شود. در نتیجه عواملی که پارامتر هدف مشابهی داشته باشند عوامل همسان هستند.

عوامل با توجه به پارامتر هدف خود به جستجو در محیط پرداخته و سعی در یافتن عواملی با پارامتر هدف همسان با خود را دارند. در بهترین حالت انتظار می‌رود که هر عامل در دسته‌ای قرار گیرد که تمامی عوامل همسان با عامل در آن قرار گرفته باشند. در این حالت، در حقیقت به نوعی خوشه بندی عوامل در محیط صورت پذیرفته است.

برای سهولت در پیاده‌سازی مسأله، فرض شده است که درکل  $D$  نوع عامل (دارای  $D$  هدف مختلف و یا به عبارتی  $D$  دسته‌ی مختلف) وجود دارد که هر عامل با توجه به بردار هدف خود در یکی از این دسته‌ها قرار خواهد گرفت. هر عامل باید به جستجو در محیط پرداخته و محل تجمع عواملی را بیابد که با او در یک دسته قرار دارند.

## ۴-۲- راهکار نخست: هماهنگی عوامل پراکنده مبتنی بر اتوماتاهای یادگیر

در این قسمت، به معرفی روش جدیدی مبتنی بر اتوماتای یادگیر برای هماهنگ‌سازی عوامل در یک محیط گسترده پرداخته می‌شود. به منظور حل این مسأله، فرض شده است که عوامل در یک محیط سلولی پراکنده شده‌اند و هر سلول از این فضا، مجهز به  $D+1$  اتوماتای یادگیر است. یکی از اتوماتاهای یادگیر به عنوان **فعال کننده**ی  $D$  اتوماتای یادگیر دیگر استفاده می‌شود. سایر اتوماتاهای موجود در هر سلول، هریک متناظر با یکی از  $D$  دسته از عوامل می‌باشند. یادآوری می‌شود که در مجموع فرض بر آن است که عوامل بر حسب بردار هدف خود متعلق به یکی از  $D$  دسته هستند. اتوماتای یادگیر متناظر هر نوع عامل در هر سلول، وظیفه‌ی کنترل حرکت تمامی عوامل از همان نوع را که ممکن است وارد آن سلول شوند بر عهده دارد.

#### ۴-۳- راهکار دوم: هماهنگی عوامل پراکنده مبتنی

##### بر بردار احتمال پویا

در این قسمت به ارائه‌ی روش جدید دیگری می‌پردازیم که برای حل مسأله‌ی گردآوری عوامل پراکنده در محیط ارائه شده است. در این روش هماهنگ‌سازی میان عوامل با بکارگیری یک **بردار احتمال پویا** صورت می‌پذیرد. فرض شده است که هر سلول به جای داشتن اتوماتاهای یادگیر، دارای  $D$  **بردار احتمال مجزا**  $\bar{P}_i$  برای هر نوع عامل  $i$  است. هر یک از بردارها دارای ۹ مقدار (مربوط به احتمال ماندن عامل در یک سلول و یا احتمال جابجا شدن عامل به یکی از ۸ سلول همسایه‌اش) است. هر عامل در هر گام با استفاده از این بردار و محاسبه‌ی بردار دیگری که بر اساس شرایط همسایگی عامل تولید شده و بردار چگالی همسایگی  $\bar{V}_{Neighbour}$  نامیده می‌شود، بردار احتمال نهایی  $\bar{P}_{iTotal}$  را محاسبه می‌نماید.

نحوه‌ی محاسبه‌ی **بردار چگالی همسایگی** عامل به این شکل است که هر عامل میانگین فراوانی عوامل همسان به خود را در چهار جهت بالا، پایین، چپ و راست محاسبه می‌کند. به عبارتی در هر گام، عامل از میان مقادیر **چگالی** تمامی عوامل همسان که در اطرافش وجود دارد، برای هر جهت میانگین‌گیری کرده و نتایج را به عنوان بردار وضعیت جاری در محاسبه‌ی بردار احتمال نهایی لحاظ می‌کند. در نتیجه بردار چگالی همسایگی با چهار مؤلفه در هر گام توسط عامل محاسبه می‌شود.

