# استخراج گذرگاه‌ها با استفاده از تشخیص اشیا در یادگیری تقویتی

بهزاد غضنفری، ناصر مزینی و محمدرضا جاهد مطلق

**لیده: این مقاله روش جدیدی را مطرح می‌کند که قادر به استخراج ه‌ها به‌صورت اتوماتیک برای عامل یادگیری تقویتی است. روش پیشنهادی ستم‌های بیولوژیکی، رفتار و مسیریابی حیوانات الهام گرفته شده است و طه تعاملات عامل با محیط پیرامونی‌اش عمل می‌کند. عامل با استفاده از بندی و تشخیص اشیا به‌صورت سلسله مراتبی، نشانه‌هایی را پیدا می‌کند. ن نشانه‌ها در فضای اقدام به هم نزدیک باشند، گذرگاه‌ها با استفاده از های بین آنها استخراج می‌شوند. نتایج آزمایش‌ها بهبود قابل ملاحظه‌ای را ایند یادگیری تقویتی در مقایسه با سایر روش‌های مشابه نشان می‌دهد.**

**ید واژه: یادگیری تقویتی، خوشه‌بندی اشیا، یادگیری تقویتی سلسله مراتبی، ت گسترش‌یافته زمانی.**

## ۱- مقدمه

یادگیری تقویتی [1] (RL) در صدد هستیم تنها با استفاده از پاداش و ه عامل را برنامه‌ریزی کنیم، بدون این که به عامل بگوییم که ه این وظیفه را انجام دهد. عامل بایستی اقداماتی که باعث افزایش دت مجموع ارزش سیگنال‌های تقویتی می‌شوند را انتخاب کند. این کار را می‌تواند در طول زمان از طریق قاعده‌مند کردن آزمایش لا یاد بگیرد [۱].

لی‌رغم موفقیت روش‌های موجود در یادگیری تقویتی، در مسایلی با بالا این روش‌ها به خوبی مقیاس‌پذیر نیستند. تلاش‌های اخیر برای بر معضل ابعاد [2] بالا به سمت شیوه‌های مبتنی بر تجرید [3] در RL برده است، که به‌صورت طبیعی منجر به معماری کنترلی سلسله مراتبی و یتم‌های یادگیری وابسته به آن می‌شوند [۲] و [۳]. در اکثر موارد راه های سلسله مراتبی جواب‌های نزدیک به بهینه‌ای را در کارایشان می‌دهند و همچنین هزینه مناسب‌تری را در زمان اجرا، زمان یادگیری سای مورد نیاز برای حل مسایل در مقابل تکنیک‌های RL محض می‌کنند [۱].

غلب تحقیقات در RL مبتنی بر تئوری چارچوب فضا و اقدام گسسته ، از فرایند تصمیم‌گیری مارکوف [4] (MDP) است. چارچوب MDPs ورت مرسوم شامل اقدامات گسترش‌یافته زمانی [5] نیستند. آنها اصولاً ساس گام‌های گسسته زمانی هستند: اقدامی که در زمان $t$ انتخاب می‌شود، بر حالات و پاداش در زمان $t+1$ تأثیر می‌گذارد. اساساً هیچ مفهومی از شیوه انجام کار که تأکید بر روی بازه متغیری از زمان داشته باشد، وجود ندارد. در نتیجه MDPs به‌صورت متداول قادر به استفاده از مزایای ساده‌سازی و کارآمدی که در سطوح بالاتر از تجرید زمانی وجود دارد، نیستند. تجرید زمانی می‌تواند درون یادگیری تقویتی به شکل‌های متفاوتی مطرح شود [۴]. یکی از مهم‌ترین ویژگی‌های انجام تجرید زمانی در چارچوب یادگیری تقویتی، بنانهادن تئوری فرایندهای تصمیم‌گیری شبه مارکوف [6] (SMDPs) است. همان گونه که گفته شد، در MDP فرض می‌شود که عامل دارای توانایی دستیابی کامل به حالت محیط است و هر اقدامش را در یک گام زمانی انتخاب می‌کند. SMDP در فرض دوم این محدودیت را ندارد و این امکان را به عامل می‌دهد که اقداماتش را در چندین گام زمانی انجام دهد [۴].

خوشبختانه اگرچه تعداد زیادی از مسایل یادگیری تقویتی بسیار پیچیده هستند، آنها اغلب به‌صورت سلسله مراتبی قابل تجزیه به یک سری از زیروظایف ساده‌تر هستند. انسان‌ها نیز از چنین ساختارهای سلسله مراتبی تجزیه‌شدنی در حل مسایلی که پیچیده و دارای مقیاس بزرگی هستند، بهره می‌برند [۳].

همان طور که می‌دانیم، اقدامات توسعه داده شده زمانی در یادگیری تقویتی سلسله مراتبی [7] (HRL) مورد مطالعه قرار گرفته‌اند. HRL یک چارچوب کلی برای مقیاس پذیر کردن RL به مسایلی با فضای حالات بزرگ با استفاده از ساختار کار (یا اقدام) برای محدودکردن فضای خط مشی‌ها است. اصل کلیدی که HRL در بر گرفته است، توسعه الگوریتم‌های یادگیری است که به یادگیری خط مشی‌ها از ابتدا نیازی ندارند، بلکه در عوض از خط مشی‌های موجود برای زیروظایف ساده‌تر (یا ماکرو- اقدامات) دوباره استفاده می‌کنند و استراتژی تقسیم و غلبه را به کار می‌برند.

SMDP به‌عنوان مدل آماری شناخته‌شده‌ای برای رفتار با اقداماتی با طول‌های متغیر است. در دهه اخیر کارهای زیادی روی مدل SMDP بسط داده شده‌اند که این کارها به‌صورت مدل‌های کار سلسله مراتبی از زیروظایف سطح پایین‌تر هستند که به‌صورت جزئی یا کامل مشخص شده‌اند. این کارها منجر به توسعه مدل‌های HRL قدرتمندی مانند سلسله مراتبی از ماشین‌های مجرد [8] (HAMs) [۵]، Optionها [۴] و MAXQ [۶] شده است. در مدل Options بررسی شده که چگونه خط مشی‌های سراسری را با معلوم‌بودن خط مشی‌های کاملاً معینی برای اجرای زیروظایف یاد بگیریم. در قالب رسمی از HAMs نشان داده شده است که چگونه یادگیری تقویتی می‌تواند حتی زمانی که خط مشی‌ها برای پایین‌ترین سطوح زیروظایف، تنها به‌صورت جزئی مشخص شده‌اند، انجام وظیفه نماید. مدل MAXQ از اولین روش‌های مطرح‌شده است که در آن تجرید زمانی با تجرید حالات ترکیب شده است. ویژگی اصلی

ن مقاله در تاریخ ۱۴ شهریور ماه ۱۳۹۰ دریافت و در تاریخ ۲۰ خرداد ماه ۱۳۹۱ ری شد.

هزاد غضنفری، دانشکده مهندسی کامپیوتر، دانشگاه علم و صنعت ایران، تهران، (email: be_ghazanfari@ieee.

صر مزینی، دانشکده مهندسی کامپیوتر، دانشگاه علم و صنعت ایران، تهران، (email: mozayani@iust.a

حمدرضا جاهد مطلق، دانشکده مهندسی کامپیوتر، دانشگاه علم و صنعت ایران، ، (email: jahedmr@iust.ac.ir)

1. Reinforcement Learning
2. Curse of Dimensionality
3. Abstraction
4. Markov Decision Process
5. Temporally Extended Action
6. Semi-Markov Decision Processes
7. Hierarchical Reinforcement Learning
8. Hierarchies of Abstract Machines

MAXQ نمایش تفکیک‌شده از تابع ارزش است [۶].

Optionها تجرید زمانی دانش و اقداماتی که در چارچوب یادگیری تقویتی قرار گرفته می‌شوند را در شیوه‌ای طبیعی و عمومی ممکن می‌سازند. در شیوه پیشنهادی از چارچوب Option برای تعریف اقدامات گسترش‌یافته زمانی استفاده شده است. یک Option سه گانه $\langle I,\pi,B\rangle$ است در جایی که $I$ مجموعه شروع است. $I \subseteq S$ مجموعه‌ای است که هر اقدام گسترش‌یافته زمانی می‌تواند از آن مجموعه فراخوانی شود. $\pi : S^{+}A \rightarrow [۰,۱]$ تابع خط مشی برای هر حالت در مجموعه آغازین از اقدامات گسترش‌یافته زمانی است که این خط مشی نگاشتی به توالی از اقدامات است. $\beta : S^{+} \rightarrow [۰,۱]$ شرط خاتمه را مشخص می‌کند که برای هر Option در هر حالتی با چه احتمالی می‌تواند خاتمه بیابد.