با ترکیب این بردار و بردار احتمال  $\bar{P}_i$ ، بردار احتمال نهایی  $\bar{P}_{iTotal}$  به صورت پویا محاسبه می‌شود. ترکیب این دو بردار، به این شکل صورت می‌گیرد که هر مؤلفه از بردار احتمال نهایی  $\bar{P}_{iTotal}$ ، از حاصل ضرب مقادیر بردار احتمال  $\bar{P}_i$  در مقادیر متناظر با جهت خود در بردار  $\bar{V}_{Neighbour}$  بدست می‌آید. به عبارت بهتر، مقادیر سه مؤلفه‌ی متناظر با همسایگان سلول در یک جهت، در مقدار مؤلفه‌ی متناظر با چگالی عوامل همسان در جهت بالا در بردار  $\bar{V}_{Neighbour}$  ضرب شده و به این ترتیب بردار احتمال نهایی محاسبه می‌شود.

محاسبه‌ی مکان بعدی هر عامل با استفاده از این بردار به این شکل است که عامل با در نظر گرفتن مقادیر احتمالها، مکان بعدی خود را به صورت غیر قطعی بر اساس این مقادیر انتخاب می‌کند. انتخاب غیر قطعی به این معنی است که به جای انتخاب قطعی مکانی که بیشترین احتمال را در بردار احتمال نهایی دارد، یکی از مقادیر بردار احتمال نهایی بر اساس میزان احتمال خود به صورت تصادفی انتخاب می‌شود. همچنین میزان دخالت **شرایط محلی** در تعیین بردار احتمال نهایی، به نسبت فراوانی عامل در آن مکان متفاوت است. به عبارت دیگر، هرگاه تعداد عوامل همسان در اطراف یک عامل بیش از یک حد معین باشد، بردار چگالی همسایگی با بردار احتمال سلول ترکیب می‌شود. در

غیر این صورت در شرایطی که تعداد عوامل مشابه در اطراف عامل کمتر از این حد معین باشد، بردار احتمال نهایی برابر با بردار احتمال  $\bar{P}_i$  در نظر گرفته می‌شود.

به منظور بهبود نتایج در طول اجرا، هر سلول در طی هر گام از شبیه‌سازی بردار احتمال سلول ( $\bar{P}_i$ ) را به نحوی تغییر می‌دهد که در نهایت مقادیر موجود در این بردار عوامل را به سمت مکان بهتر هدایت کند. با توجه به این که مجموع احتمالات بردار  $\bar{P}_i$  همواره برابر با ۱ است، اعمال تغییرات روی این بردار با قوانین پاداش (رابطه‌ی ۱) و جریمه (رابطه‌ی ۲) در اتوماتای یادگیر صورت می‌گیرد. در نتیجه مجموع مقادیر این بردار همواره برابر با ۱ باقی خواهد ماند. بنابراین در هر گام، در صورتی که عامل با انتخاب مکان بعدی به مکان مناسب‌تری از نظر چگالی عوامل همسان اطراف خود برود، احتمال  $\bar{P}_i$  برای عملی که منجر به این جابجایی شده است بیشتر می‌شود. به همین شکل در صورتی که این جابجایی منجر به استقرار عامل در مکانی نامناسب شود، امکان تکرار این جابجایی نامناسب برای عوامل همسان با کاهش میزان احتمال عمل مربوطه، کمتر خواهد شد. در شکل ۳ روند اجرای این روش در سلولهای فعال بیان شده است.

```

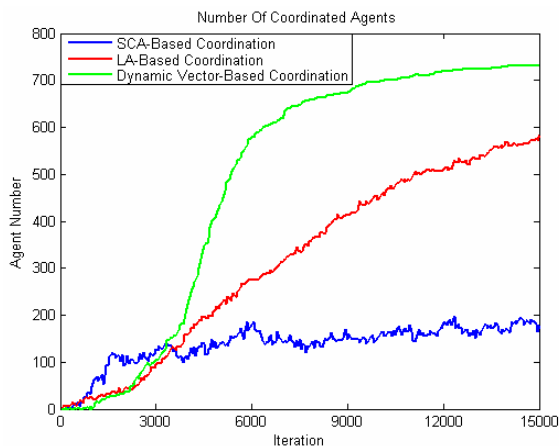
For Each Active Cell (i,j) // In Parallel
{
- Agent_Density = ComputeDensity (Agent_ID)
- If (Agent_Density > L)
- {
- Average_V = ComputeAverage (Agent_ID)
- Total_V = CreateTotalVector (Average_V, Cell_V {Agent_Goal})
- Find_Best_Movement (Total_V)
- }
- Else
- {
- Find_Best_Movement (Cell_V {Agent_Goal})
- }
- MoveToNewPlace (Agent_ID)
- FeedBack = Compute_Result (Agent_ID)
- If (FeedBack)
- Award (Cell_V {Agent_Goal})
- Else
- Penalty
}

```

شکل ۳ الگوریتم هماهنگ‌سازی مبتنی بر بردار احتمال پویا

#### ۵- ارزیابی

به منظور ارزیابی و مقایسه‌ی روشهای ارائه شده در این مقاله، ابتدا مسأله مجدداً با روش مبتنی بر اتوماتای سلولی تصادفی که در مرجع [۷] ارائه شده بود حل شد. در شکل ۶ وضعیت عوامل در محیط در طی گامهای مختلف اجرا نشان داده شده‌است. به این منظور با در نظر گرفتن  $D=5$  نوع عامل مختلف در محیطی با ابعاد  $50 \times 50$  سلول، مسأله شبیه‌سازی شد. در شبیه‌سازی انجام شده، ۸۰۰ عامل در محیط پراکنده شده‌اند. سپس با در نظر گرفتن شرایط اولیه‌ی یکسان، به حل مجدد مسأله با دو روش ارائه شده پرداخته شده و نتایج بدست آمده در جدول ۱ ذکر شده است. در این جدول تعداد عوامل هماهنگ شده از



شکل ۴ نمودار مقایسه‌ی تعداد عوامل هماهنگ شده با بیش از ۶ همسایه‌ی همسان در هر سه روش هماهنگ‌سازی

با توجه به این که اساس کار هماهنگ‌سازی در هر سه روش بر مبنای جابجایی عوامل در محیط و یافتن مکان مناسب‌تر است، لذا در صورتی که جابجایی هر عامل هزینه‌ای در بر داشته باشد لازم است تعداد جابجایی‌ها برای رسیدن به یک وضعیت مناسب نیز به عنوان یک فاکتور مهم برای تعیین روش مناسب لحاظ شود. در مقایسه‌ی دیگری که روی شبیه‌سازی با هر سه روش هماهنگ‌سازی در شکل ۵ انجام شده، تعداد عوامل متحرک در هر گام به عنوان معیار مقایسه در نظر گرفته شده است.

همچنین در روش مبتنی بر بردارهای احتمال پویا، با توجه به شکل ۸ مشاهده می‌شود که پس از طی حدود ۲۰۰۰ گام زمانی سیستم در برخی نقاط به تعادل رسیده و پس از حدود ۵۰۰۰ گام به وضعیتی نزدیک می‌شود که در آن حداقل برای هر دسته از عوامل همرنگ یک مجموعه ایجاد کرده‌است. در نهایت در گام ۱۵۰۰۰ ام، بیش از ۹۲٪ عوامل در مکان مناسب قرار گرفته‌اند.

مشاهده می‌شود که در هر گام سیستم در جهت گردآوری عوامل پراکنده‌ی بیشتری پیش می‌رود و رشد توده‌های عوامل همسان به مرور سرعت می‌گیرد. با توجه به این که بردارهای احتمال در طی گامهای اجرا به‌روز رسانی می‌شوند، به تدریج عامل با ورود به یک سلول می‌تواند از اطلاعات موجود در آن استفاده کرده و با استفاده از اطلاعات محلی مکان جاری خود برای تشکیل بردار احتمال نهایی مناسب‌ترین جهت برای حرکت خود را انتخاب کند. در این روش به دلیل استفاده از اطلاعات جاری همسایگی عامل در تعیین مکان بعدی، تعداد دسته‌های تشکیل شده برای هر نوع عامل نسبت به روش مبتنی بر اتوماتای یادگیر کاهش یافته است.

هر یک از  $D$  نوع عامل به طور مجزا ذکر شده‌است. منظور از عوامل هماهنگ‌شده در این جدول عواملی هستند که در انتهای شبیه‌سازی بیش از ۶ همسایه‌ی همسان داشته و در مکان خود ثابت شده‌باشند. در این جدول تعداد عوامل هماهنگ شده در هر روش در گام ۲۰,۰۰۰ ام اجرا نشان داده شده است. با توجه به این جدول مشاهده می‌شود که تعداد عوامل قرار گرفته در مکان مناسب که از روش مبتنی بر بردار احتمال تصادفی بهره گرفته‌اند نسبت به دو روش دیگر بیشتر است.