می‌توان نشان داد که Optionها قادر به استفاده‌شدن به‌صورت تبادل‌پذیری با اقدامات ابتدایی[1] در روش‌های برنامه‌ریزی پویا[2] (DP) و در روش‌های یادگیری مانند یادگیری Q هستند. برای ارزیابی ارزش هر اقدام و حالت از جدولی به‌عنوان جدول Q استفاده می‌شود و برای اطلاعات بیشتر در مورد یادگیری Q به [۱] مراجعه شود.

در این نوشتار همانند سایر مقالاتی که در این زمینه داده شده است، تنها اقدامات گسترش یافته زمانی به جدول یادگیری Q که از جدول حالت و اقدامات ابتدایی استفاده می‌کند، اضافه می‌شوند. (اقدامات گسترش‌یافته شده زمانی متناظر با مجموعه حالاتی که در آنها تعریف شده‌اند، به مجموعه اقدامات اولیه آن حالات اضافه می‌شوند). بر اساس [۳] و [۴] مقادیر جدول Q در چارچوب SMDP تنها برای اقدامات گسترش‌یافته زمانی در حالتی که انتخاب شده به روز می‌شوند. در این مقاله همانند [۷] و [۸] بر اساس (۱) به‌روز رسانی مقادیر جدول Q انجام می‌گیرد. در واقع اقداماتی به مجموعه اقدامات عامل اضافه می‌گردند و نیاز به ساختار خاص دیگری نیست -برای اطلاعات بیشتر به [۳] و [۴] مراجعه شود-. هدف، تنها ارائه روشی برای استخراج گذرگاه است تا در فرایند یادگیری تسریع ایجاد کند و زمینه را برای انتقال دانش فراهم نماید. فرایند یادگیری مشابه با [۷] تا [۱۲] است

$$\begin{aligned} Q(s_t,o_t) &= Q(s_t,o_t) + \alpha(n(t,s_t,o_t)) \\ &\times[\gamma^{\tau}\max_{a'\in A'(s_t+\tau)} Q(s_{t+\tau},a') - Q(s_t,o_t) \\ &+ r_t + \gamma r_{t+1} + \ldots + \gamma^{\tau-1} r_{t+\tau-1}] \end{aligned} \qquad (۱)$$

که (۱) به‌روز رسانی یادگیری را انجام می‌دهد، در جایی که $\tau$ زمان واقعی برای Option، $o_t$، است $\alpha(n(t,s_t,o_t))$ تابع نرخ یادگیری و آرگومان ورودی آن تعداد دفعاتی است که $o_t$ در حالت $s_t$ تا زمان $t$، تجربه شده است [۳].

عموم الگوریتم‌هایی که به استخراج گذرگاه می‌پردازند با استفاده از الگوریتم‌های پارتیشن‌بندی گراف مانند [۸]، [۱۱] و [۱۲]، یا به بررسی نقش هر حالت در مسیرها مانند [۹]، [۱۳] و [۱۴] به دنبال پیدا کردن گذرگاه هستند. در روش‌های پارتیشن‌بندی به طرق مختلف عموماً ترکیبات مختلف حالت‌ها بر شماری می‌شود و معیارهایی برای خوشه‌ها چک می‌شود تا بتوانند گراف را خوشه‌بندی کنند. در روشی مانند [۱۴] معیارهایی برای هر حالت در مسیرهای متفاوت بررسی می‌شوند. مثلاً برای هر مسیری که بین دو حالت وجود دارد، حالت‌هایی که در کم هزینه‌ترین مسیر بین این دو حالت قرار دارند استخراج می‌شوند و شایستگی آنها برای گذرگاه‌بودن افزایش می‌یابد. این عملیات‌ها دارای پیچیدگی محاسباتی بالایی هستند و همچنین دقت آنها به اندازه خوشه‌ها در فضای حالت بستگی دارد. نوآوری روش پیشنهادی این است که با توجه به تأثیر اقدامات، اشیا و مفاهیم استخراج می‌شوند و با استفاده از سلسله مراتب تجرید نقاط کلیدی برای پیداکردن گذرگاه استخراج می‌شوند.

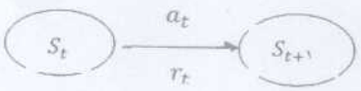
در بخش ۲ پیش‌زمینه‌ای از شیوه پیشنهادی و در بخش ۳ کارهای مشابهی که در این زمینه انجام شده، ارائه شده است. بخش ۴ به بررسی و توضیح روش پیشنهادی و هر کدام از اجزایش و همچنین به مزایا و پیچیدگی آن اختصاص دارد. ارائه و شرح آزمایش‌ها و نتایج در بخش ۵ آمده است و در نهایت در بخش ۶ جمع‌بندی و نتیجه‌گیری از این نوشتار ارائه شده است.

## ۲- پیش‌زمینه

اجازه دهید توجیه روش پیشنهادی را بدین ترتیب مطرح نماییم که در ابتدا، یک کودک هیچ ذهنیتی نسبت به دنیای اطراف خود ندارد و تلاش می‌کند تمام اقدامات و حرکاتی را که می‌تواند، انجام دهد. این مهم کاملاً مشابه با آنچه است که برای یک عامل در یادگیری تقویتی برای بهینگی خط مشی به‌دست آمده‌اش لازم است. ما همچنین مشاهده می‌کنیم که یک کودک بسیار مشتاق است تا از طریق تجربی به ماهیت اجسام پی ببرد، مانند مزه‌کردن هر جسم و به تبع آن نتیجه مثبت و منفی عمل خود را تجربه نماید. باز هم به‌طور مشابه برای عامل در یادگیری تقویتی در نتیجه اقداماتش، پاداش‌های مثبت یا منفی را دریافت می‌کند و حالات (اشیا) متفاوت را از طریق پاداشی که از آنها کسب کرده، متمایز می‌کند. عامل برای ماکزیمم‌کردن مقدار بازگشتی[3] از محیط باید تأثیرات هر اقدام ممکن در هر حالت را امتحان کند. این یکی از پیش‌شرط‌های بهینگی در همه روش‌های یادگیری تقویتی است. به تدریج مفاهیمی از اشیای متفاوت برای کودک شکل می‌گیرد که وی را قادر می‌سازد تأثیرات اقداماتش را پیش‌بینی کند. کودک رفته رفته قادر است الگوها و اجسام را از هم دیگر تفکیک کند و وی دیگر راغب نخواهد بود که تجارب جدیدی را در برخورد با یک دیوار متفاوت کسب کند.

انسان قادر است ویژگی‌های اصلی و متمایزکننده از اشیا را استخراج و سپس الگوهای مشابه را به درستی دسته‌بندی کند. این تجرید در یادگیری وی را قادر می‌سازد که توانایی تشخیص و تصمیم‌گیری دقیق و بالایی داشته باشد. طبق نظر محققان علوم شناختی، دانش انسان دارای ساختار سلسله مراتبی است و داده‌ها در روال‌هایی با چندین مرحله پردازش می‌شوند [۱۵]. ساختارهای سلسله مراتبی برای تشخیص اشیا و استفاده از نشانه‌ها[4] برای مسیریابی[5] در انسان و حیوانات، روش‌های دقیق و مقاومی مقاومی در مقایسه با سیستم‌های مصنوعی هستند [۱۶]. مزایای این سیستم‌ها برای سیستم‌های هوش مصنوعی این انگیزه را ایجاد می‌کند که از آنها در فرایند تشخیص گذرگاه‌ها[6] استفاده شود. در این مقاله نیز سعی شده است که با الهام از سیستم‌های بیولوژیکی و رفتار حیوانات، با در نظر گرفتن محدودیت‌های عامل و تعاریف ممکن در این حوزه یک شیوه کلی مطرح شود که مزیت‌های این سیستم‌ها را تا حد امکان داشته باشد.

1. Primitive Actions
2. Dynamic Programming
3. Returned Value
4. Landmarks
5. Navigation
6. Bottlenecks




شکل ۱: انتقال حالت در RL.

پیداکردن گذرگاه‌ها شبیه پیداکردن درب خروج یک اتاق است اما بین توانایی‌های عامل مصنوعی و کودک تفاوت بسیار زیادی در این مورد وجود دارد. کودک به سادگی می‌تواند یک دیوار و مانع را از فضای باز بدون نزدیک‌شدن به آنها تفکیک کند. می‌توان محدوده دید، برد دید و تجارب قبلی را به‌عنوان مهم‌ترین ویژگی‌ها برای پیداکردن و خروج از یک گذرگاه برای انسان‌ها در نظر گرفت. عموماً عامل‌ها در چارچوب MDP و SMDP دارای هیچ محدوده و برد دیدی نیستند یا این که در مقابل وسعت محیط، بسیار جزئی‌اند. در فرایندهای شبیه‌سازی عموماً عامل فاقد این توانایی‌ها است.

گروه‌بندی اشیا مبتنی بر یک سری از ویژگی‌ها، شیوه متداولی است. در انتقال‌های حالت که در RL انجام می‌گیرد تنها چهار ویژگی برای یک عامل وجود دارد: حالت فعلی $s_t$، اقدام انجام‌شده $a_t$، پاداش $r_t$ و حالت بعدی $s_{t+1}$ که در شکل ۱ نمایش داده شده است. این مورد می‌تواند به این شکل دیده شود که حالات به‌عنوان اشیا به حساب بیایند و نتیجه هر اقدام بر روی هر حالت - حالات بعدی - به‌عنوان ویژگی‌ها در نظر گرفته شوند. به این ترتیب مانع یا دیوار، اشیایی هستند که اگر عامل به سمت آنها برود حالت بعدی و فعلی‌اش یکسان خواهد بود و در محیط‌های نویزی و تصادفی می‌توان گفت که در اکثر مواقع این رویداد رخ می‌دهد.

استفاده از نشانه‌ها برای تشخیص گذرگاه‌ها و شیوه استخراج گذرگاه‌ها را می‌توان از نوآوری‌های روش پیشنهادشده در این حوزه قلمداد کرد. در این مقاله، نشانه به‌عنوان یک ویژگی ادراکی برجسته در نظر گرفته شده است و به عبارت دیگر بعضی از حالات در فضای حالت، نشانه‌هایی برای یافتن گذرگاه‌ها هستند. در این مقاله آنها به وسیله یک مکانیزم تشخیص اشیا به‌صورت سلسله مراتبی از عناصر محیطی که به ندرت تغییر می‌کنند، استخراج می‌شوند.

عامل، ویژگی‌های سطح بالایی را مبتنی بر ترکیب هوشمندانه‌ای از تجارب و مشاهدات سطح پایین می‌تواند استخراج کند. از یک شیوه خوشه‌بندی، مبتنی بر اقدامات عامل برای تشخیص اشیایی مانند دیوارها و گوشه‌ها استفاده شده است. این اشیا در یک شیوه سلسله مراتبی برای شکل‌دادن اشیای سطح بالاتر بررسی و ترکیب می‌شوند.

## ۳- کارهای مشابه

روش ارائه‌شده در [۱۷]، بعضی از مفاهیم تقریباً مشابه با آنچه در این مقاله به‌کار گرفته شده را داراست مانند خطوط بحرانی[1]. خطوط بحرانی در این مقاله متناظر با گذرگاه‌ها یا حالاتی است که در بین نشانه‌ها قرار گرفته‌اند. اما این تفاوت وجود دارد که او از یک سیستم سنسوری (شامل چندین سنسور) برای ربات استفاده کرده است که به عامل این توانایی را می‌دهد که برد و محدوده دید داشته باشد و با ترکیب اطلاعات سنسوری از هر کدام از اجزا می‌تواند یک نگاشت از دنیای اطراف خودش را مهیا کند. در حالی که در روش پیشنهادی نحوه به‌دست آوردن گذرگاه‌ها کاملاً متفاوت است.

روش‌های [۱۸] تا [۲۰] ساخت به سلسله مراتبی از تجریدها را در مسایلی با فضاهای حالت فاکتورشده[2] مطرح کرده‌اند. در [۲۱] تحت مفهومی به‌عنوان skill، تجرید زمانی را با پیداکردن زیر خط مشی‌هایی که به‌صورت مکرر در راه حل‌هایی برای یک مجموعه از کارها روی می‌دهند، به‌دست آورده‌اند.

زیراهداف [۱۳] حالاتی هستند که به‌صورت مکرر دیده می‌شوند یا آنهایی که دارای یک گرادیان پاداش بالا هستند. در محیط‌هایی که پاداش‌ها با تأخیر همراه هستند این روش ممکن است نتواند کارایی خوبی را به نمایش بگذارد. روش [۹] به‌عنوان زیراهداف، نواحی‌ای از فضای حالت در نظر می‌گیرد که عامل آنها را به‌صورت مکرر بر روی خط سیر موفقیت و نه بر روی عدم موفقیت ملاقات می‌کند. این شیوه به تعداد زیادی گام نیاز دارد تا بتواند زیراهداف را استخراج کند.

در شیوه‌هایی مانند [۷]، [۸]، [۱۰] و [۱۱] گراف تراکنش حالات MDP را به‌عنوان یک گراف کامل در نظر می‌گیرند. در [۱۱] با استفاده از الگوریتم max - flow/min - cut روی این گراف، زیراهداف را به‌عنوان حالات کرانه‌ای از نواحی‌ای که در گراف انتقال MDP به‌صورت داخلی قویاً به هم متصل هستند، پیدا می‌کند. روش ارائه‌شده در [۸] با استفاده از یک شیوه خوشه‌بندی مبتنی بر چگالی، خوشه‌های حالات را پیدا کرده و اقدامات گسترش‌یافته زمانی را تعریف می‌کند که به‌عنوان یک زیر خط مشی این اجازه را به عامل می‌دهند که به‌صورت کارایی از یک خوشه به دیگری انتقال پیدا کند. در این مقاله از دو مکانیزم خوشه‌بندی متفاوت استفاده شده است که یکی تنها توپولوژی را به کار می‌گیرد و دیگری از ساختار پاداش از مسایل، اضافه بر توپولوژی استفاده می‌کند. زمانی که حالت جدیدی برای T گام زمانی مشاهده نشود، روال خوشه‌بندی فوق فعال می‌شود و تخمین‌های کلّی از گذرگاه‌ها می‌زند و به تدریج دقت خود را بالا می‌برد. بر طبق [۸] شیوه‌های مبتنی بر خوشه‌بندی توانایی تقویت یادگیری را دارند اگر فضای حالت بتواند به خوبی تفکیک شود. در شیوه مطرح‌شده در [۱۰] که مبتنی بر تئوری گراف طیفی[3] است، با استفاده از مقایسه در میان یال‌های متصل‌کننده، دو رأس متصل روش ارائه‌شده در [۱۲] را ارتقا داده است.

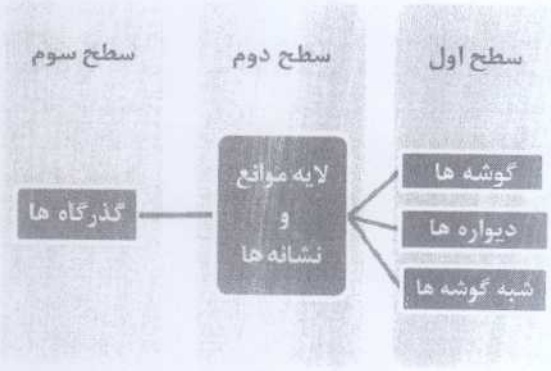
اکثر شیوهای مبتنی بر تقسیم‌بندی گراف دارای پیچیدگی زمانی $O(n^2)$ هستند که $n$ تعداد رأس‌های گراف یا حالت‌ها است. پیچیدگی زمانی روش ارائه‌شده در [۷]، $O(m)$ است در جایی که $m$ تعداد لبه‌ها در گراف است. در این شیوه با الهام از ایده‌های مبتنی بر پیداکردن لبه‌ها در پردازش تصویر، سعی شده است تا گذرگاه‌ها استخراج شوند. برای اعمال این الگوریتم در این شیوه پیش‌فرض‌هایی باید رعایت شود که عملکرد این شیوه را برای محیط‌هایی که در آن اقدامات تصادفی هستند، دشوار می‌سازد. تصادفی عمل کردن در حذف یال‌ها گاهی اوقات باعث نیافتن یک گذرگاه خاص می‌شود و همچنین نوع گذرگاه‌هایی که می‌تواند تشخیص دهد محدود است. در روش‌های [۷]، [۱۰] و [۱۲] راهکاری برای تعمیم الگوریتم برای محیط‌هایی که گراف‌های انتقال حالت جهت‌دار و نامتقارن است ارائه نشده است.