به منظور ارزیابی دقیق‌تر هر روش، مقایسه‌ای بر اساس معیارهای مختلف انجام شده است. در نمودار شکل ۴ مقایسه‌ای میان تعداد عوامل هماهنگ شده در هر گام برای هر سه روش نشان داده شده‌است. مطابق این نمودار مشاهده می‌شود که روش هماهنگ‌سازی با استفاده از بردار احتمال پویا نتایج بسیار بهتری نسبت به دو روش قبل ارائه کرده‌است.

پس از انجام شبیه‌سازی در طی ۱۵۰۰۰ گام در روش مبتنی بر اتوماتای یادگیر، مشاهده می‌شود که دسته‌های تشکیل شده از عوامل همسان به مرور رشد کرده و عوامل بیشتری در مکان مناسب مستقر می‌شوند. نتایج شبیه‌سازی این روش در شکل ۷ ارائه شده‌است.

با اجرای ۱۰۰ شبیه‌سازی برای دسته‌های مختلف عوامل با پراکندگی متفاوت، مشاهده می‌شود این روش در تمامی شبیه‌سازی‌ها قادر به هدایت عوامل به سمت مکان مناسب و در نتیجه رسیدن سیستم به سمت یک حالت پایدار هدایت است که در آن بیش از ۷۲٪ عوامل در مکان مناسب قرار گرفته باشند.

نتایج بررسی وضعیت بردار احتمال عمل در اتوماتاهای یادگیر نشان می‌دهد در سلولهایی از محیط که دسته‌های پایدار عوامل همسان در آن‌ها قرار گرفته‌اند، اتوماتای یادگیر فعال کننده‌ی هر سلول به مرور زمان با افزایش احتمال عمل "غیر فعال" منجر به حفظ عوامل در مکان خود و در نتیجه پایداری دسته‌های تشکیل شده است. همچنین در سایر نقاط محیط که هیچ دسته‌ای از عوامل در آن تشکیل نشده است اتوماتای یادگیر فعال کننده به مرور احتمال عمل "فعال" خود را افزایش داده است. به این ترتیب عوامل امکان توقف در مکانهای نامناسب را نداشته و با توجه به عملی که توسط اتوماتای یادگیر متناظر با عامل اتخاذ می‌شود ملزم به جابجایی است.

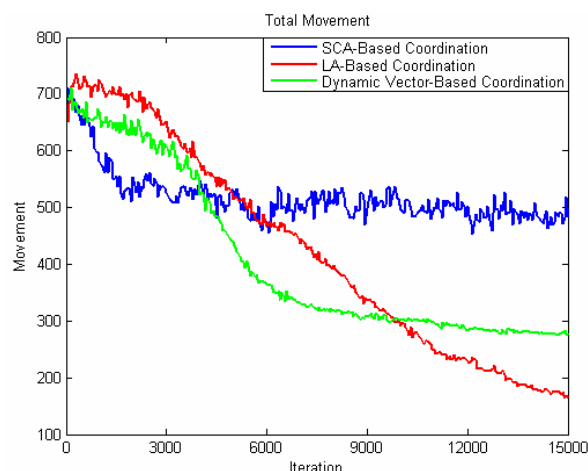
در اتوماتاهای یادگیر متناظر با نوع عواملی که در یک دسته قرار گرفته‌اند، میزان احتمال عمل متناظر با حرکت عامل به سمت مراکز دسته‌ها در محل تشکیل دسته‌ها نسبتاً بیشتر از سایر مقادیر است. در سایر اتوماتاهای یادگیر موجود در سلولهایی که هیچ دسته‌ای در آنها تشکیل نشده است نیز تغییر محسوسی مشاهده نشد. با توجه به این که اتوماتاهای یادگیر با استفاده از اطلاعات کاملاً محلی به‌روز رسانی شده و عمل بعدی خود را انتخاب می‌کنند، نتایج شبیه‌سازی‌های متعدد نشان می‌دهد که این روش منجر به ایجاد بیش از یک دسته برای هر گروه از عوامل شده‌است.