در اکثر روش‌های ارائه‌شده در این زمینه نقاط ضعف مشابهی یافت می‌شود مانند نیاز به کمک طراح برای آنالیز نتایج [۱۱] و [۱۲]، نیاز به ذخیره و پردازش دنباله‌های اقدام و حالت [۱۲] و [۹] و همچنین شرایط محدوده‌کننده‌ای مانند این که خوشه‌های حالت‌های محیط باید دارای اندازه‌های مشابهی باشند. این شرایط نقش تعیین‌کننده‌ای در کیفیت نتایج

1. Critical Lines

2. Factored MDP

3. Spectral Graph Theory




(ب)

۱) تشخیص اشیا به صورت سلسله مراتبی

الف) محاسبه اقدامات موثر برای هر حالت.

ب) خوشه بندی حالت‌ها بر اساس اقدامات موثر.

ج) استخراج خوشه‌های حالت‌های گوشه، دیوار و شبه گوشه‌ها.

۲) استخراج نشانه‌ها

ترکیب اشیا سطح پایین،پیداکردن لایه‌های مانع ونشانه‌ها.

۳) پیدا کردن گذرگاه‌ها

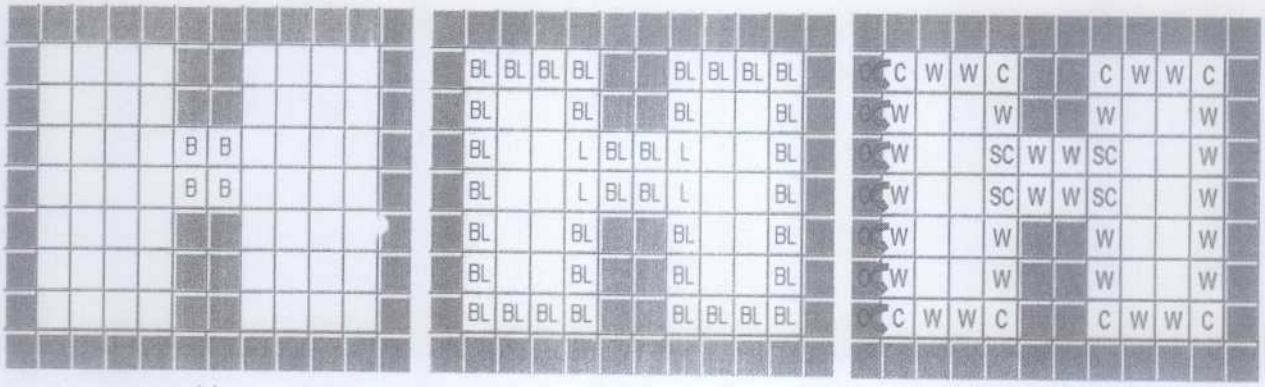
استخراج گذرگاه‌ها بر اساس جستجوی محلی.

٤) شکل گیری optionها

ایجاد اقدامات گسترش یافته زمانی براساس گذرگاه‌های استخراج شده.

(الف)

شکل ۲: (الف) مراحل الگوریتم پیشنهادی و (ب) سلسله مراتب ترکیب اشیا و عناصری که در هر لایه مورد استفاده قرار می‌گیرند.




(ج) (ب) (الف)

شکل ۳: (الف) یک دنیای ماز که گوشه‌ها با 'C'، حالت‌های دیوار با 'W'، شبه‌گوشه‌ها با 'SC' و حالت‌های مجاور با دیوار چپ با 'O' علامت‌گذاری شده‌اند. بردارها نتیجه اقدام چپ را برای عامل در حالات مجاور با این دیوار نشان می‌دهد که حالت فعلی با حالت بعدی یکسان خواهد بود. (ب) حالت‌های مانع که با 'BL' و نشانه‌ها با 'L' علامت‌گذاری شده‌اند. در نهایت با اعمال تشخیص اشیا و ترکیب آنها در آخرین سطح از سلسله مراتب، ماهیت حالت‌ها مطابق علامت‌گذاری خواهد بود. (ج) در آخرین سطح با استفاده از نشانه‌ها، حالت‌های گذرگاه استخراج می‌شوند که این حالت‌ها با علامت 'B' نشان داده شده‌اند.

به‌دست آمده از این روش‌ها دارند. نیاز به دانش قبلی در مورد تعداد خوشه‌ها [۸] و [۱۰] یا تخمین‌هایی از فاصله سایر حالت‌ها به حالت هدف [۲۲]، از ابتدا معمولاً مهیا یا به آسانی قابل به‌دست آمدن نیستند. برای مثال تعیین‌کردن مقدار $k$، تعداد برش‌ها، در روش پیشنهادی [۱۰] دشوار است. در روش‌هایی مانند [۱۲] و [۱۴] زمانی که یک مجموعه حالت گذرگاه را تشکیل می‌دهند کارایی و صحت تشخیص آنها به شکل قابل ملاحظه‌ای کاهش پیدا می‌کند.

## ٤- توصیف الگوریتم پیشنهادی

در [۳] و [۴] در واقع به چارچوب SMDP و نقشی که اقدامات گسترش‌یافته زمانی بر اساس چارچوب Option باید داشته باشند و این که این اقدامات می‌توانند باعث افزایش سرعت یادگیری شوند پرداخته شده است. در [۴] به این که چگونه اقدامات گسترش‌یافته زمانی می‌توانند استخراج شوند پرداخته نشده است و این اقدامات از قبل توسط طراح به عامل داده شده‌اند. اما در این نوشتار به بررسی این چالش پرداخته شده است که این اقدامات گسترش‌یافته زمانی را استخراج کند.

ما در این مقاله الگوریتمی ارائه داده‌ایم که می‌تواند به‌صورت اتوماتیک Optionها را از طریق استخراج گذرگاه‌ها در RL ایجاد کند. تشخیص گذرگاه‌ها و تعریف Optionها با تولید یک نقشه از محیط و نشانه‌ها به‌دست می‌آید. برای این هدف و برای بهتر قابل فهم شدن روش پیشنهادی، از ترم‌های ساده‌ای مانند حالات دیوار و گوشه استفاده کرده‌ایم. خلاصه‌ای از روند عملیات در شکل ۲- الف نشان داده شده است.

عامل می‌خواهد گذرگاه‌ها را از طریق تشخیص اشیا و استخراج نشانه‌ها پیدا کند. عامل از شیوه‌های خوشه‌بندی برای روال تشخیص اشیا به‌صورت سلسله مراتبی استفاده می‌کند که این روال مبتنی بر ویژگی‌های حالت‌ها است. به عبارت دیگر، از محدودیت‌های حرکتی در حالات مجاور با دیوارها و موانع استفاده می‌شود. از آنجایی که برای یک عامل، دیوار و موانع هیچ نگاشت مستقیمی در فضای حالت ندارند، عامل از حالات مجاور با دیوارها و موانع، به‌عنوان حالت‌های متناظر با آنها در تشخیص اشیا استفاده می‌کند. ما در اینجا حالت‌هایی که عامل نمی‌تواند در آنها با استفاده از بعضی اقداماتش حالت فعلی خود را تغییر دهد به‌عنوان لایه موانع[1] تعریف می‌کنیم (شکل ۳- ب).