جدول ۱ مقایسه‌ی تعداد عوامل گردآوری شده در یک محیط با استفاده از روشهای مختلف

## ۵-۱- خوشه‌بندی عوامل با استفاده از اتوماتاهای یادگیر

نتایج به دست آمده از شبیه‌سازی روشهای مختلف ارائه شده نشان می‌دهد که با استفاده از ابزارهای ساده‌ای با قابلیت یادگیری و اصلاح، نتایج قابل توجهی از هماهنگ‌سازی عوامل پراکنده در محیط قابل حصول است. لیکن در تمامی این روشها به دلیل محلی بودن اطلاعات و وسعت محیط، نتایج کلاستر کردن و خوشه‌بندی عوامل کاملاً ایده‌آل نبوده و ممکن است برای یک گروه از عوامل همسان، بیش از یک دسته تشکیل شود. از طرفی با توجه به سهولت پیاده‌سازی و محاسن استفاده از روش‌های مبتنی بر یادگیری برای هماهنگ‌سازی عوامل، در صورتی که از تشکیل بیش از یک دسته برای هر نوع عامل جلوگیری شود، نتایج بدست آمده در حوزه‌ی داده کاوی و خوشه‌بندی اطلاعات نیز مفید خواهد بود. بنابراین می‌توان با تجهیز عوامل به امکاناتی برای مشاهده و دریافت اطلاعات وسیع‌تری از محیط اطراف خود، عمل دسته‌بندی عوامل را بهینه کرده و نتایج را به سمت پاسخ‌انده‌آل بهبود بخشید.

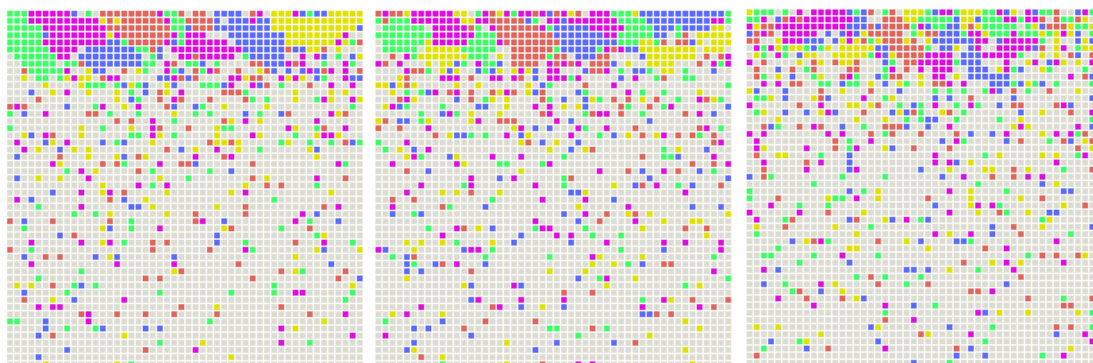
یکی از روشهایی که می‌توان از آن برای جلوگیری از تشکیل دسته‌های کوچک و پراکنده استفاده کرد آن است که از ابتدا برای هر نوع عامل در محیط، یک مکان معین به عنوان **مرکز تجمع** در نظر گرفته شود. به عبارت بهتر، نقاطی از محیط به عنوان **مراکز تجمع عوامل** تعریف شود. در نتیجه تمامی عوامل باید در نهایت به مرکز تجمع عوامل همسان با خود نزدیک شده و در نهایت همگی در یک دسته قرار گیرند. تمامی عوامل برای یافتن این نقاط در محیط به جستجو پرداخته و در صورت نزدیک شدن به این مراکز در مکان مناسب مستقر خواهند شد. فرض بر این است که تعداد و مکان دسته‌ها از قبل معین است لیکن عوامل از مکان مرکز تجمع عوامل همسان خود کاملاً بی اطلاع هستند.



شکل ۵ نمودار مقایسه‌ی تعداد جابجایی عوامل در هر گام با استفاده از سه روش هماهنگ‌سازی