عامل، اشیای سطح بالاتر را از اشیای سطح پایین‌تر مبتنی بر سلسله مراتب تجرید به‌دست می‌آورد. بنابراین شرایط سخت‌تری در لایه‌های پایین‌تر از سلسله مراتب برای پیداکردن بخش‌هایی از اشیا وجود دارد و هرچه سطوح بالاتر می‌روند، این شرایط ساده‌تر می‌شوند چرا که لایه‌های پردازشی بالاتر ورودی‌های شایسته‌تری را بررسی می‌کنند و نگاه کلی‌تری دارند. بنابراین شرایط و ویژگی‌ها در هر لایه از سلسله مراتب با یکدیگر متفاوت هستند. شمای کلی از کار در شکل ۲- ب ارائه شده است.

### ٤-۱ اجزای الگوریتم

این الگوریتم دارای ۴ قسمت است که به‌صورت زیر توصیف شده‌اند:

1. Barrier Layer

### ٤-١-١ تشخیص اشیا به‌صورت سلسله مراتبی

عامل حالات را مبتنی بر ویژگی‌ها دسته‌بندی می‌کند. اگر اقدامات عامل تصادفی هستند، عامل می‌تواند از استراتژی رأی اکثریت برای پیداکردن اقدامات حاکم اصلی در هر حالت استفاده کند. در ابتدا هیچ خوشه‌ای وجود ندارد و حالت اول، یک خوشه جداگانه را شکل می‌دهد. حالات با خوشه‌های موجود بر اساس شرایطی که در ادامه توضیح داده شده‌اند چک می‌شود. اگر این شرایط بین حالت و خوشه‌ای برقرار باشد، این حالت به آن خوشه اضافه و جستجو متوقف می‌شود، در غیر این صورت یک خوشه جدید شکل خواهد گرفت.

دو شرط فوق به شکل یک سلسله مراتب تعریف شده‌اند. اگر اولین شرط رد بشود، این حالت با خوشه دیگری چک می‌شود و در غیر این صورت شرط دیگری باید چک شود. در پایین‌ترین سطح از سلسله مراتب، هر کدام از حالات دیوار برای عامل به‌عنوان یک خوشه از حالات تفسیر می‌شود که دارای دو ویژگی هستند. در پایین‌ترین سطح سلسله مراتب، اولین ویژگی برای اعضای یک خوشه این است که اقداماتی که به‌ازای آنها حالت فعلی با حالت نتیجه برابر است، برای اعضای یک خوشه یکسان هستند. شرط دوم در پایین‌ترین سطح عبارت است از این که اگر خوشه‌ای بیشتر از یک عضو داشته باشد، برای هر کدام از اعضا باید حداقل عضو دیگری وجود داشته باشد به‌گونه‌ای که آنها بایستی با یکدیگر مجاور باشند. در سطح دوم، حالت‌هایی که در یک خوشه قرار دارند دارای این خاصیت هستند که به‌ازای هر دو حالتی که با یکدیگر مجاورند، حداقل باید همسایه‌ای از این حالت‌ها وجود داشته باشد که آنها نیز با یکدیگر مجاور باشند. به عبارت دیگر شرط دوم در صورتی برقرار است که اگر یک حالت برای مثال $S^1$ تحت یک اقدام به حالت دیگری مانند $S^2$ برود، همسایه‌های $S^1$ تحت حداقل یک اقدام با $S^2$ مجاور باشند. اگر این شرط برقرار نباشد ما می‌توانیم $S^1$ و $S^2$ را به‌عنوان یک ورودی گذرگاه در نظر بگیریم. این ویژگی گذرگاه را در مواردی مانند مثال میانبر در [۱۲] و مثال تاکسی که در [۶] معرفی شده است، تشخیص می‌دهد. با این شرایط، خوشه‌هایی از حالت‌ها شکل می‌گیرند که با یک دیوار مجاور هستند (البته بدون گوشه‌ها). هر کدام از گوشه‌ها دارای خوشه‌هایی متعلق به خودشان هستند. حالت‌های موانع، ترکیبی از حالات گوشه و دیوارها هستند که با یکدیگر مجاور بوده‌اند. حالت‌هایی به‌عنوان شبه‌گوشه‌ها نیز استخراج می‌شوند که به‌عنوان عناصر متصل‌کننده گوشه‌ها و دیوارها استفاده می‌شوند. این عناصر در شکل ۳- الف نشان داده شده‌اند و به ترتیبی که در شکل ۲- ب آمده است عناصر سطح بالاتری از آنها استخراج شده‌اند.

### ٤-١-٢ استخراج نشانه‌ها

یک حالت شبه‌گوشه ممکن است یک نشانه باشد، در غیر این صورت در سطوح بالاتر برای ترکیب حالت‌های مانع و پیداکردن نشانه‌های دقیق‌تر استفاده می‌شود. این روال نیز تا حدی شبیه فرایند تشخیص اشیا به‌صورت سلسله مراتبی است. با استفاده از شرایط تعریف‌شده، خوشه‌هایی از عناصر حالت‌های مانع، مانند دیوارها و گوشه‌ها استخراج می‌شوند که بعد از این رویداد آنها با یکدیگر ترکیب می‌شوند.

با استفاده از مفهوم گوشه‌ها و شبه‌گوشه‌ها، می‌توانیم اشیا را در سطوح پایین‌تر ترکیب کنیم و آنها را در یک خوشه قرار دهیم. در سطح دوم از سلسله مراتب، شبه‌گوشه‌ها بررسی می‌شوند که آیا آنها نشانه هستند و همچنین بعضی از اشیای سطح پایین‌تر شبیه دیوارها، گوشه‌ها و شبه‌گوشه‌ها برای ترکیب‌شدن با یکدیگر مورد بررسی قرار می‌گیرند. اگر شرایط بین آنها برقرار باشد، آنها با یکدیگر ترکیب شده و اشیای سطح بالاتری را شکل می‌دهند. فرایند ترکیب عموماً از لبه‌های خوشه‌های مربوط به حالت‌های مانع به‌عنوان مهم‌ترین بخش از خوشه‌های اشیا از حالات برای ترکیب و نمایش آنها استفاده می‌کند. برای حرکت‌کردن یا مرتبط‌کردن اشیا در فضا به‌طور ضمنی، ما به نزدیک‌ترین بخش از موانع یا اجسام توجه می‌کنیم. در فضاهای دوبعدی، لبه‌های دیوارها، دو حالت هستند که در انتهای دو سر آن قرار گرفته‌اند (شکل ۳- الف). در فضای سه‌بعدی، گوشه‌های یک صفحه، لبه‌ها هستند. در گام نهایی، لبه‌هایی از اشیای سطح بالاتر به‌عنوان نشانه‌ها شناخته می‌شوند (شکل ۳- ب). در حقیقت ما به دنبال خطوط بحرانی در فضای حالت هستیم. خطوط بحرانی همان گونه که در [۱۷] اشاره شده است متناظر با گذرگاه‌های باریک هستند. به عبارت دیگر اگر تعداد اقداماتی که مورد نیاز است برای رفتن از یک نشانه به دیگری کمتر از یک آستانه باشد، این مقدار آستانه می‌تواند متناسب با اندازه خوشه‌های متصل‌شده باشد. می‌توان حالت‌هایی که بین نشانه‌ها قرار دارند را با در نظر گرفتن شرایطی، به‌عنوان حالت‌های گذرگاه در نظر بگیریم. این شرایط بسیار ساده هستند، مانند این که حجم خوشه‌های متصل‌شده نسبت به اندازه حالت‌های قرارگرفته در خطوط بحرانی مقدار متناسبی باشد.

### ٤-١-٣ پیداکردن گذرگاه‌ها

از آنجایی که ما گذرگاه‌ها را بر اساس نشانه‌ها (لبه‌های حالت‌های موانع) تشخیص می‌دهیم، عامل در واقع خطوط بحرانی را در نظر می‌گیرد. اگر یک نشانه پیدا شد، از آنجایی که عامل دارای محدوده دید نیست و نمی‌تواند رابطه نشانه را با بقیه عناصر به‌صورت درجا در نظر بگیرد، او باید آن نشانه را مورد بررسی قرار دهد که آیا هیچ نشانه یا مانع دیگری وجود دارد به‌گونه‌ای که آنها دارای فاصله نزدیکی در فضای اقدام باشند (شکل ۳- ج).