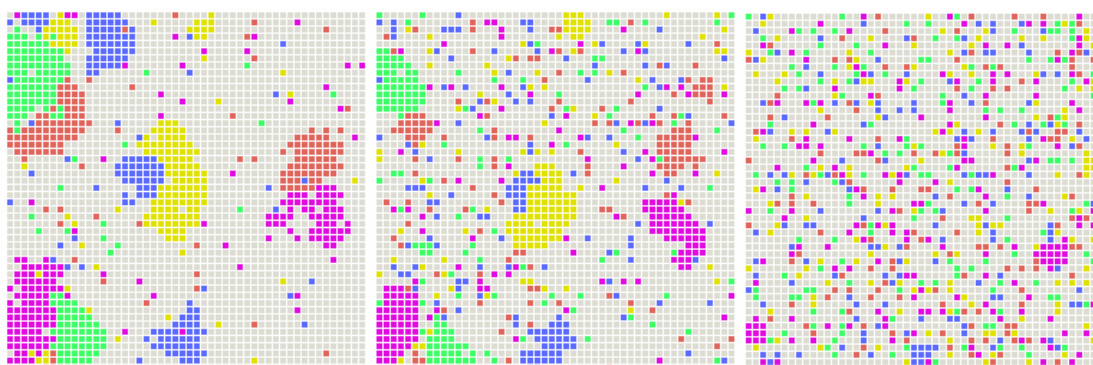
در نمودار شکل ۵ مشاهده می‌شود که تعداد جابجایی عوامل در دو روش مبتنی بر اتوماتای یادگیر و بردار احتمال پویا در ابتدای شبیه‌سازی تا حدود گام ۴۵۰۰م، بیشتر از روش مبتنی بر اتوماتای سلولی تصادفی است. همچنین در مقایسه‌ی میزان جابجایی عوامل در هماهنگ‌سازی با دو روش پیشنهادی مشاهده می‌شود که از ابتدای شبیه‌سازی تا گام ۱۰۰۰۰، روش مبتنی بر بردار احتمال پویا به نسبت روش مبتنی بر اتوماتاهای یادگیر تعداد جابجایی کمتری دارد. از گام ۱۰۰۰۰ به بعد نرخ جابجایی در روش مبتنی بر اتوماتاهای یادگیر به سرعت کاهش یافته و با یک سیر نزولی نسبتاً خطی در هر گام از تعداد عوامل متحرک کاسته می‌شود.

تعداد عوامل	روش مبتنی بر اتوماتای سلولی تصادفی	روش مبتنی بر اتوماتای یادگیر	روش مبتنی بر بردار احتمال پویا
$D_1=194$	35	106	179
$D_2=106$	59	83	96
$D_3=117$	28	98	110
$D_4=228$	49	194	219
$D_5=155$	61	123	143
Total=1000	24%	76%	92/5%



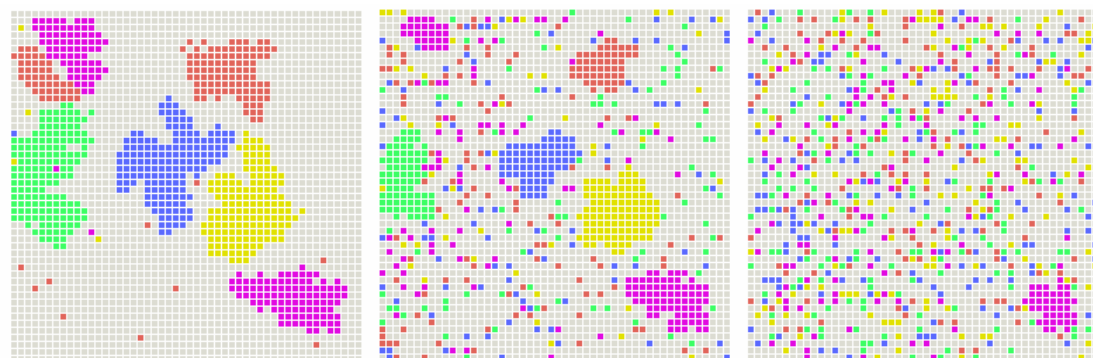
الف) گام ۲۰۰۰      ب) گام ۵۰۰۰      ج) گام ۱۵۰۰۰

شکل ۶ پراکندگی عوامل هماهنگ‌سازی شده با اتوماتاهای تصادفی در گامهای مختلف اجرا



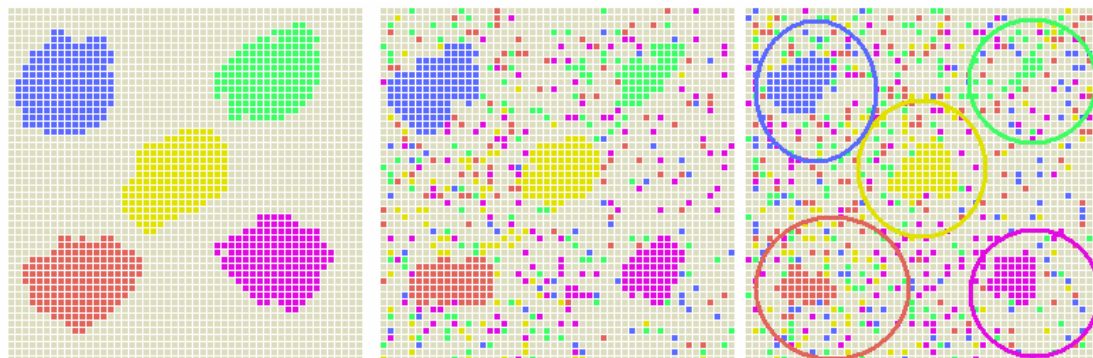
الف) گام ۲۰۰۰      ب) گام ۵۰۰۰      ج) گام ۱۵۰۰۰