برای یافتن نشانه‌ها یک برد محدود چک می‌شود که آیا نشانه دیگری وجود دارد یا نه. اگر وجود داشته باشد، حالت‌های بین آنها به‌عنوان گذرگاه شناخته می‌شوند. چندین استراتژی را برای پیداکردن گذرگاه می‌توان در نظر داشت. اول این که اگر هر حالت محیط با یک مختصات $(x, y)$ مشخص بشود، عامل می‌تواند از این مختصات و معیار فاصله‌ای مبتنی بر آن به‌عنوان یک راهنما برای تشخیص نشانه دیگر استفاده کند. راه دیگر استفاده از فاصله حقیقی است که برای اهداف دیگری (تعیین حالت‌هایی که در مجموعه اولیه باید قرار بگیرند) در [۱۲] استفاده می‌شود. اگر فاصله نشانه با مانع یا نشانه دیگری کمتر از مقدار مشخصی باشد، این حالت‌ها یک ورودی از گذرگاه را تشکیل می‌دهند. این که تا چه بردی از نشانه یافت‌شده مورد بررسی قرار بگیرد تنها پارامتر موجود در روش پیشنهادی است.

### ٤-١-٤ شکل‌گیری Optionها

بعد از این که عامل یک گذرگاه را پیدا کند، می‌تواند یک Option را برای این خوشه شکل دهد. در واقع، زمانی که خوشه‌ها و گذرگاه‌ها بین آنها شکل گرفتند، یک Option برای هر مجموعه گذرگاه در هر خوشه از حالات شکل می‌گیرد. بر طبق تعریف Option برای همه حالت‌ها در هر یک از خوشه‌ها، مجموعه ورودی Option، $I$، به مقدار یک مقداردهی می‌شود. شرط خاتمه برای این Option برای هر حالت در این خوشه به مقدار صفر نسبت‌دهی می‌شود و برای حالت‌های دیگر خارج از این خوشه یک مقداردهی می‌شود. خط مشی Option، $\pi$، بر طبق رویه پیشنهادی در [۴] می‌تواند شکل بگیرد. ما زیرمسایلی داریم و باید یک یادگیری

جدید برای هر کدام از آنها انجام بدهیم. برای مثال اقداماتی که باعث رسیدن به گذرگاه می‌شوند یک پاداش مثبت و بقیه اقدامات در حالت عادی یک پاداش صفر دریافت می‌کنند. عامل بدین صورت زیر خط مشی‌هایی را برای خروج از هر خوشه یاد می‌گیرد. در این مقاله همانند سایر مقالاتی که در زمینه استخراج گذرگاه داده شده است، بررسی چارچوب Option به‌صورت مفصل‌تر را به [۳] و [۴] ارجاع می‌دهیم.

### ۴-۲ مزایا

ما ادعا می‌کنیم که روش ارائه‌شده می‌تواند گذرگاه‌ها را با دقت بالایی تشخیص دهد چرا که او نیازمندی‌های اصلی- ویژگی‌های لازم- از امکان گذرگاه‌بودن را تشخیص می‌دهد. بر خلاف روش‌هایی مانند [۸]، [۱۰]، [۱۲] و [۱۴] که دقت آنها به اندازه خوشه‌ها در مقایسه با یکدیگر وابسته است یا فقط توانایی تشخیص گذرگاه‌های خاصی را دارند، روش پیشنهادی گذرگاه‌ها را با دقت بالایی استخراج می‌کند. همچنین هزینه محاسباتی روش ارائه‌شده در مقایسه با سایر روش‌ها از نکات قوت آن به حساب می‌آید.

نتایج ما نشان می‌دهد که نشانه‌های استخراج‌شده توسط روال تشخیص اشیا، نشانه‌های مطمئنی برای تشخیص گذرگاه هستند. همچنین کمتر نیازمند و حساس به تنظیم پارامتر ورودی‌اند، بنابراین به دانش اولیه در مورد خصوصیات محیط یا اطلاعاتی که عامل عموماً هیچ ذهنیتی نسبت به آنها ندارد، کمتر نیازمند هستند و بنابراین خودمختاری عامل بهتر حفظ می‌شود و همچنین پارامتر ورودی آن قابل درک است. گذرگاه‌های استخراج‌شده دارای قطعیت هستند به این معنی که در همه اجراها، حالت‌های مشخصی به‌عنوان گذرگاه شناخته می‌شوند. زمانی که عامل به دنبال گذرگاه تنها روی مسیرهای موفقیت‌آمیز است (زیراهداف)، ممکن است که گذرگاه‌های دیگری نادیده گرفته شوند. بدون شک همه گذرگاه‌ها روی مسیرهای موفقیت‌آمیز نیستند. با تشخیص ویژگی‌های اصلی برای وجود گذرگاه می‌توان از آنهایی که چک شده‌اند، فارغ از هدف جاری استفاده کرد. آنها برای کاراترکردن اکتشاف در سایر بخش‌های فضای حالت و یا برای تعمیم دانش فعلی برای محیط‌های دیگر مشابه (که تنها تابع پاداش آنها متفاوت است)، می‌توانند مفید باشند. عامل خواهد توانست با تلاش بسیار کمتر روی کارهای متفاوت نتیجه بگیرد. بعضی از مزیت‌های دیگر استفاده از گذرگاه در [۱۲] شرح داده شده است.

### ۴-۳ پیچیدگی الگوریتم

روش ارائه‌شده نیازی به ذخیره یا پردازش اطلاعات بر روی مسیرها مانند عملیات‌های پیش‌پردازشی شبیه حذف حلقه‌ها ندارد. پیچیدگی زمانی الگوریتم برای یک دنیای توری[1] با $n$ حالت برابر با $O(n^2)$ است. فرض کنید فضای حالت داری $k$ خوشه باشد. حالت‌ها به تدریج قرار است در خوشه‌ها قرار بگیرند و خوشه‌ها نیز به تدریج شکل می‌گیرند. برای این که یک عنصر در یک خوشه قرار بگیرد باید بررسی شود که آیا حالت مورد بررسی با حالتی در این خوشه از طریق اقدامی قابل دسترسی است یا نه. بنابراین ارتباط هر عنصر با عناصر فعلی موجود در خوشه‌هایی که تاکنون شکل گرفته‌اند باید بررسی شود. از آنجایی که ارتباط هر عنصر باید با عناصر یک خوشه چک شود و خوشه‌ها به تدریج شکل می‌گیرند پس تعداد مقایسه‌ها برابر است با

$$1+2+3+\ldots+(n-1)=O(n^2) \qquad (2)$$

1. Grid

## ۵- نتایج آزمایش

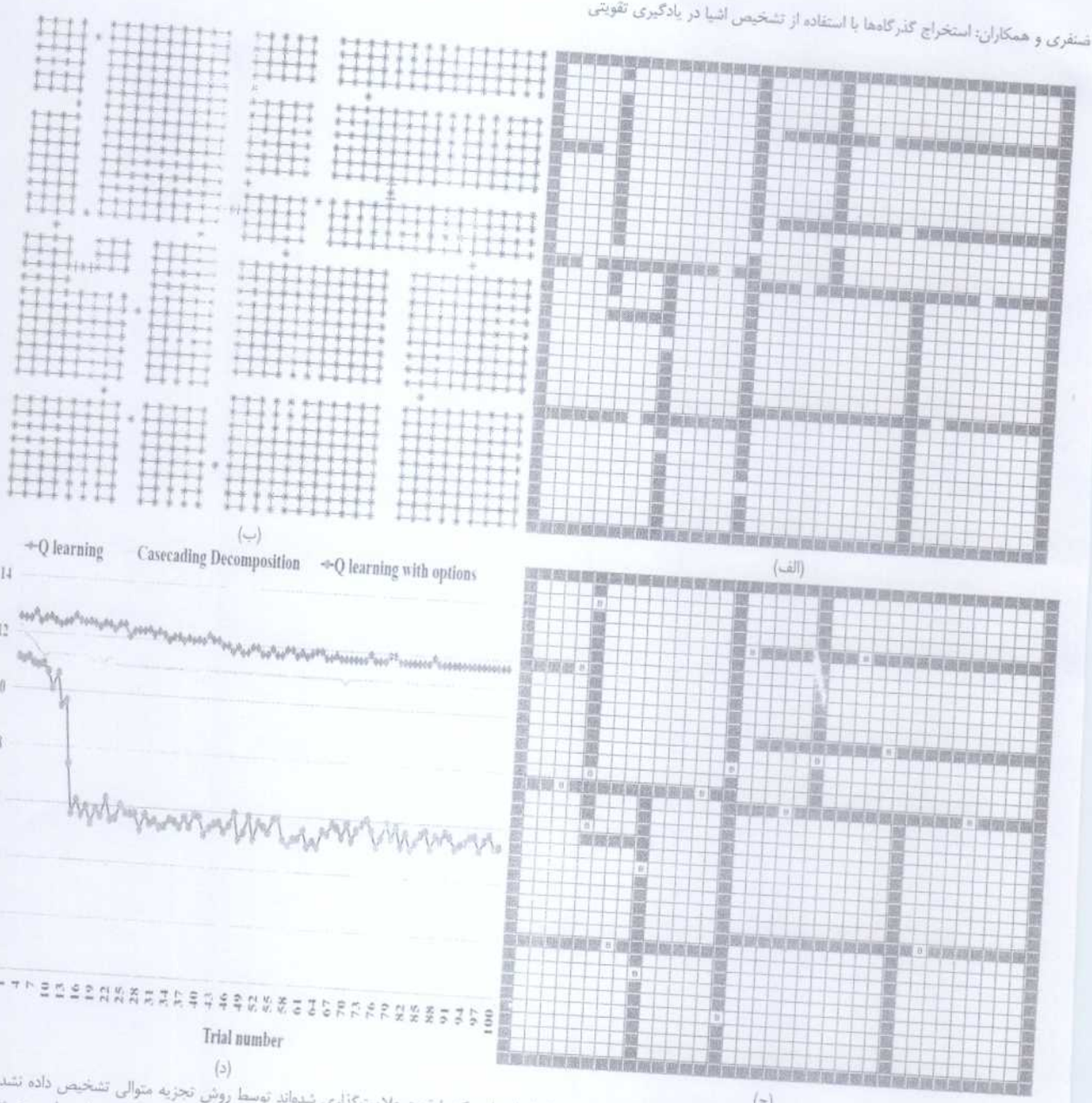
از آنجایی که ایده پیشنهادی مقاله مربوط به تئوری الگوریتم‌های RL است، بنابراین Data Set خاصی وجود ندارد و از بسترهای استاندارد آزمایشی متداولی همانند [۷] تا [۱۴] و [۲۲] که در این زمینه داده می‌شوند استفاده می‌کنیم.

در این بخش ما نتایج الگوریتم پیشنهادی را بر روی یک محیط آزمایش ارائه می‌کنیم و آنها را با روش‌های تجزیه متوالی که در [۱۰] ارائه شده است، مقایسه می‌کنیم. روش پیشنهادی بر اساس کل گراف تراکنش‌های حالت و اقدام کار می‌کند (حالت غیر هم‌زمان). پارامتر ارزیابی که در این مقاله مشابه مقالات دیگر در این حوزه مورد بررسی قرار می‌گیرد عبارت است از سرعت یادگیری (نمودار تعداد گام‌های مورد نیاز برای رسیدن به هدف در تعداد trial). در این نوشتار نیز با استفاده از این معیار به مقایسه روش پیشنهادی با سایر روش‌های دیگر پرداخته شده است که در نمودارهای شکل ۴ مشاهده می‌شود. همچنین روش‌های مورد نظر از منظر نیاز به طراح برای مقداردهی پارامترهای مورد نیاز برای استخراج مورد ارزیابی قرار می‌گیرند که در بخش کارهای مشابه به آن پرداخته شده است. تقریباً در تمامی مقالاتی که در این زمینه داده شده است معیارهای ارزیابی که مورد بررسی قرار می‌گیرند عبارتند از سرعت یادگیری و پارامترهای مورد نیاز که توسط روش مورد استفاده قرار می‌گیرند و در این نوشتار نیز از این دو معیار استفاده شده است.

فرض می‌کنیم نتیجه حرکت عامل با احتمال ۰/۹ مطابق با اقدامی است که انجام داده است و در غیر این صورت، به شکل تصادفی با احتمال ۰/۱ در یکی از جهت‌های دیگر حرکت می‌کند. حالت‌های شروع و هدف به‌صورت تصادفی در هر اجرا انتخاب می‌شوند. پاداش ۰ برای هر اقدام در نظر گرفته می‌شود مگر اقداماتی که عامل را به هدف برسانند که برای آنها پاداش ۱۰+ قایل می‌شویم. نرخ کاهش با مقدار $\gamma=0.9$ مقداردهی شده است و نرخ یادگیری با مقدار $\alpha=0.1$ ثابت نگه داشته می‌شود. نرخ $\varepsilon$ در خط مشی $\varepsilon-greedy$ (مکانیزم انتخاب اقدام) برابر با ۰/۱ در نظر گرفته می‌شود و مقادیر اولیه Q با صفر مقداردهی می‌شوند. برای هر مقایسه تعداد پنج حالت تصادفی برای شروع و پنج حالت هدف در خوشه‌هایی که گذرگاه آنها تشخیص داده نشده‌اند تعیین می‌شوند. اگر خوشه‌های حالت به درستی تشخیص داده نشوند آنگاه یادگیری تقویتی با استفاده از Optionها ممکن است یادگیری را نه تنها بهبود ندهد بلکه آن را بدتر نیز نماید. به این منظور نقاط هدف برای نشان‌دادن این مهم در این خوشه‌ها انتخاب شده‌اند. برای هر کدام از این جفت‌ها تعداد اجراها برابر ۵ در نظر گرفته می‌شود. در هر دور زمانی که عامل به نقطه هدف برسد، دور جاری خاتمه می‌یابد. در آزمایشاتی که در ادامه به آنها پرداخته می‌شود، فرض می‌کنیم عامل در هر گام تنها چهار اقدام ممکن دارد: بالا، پایین، چپ و راست.

به طور مشابه با چارچوب MDP در هر گام زمانی $t$ عامل در یک حالت که با $a_t$ مشخص می‌شود، قرار دارد و می‌تواند یک اقدام $a_t$ را از مجموعه اقدامات ممکن خود انتخاب کند. در برایند این اقدام، یک پاداش $r_{t+1}$ از محیط او دریافت می‌کند و حالت محیط به $s_{t+1}$ تغییر می‌کند. اگر مانعی در جهت اقدامی که عامل آن را انتخاب کرده است وجود داشته باشد، مکان عامل تغییر نمی‌کند.

از یک ماز ساده که در شکل ۴- الف آمده است، به‌عنوان محیط آزمایش استفاده شده است. این شکل با اعمال تغییراتی از محیط آزمایشی در [۸]، برگرفته شده است. در این آزمایش روش ارائه‌شده در [۱۰] نمی‌تواند همه گذرگاه‌ها را تشخیص دهد (شکل ۴- ب). روش ارائه‌شده




شکل ۴: (الف) یک دنیای Maze ساده که دارای ۱۰۹۰ حالت و ۱۸ اتاق است. (ب) بال‌هایی که با قرمز علامت‌گذاری شده‌اند توسط روش تجزیه متوالی تشخیص داده نشده‌اند. (ج) گذرگاه‌ها به درستی توسط روش پیشنهادی تشخیص داده شده‌اند. (د) برای ارزیابی سرعت یادگیری تعداد ۵ حالت شروع و هدف به‌صورت تصادفی انتخاب شده‌اند. برای بهتر نشان داده شدن تفاوت‌ها از لگاریتم مقادیر استفاده شده است (مقایسه‌ای بین روش‌هایی که گذرگاه را به درستی تشخیص داده‌اند مانند روش پیشنهادی، یادگیری Q و روش تجزیه متوالی).

در این مقاله و در [۱۰] نمی‌تواند همه گذرگاه‌ها را تشخیص دهد (شکل ۴- ب). روش ارائه‌شده در این مقاله و [۷] همه گذرگاه‌ها را به درستی تشخیص می‌دهند (شکل ۴- ج). در شکل ۴- د برای نشان‌دادن تأثیر مناسب تشخیص درست همه گذرگاه‌ها، مقایسه‌ای بین روش ارائه‌شده یا [۷] که توانسته‌اند گذرگاه‌ها را به درستی تشخیص دهند و یادگیری $Q$ و تجزیه متوالی که بعضی از گذرگاه‌ها را تشخیص نمی‌دهد، ارائه شده است.