شکل ۷ پراکندگی عوامل هماهنگ‌سازی شده با اتوماتاهای یادگیر در گامهای مختلف اجرا



الف) گام ۲۰۰۰      ب) گام ۵۰۰۰      ج) گام ۱۵۰۰۰

شکل ۸ پراکندگی عوامل هماهنگ‌سازی شده روش مبتنی بر بردار احتمال پویا در گامهای مختلف اجرا



الف) گام ۱۰۰۰ ام      ب) گام ۳۰۰۰ ام      ج) گام ۱۰۰۰ ام

شکل ۹ خوشه بندی عوامل با استفاده از روش هماهنگ سازی مبتنی بر اتوماتاهای یادگیر و مراکز تجمع از پیش تعیین شده در گامهای مختلف اجرا



به این ترتیب لازم است هر عامل به نحوی از مکان این مرکز مطلع شود. بدین منظور، از محل **مراکز تجمع** عوامل سیگنال با شعاع انتشار محدود منتشر می‌شود. با انتشار این سیگنال از مکانی که به عنوان مرکز تجمع یک نوع عامل معین در نظر گرفته می‌شود، می‌توان عامل را از مکان این مرکز مطلع کرد. همچنین عوامل موجود در محیط نیز مجهز به ابزاری جهت دریافت این سیگنالها بوده و می‌توانند نوع عاملی که باید در آن مرکز قرار گیرد را تشخیص دهند. در صورتی که عاملی در شعاع انتشار سیگنالی باشد که مربوط به مرکز تجمع عوامل همسان با خود است، از این سیگنال به عنوان بازخورد محیط استفاده می‌کند. اگر حرکت عامل در یک گام باعث قوی‌تر شدن سیگنال شود، نشان دهنده‌ی نزدیکتر شدن عامل به **مرکز تجمع** عوامل همسان با خود است. بنابراین عامل به اتوماتای یادگیری که منجر به این جابجایی شده است بازخورد مثبت خواهد داد و در صورت ضعیف‌تر شدن این سیگنال بازخورد منفی می‌دهد.

شعاع انتشار مرکز تجمع هر نوع عامل محدود بوده و در نتیجه هر عامل تنها با قرار گرفتن در فاصله‌ای معین از این فاصله قادر به دریافت سیگنال‌های این مرکز است. در صورتی که عامل فاصله‌ی زیادی از مکان نهایی خود داشته و به عبارتی هیچ سیگنالی از مرکز تجمع عوامل هم نوع خود دریافت نکند، باید همچنان به جستجو در محیط ادامه دهد و نمی‌تواند در مکانی مستقر شود که از مرکز تجمع عوامل همسانش دورتر است. استفاده از این سیگنال منجر به عدم تشکیل دسته‌های پراکنده‌ی عوامل از یکدیگر شده و عوامل را ملزم به یافتن مرکز تجمع عوامل همسان خود می‌کند.

در شکل ۹ وضعیت سیستم در گامهای مختلف اجرا نشان داده شده است. با توجه به شکل ۹ (الف) شعاع انتشار مراکز تجمع برای هر نوع عامل مشخص شده است. مشاهده می‌شود که در نهایت در حدود گام ۱۰۰۰ ام، دسته‌ها به طور کامل شکل گرفته اند.

## ۵-۲- نتیجه‌گیری

در این بررسی مشاهده شد که در حل مسأله‌ی هماهنگ‌سازی سیستم‌های چندعامله، اتوماتاهای یادگیر به عنوان ابزارهای یادگیری ساده، قادر به بهبود رفتار خود در طول زمان بوده و در نتیجه نتایج قابل توجهی در زمینه‌ی ایجاد هماهنگی میان عوامل یک سیستم چندعامله ارائه می‌کنند. نتایج نشان می‌دهد که هماهنگی ایجاد شده در مقایسه با روش مبتنی بر اتوماتای سلولی تصادفی، پایدارتر و همچنین تعداد عوامل هماهنگ‌شده بسیار بیشتر است.