## ۶- نتیجه‌گیری

نتایج ما نشان می‌دهد که دقت شیوه پیشنهادشده در محیط‌های متفاوت قابل قبول است. این دقت، در همگرایی به خط مشی بهینه به شدت تأثیرگذار است. همچنین پیچیدگی زمانی روش پیشنهادی $O(n^2)$ است که بیانگر آن است که روش پیشنهادی دارای پیچیدگی زمانی کمتری نسبت به روش‌های مبتنی بر تئوری گراف طیفی یا تجزیه متوالی است و همچنین نیازی به ذخیره مسیرها ندارد.

تصور کنید که در یک سبد ۲ مهره قرمز و ۱۰ مهره آبی وجود دارد و ما می‌خواهیم مهره‌ها را تنها از همدیگر تفکیک کنیم. ما ۲ مهره قرمز را پیدا می‌کنیم و آنها را بر می‌داریم تا این که بخواهیم ۱۰ مهره آبی را برداریم و در واقع با انتخاب شیء اقلیت، تفکیک هوشمندانه‌ای را انجام می‌دهیم. از آنجایی که عموماً در محیط‌هایی که عامل‌ها با آنها سر و کار دارند، فضای حالت دارای تعداد بسیار زیادی حالت است، ما معتقد هستیم استفاده از نشانه‌ها برای پیداکردن گذرگاه کاری معقول و با توجیه بیولوژیک است.

## مراجع

[1] L. Kaelbling, M. Littman, and A. Moore, "Reinforcement learning: a survey," *J. of Artificial Intelligence Research*, vol. 4, pp. 237-285, 1996.

[2] M. Ghavamzadeh, S. Mahadevan, and R. Makar, "Hierarchical multi-agent reinforcement learning," *Autonomous Agents and Multi-Agent Systems*, vol. 13, no. 2, pp. 197-229, Sep. 2006.

[3] A. Barto and S. Mahadevan, "Recent advances in hierarchical reinforcement learning markov and semi-markov decision processes," *Discrete Event Dynamic Systems*, vol. 13, pp. 41-77, 2003.

[4] R. S. Sutton, D. Precup, and S. Singh, "Between MDPs and semi-MDPs: a framework for temporal abstraction in reinforcement learning," *Artificial Intelligence*, vol. 112, no. 1-2, pp. 181-211, Aug. 1999.

[5] R. Parr and S. Russell, "Reinforcement learning with hierarchies of machines," in *Proc. Conf. on Advances in Neural Information Processing Systems*, pp. 1043-1049, 1997.

[6] T. G. Dietterich, "Hierarchical reinforcement learning with the MAXQ value function decomposition," *J. of Artificial Intelligence Research*, vol. 13, pp. 227-303, 2000.

[7] G. Kheradmandian and M. Rahmati, "Automatic abstraction in reinforcement learning using data mining techniques," *Robotics and Autonomous Systems*, vol. 57, no. 11, pp. 1119-1128, Nov. 2009.

[8] S. Mannor, I. Menache, A. Hoze, and U. Klein, "Dynamic abstraction in reinforcement learning via clustering," in *Proc. 21st Int. Conf. on Machine learning, ICML'04*, p. 560-567, 2004.

[9] E. A. Mcgovern, Autonomous Discovery of Temporal Abstractions from Interaction with an Environment, Citeseer, 2002.

[10] C. Chiu and V. W. Soo, "Automatic complexity reduction in reinforcement learning," *Computational Intelligence*, vol. 26, no. 1, pp. 1-25, Feb. 2010.

[11] I. Menache, S. Mannor, and N. Shimkin, "Q-cut-dynamic discovery of sub-goals in reinforcement learning," in *Proc. of the 13th European Conf. on Machine Learning*. pp. 295-3062002.

[12] O. Simsek, A. P. Wolfe, and A. G. Barto, "Identifying useful subgoals in reinforcement learning by local graph partitioning," in *Proc. of the 22nd Int. Conf. on Machine Learning*, *ICML'05*, pp. 816-823, 2005.

[13] B. Digney, "Learning hierarchical control structures for multiple tasks and changing environments," in: *Proc. of 5th Int. Conf. on Simulation of Adaptive Behavior: From Animals to Animats 5*, pp. 321-330, 1998.

[14] O. Simsek and A. Barto, "Skill characterization based on betweenness," in *Proc. 22nd Annual Conf. on Advances in Neural Information Processing Systems, NIPS'08*, pp. 1497-1504, 2008.

[15] M. Riesenhuber and T. Poggio, "Hierarchical models of object recognition in cortex," *Nature Neuroscience*, vol. 2, no. 11, pp. 1019-25, Nov. 1999.

[16] T. S. Collett and P. Graham, "Animal navigation: path integration, visual landmarks, and cognitive maps," *Current Biology*, vol. 14, no. 12, pp. 475-457, Jun. 2004.

[17] S. Thrun, "Learning metric - topological maps for indoor mobile robot navigation," *Artificial Intelligence*, vol. 99, no. 1, pp. 21-71, 1998.

[18] N. Mehta, S. Ray, P. Tadepalli, and T. Dietterich, "Automatic discovery and transfer of task hierarchies in reinforcement learning," *AI Magazine*, vol. 32, no. 1, p. 35, 2011.

[19] A. Jonsson, *A Causal Approach to Hierarchical Decomposition in Reinforcement Learning*, Ph. D. Thesis, University of Massachusetts Amherst, Feb. 2006.

[20] B. Hengst, *Discovering Hierarchy in Reinforcement Learning*, Ph. D. Thesis, University of New South Wales, Australia, Dec. 2003.

[21] S. Thrun and A. Schwartz, "Finding structure in reinforcement learning," *Proc. 5th Annual Conf. on Advances in Neural Information Processing Systems, NIPS'95*, pp. 385-392, 1995.

[22] C. C. Chiu, "Subgoal identification for reinforcement learning and planning in multiagent problem solving," in *Proc. of 5th German Conf. on Multiagent System Technologies*, pp. 37-48, 2007.

**بهزاد غضنفری** تحصیلات خود را در مقطع کارشناسی در دانشگاه فردوسی مشهد و کارشناسی ارشد در دانشگاه علم وصنعت ایران به‌ترتیب در سال‌های ۱۳۸۸ و ۱۳۹۰ به پایان رسانده است. زمینه‌های تحقیقاتی ایشان عبارتند از: پردازش نرم، یادگیری ماشین، یادگیری تقویتی، سیستم‌های چند عاملی و الگوریتم‌های تطابق الگو و رشته.

**ناصر مزینی** در سال ۱۳۶۹ مدرک کارشناسی خود را از دانشگاه صنعتی شریف در مهندسی برق گرایش کامپ - سخت افزار اخذ نمود و سپس در سال ۱۳۷۲ مدرک کارشناسی ارشد را در رشته سیستم‌های اطلاعاتی و تله‌ماتیک از سوپلک فرانسه و همچنین در سال ۱۳۷۷ از دانشگاه رن یک فرانسه در رشته انفورماتیک دریافت نمود. وی از سال ۱۳۷۹ در دانشکده مهندسی کامپیوتر دانشگاه علم و صنعت ایران مشغول به فعالیت گردید و اینک نیز عضو هیأت علمی این دانشکده می‌باشد. زمینه‌های علمی مورد علاقه نامبرده عمدتاً در زمینه رایانش نرم، فناوری اطلاعات و شبکه‌های کامپیوتری است.

**محمدرضا جاهد مطلق** در سال ۱۳۵۷ مدرک کارشناسی مهندسی برق خود را از دانشگاه صنعتی شریف دریافت نمود و پس از ۶ سال در فعالیت‌های صنعتی برای ادامه تحصیلات خود به انگلستان رفت و مدارک کارشناسی ارشد و دکتری خود را در سال‌های ۱۳۶۶ و ۱۳۷۰ از دانشگاه برادفورد انگلستان در زمینه مهندسی کنترل اخذ نمود. پس از بازگشت از سال ۱۳۷۰ تاکنون در دانشگاه علم و صنعت ایران مشغول به فعالیت می‌باشد زمینه‌های علمی مورد علاقه نامبرده متنوع بوده و شامل موضوعاتی مانند سیستم‌های پیچیده محاسبات آشوب گونه سیستم‌های هایبرید رباتیک و کنترل سیستم‌های پیچیده می‌باشد.