به منظور دخالت دادن شرایط جاری محیط بر تعیین مکان عوامل، روش دیگری در این مقاله ارائه شد که به جای استفاده از اتوماتاهای یادگیر از چند بردار احتمال در هر سلول بهره می‌برد. در این روش با استفاده از این **بردارهای احتمال** و ترکیب شرایط

**فعلی** هر عامل با این بردارها، یک **احتمال نهایی پویا** تشکیل شده و انتخاب مکان بعدی هر عامل با استفاده از این بردار انجام می‌شود. بردارهای احتمال هر سلول در هر گام به منظور بهبود رفتار سراسری سیستم با دریافت بازخورد از عامل به‌روز رسانی می‌شود. نتایج بدست آمده نشان می‌دهد که استفاده از اطلاعات جاری هر گام و تشکیل بردار احتمال نهایی به صورت پویا نتایج سراسری را بهبود بخشیده و تعداد دسته‌های ایجاد شده برای یک نوع عامل را نسبت به دو روش قبل کاهش داده است.

با توجه به ارزیابی‌های انجام شده مشاهده می‌شود که تاثیر یادگیری در نتایج هماهنگ‌سازی سیستم‌های چند عامله بسیار قابل توجه است. همچنین به منظور بهبود روش ارائه شده در حل مسائل خوشه‌بندی با استفاده از مکانهایی از پیش تعیین شده به عنوان مراکز تجمع عوامل و انتشار سیگنالی از این مراکز برای عوامل موجود در محیط، مسأله را مجدداً حل کرده و پاسخ بهینه برای خوشه بندی عوامل پراکنده در سیستم بدست آمد.

## ۶- مراجع

- [۱] K. P. Sycara, "Multiagent Systems," *AI Magazine*, Intelligent Agents, vol. 19, no.2, pp.79-97, 1998.
- [۲] Xuan, P. and Lesser, V., "Multi-Agent Policies: From Centralized Ones to Decentralized Ones," *Proceedings of the First International Joint Conference on Autonomous Agents and Multi Agent Systems*, ACM Press, Bologna, Italy, pp. 1098-1105, 2002.
- [۳] L.F. Schroeder, and A.L.C. Bazzan, "A Multi-agent System to Facilitate Knowledge Discovery: an application to Bioinformatics," In *Proc. of the Workshop on Bioinformatics and Multi-Agent Systems*, pages 44-50, Bologna, Italy, 2002.
- [۴] Katholieke Universiteit Leuven, Celestijnenlaan, "The design of Multi-Agent coordination and control systems using Stigmergy", *Dept. of Mech. Engineering, Division P.M.A., B-3001 Heverlee (Leuven)*, Artificial Life 5 (1999), pp. 97-116.
- [۵] P. Valckenaers, M. Kollingbaum, H. Van Brussel, O. Bochmann, and C. Zamfirescu, "The design of multi-agent coordination and control systems using stigmergy," In *Proc. Of the 3<sup>rd</sup> International Workshop on Emergent Synthesis (IWES'01 Conference)*, Bled, Slovenia, pp. 97-116, 2001.
- [۶] A.V. Moere, J. J. Clayden, and A. Dong, "Data Clustering and Visualization using Cellular Automata Ants"

[۹] محمدرضا میبیدی، حمید بیگی و مسعود طاهرخانی،  
« اتوماتای یادگیر سلولی»، در مجموعه مقالات ششمین  
کنفرانس انجمن کامپیوتر ایران، ص ۱۶۳-۱۵۳، ۱۳۷۹.

[۷] T. D. Barfoot and G. M. T. D'Eleuterio,  
“Multiagent Coordination by Stochastic Cellular  
Automata”, *Presented at the International Joint  
Conference on Artificial Intelligence  
(Proceeding of IJCAI)*, Seattle, USA, 4–10  
August 2001.

[۸] K.S. Narendra and M.A.L. Thathachar, “Learning  
Automata: An Introduction,” Prentice-Hall,  
Englewood Clis, NJ, USA, 1989.

## زیر نویس ها

---

<sup>1</sup>Learning Automata

<sup>2</sup>Clustering  
Classification

<sup>4</sup>Inductor

<sup>5</sup>Ant Colony

<sup>6</sup>Stigmergy

<sup>7</sup>Cellular Ant