

# بررسی الگوریتمهای مبتنی بر اتوماتانهای یادگیر برای حل مسائل ارضای محدودیت

محمد رضا میبیدی  
مجید نمازی  
آزمایشگاه محاسبات نرم  
دانشکده مهندسی کامپیوتر  
دانشگاه صنعتی امیرکبیر  
تهران، ایران.

## چکیده

بسیاری از مسائل مطرح در زمینه هوش مصنوعی را می‌توان به صورت مسائل ارضای محدودیت توصیف نمود. این مسائل با استفاده از مجموعه‌ای از متغیرها و تعدادی محدودیت بر روی مقادیری که این متغیرها می‌توانند اختیار کنند، تعریف می‌گردند. حل این مسائل مجموعه‌ای از مقادیر منحصر به فرد برای متغیرهاست بطوریکه تمامی محدودیت‌های مورد نظر مساله ارضا شده باشند. ابزار و الگوریتم‌های مختلفی برای حل اینگونه از مسائل ارائه شده‌اند که از جمله آنها می‌توان به الگوریتم‌های مختلف عقبگرد، جلورو یا آینده‌نگر، یادگیر وضعیت‌های نامناسب، الگوریتم‌های مبتنی بر شبکه‌های عصبی و الگوریتم‌های مبتنی بر اتوماتانهای یادگیر اشاره کرد. ما در این مقاله به معرفی این الگوریتم‌ها به استثنای الگوریتم‌های مبتنی بر شبکه‌های عصبی پرداخته و الگوریتم‌های مبتنی بر اتوماتانهای یادگیر را مورد نقد و بررسی قرار خواهیم داد.

**کلمات کلیدی:** هوش مصنوعی، مسائل ارضای محدودیت، اتوماتانهای یادگیر.

## ۱- مقدمه

مسائل ارضای محدودیت<sup>۱</sup> که به آنها مسائل برچسب‌دهی سازگار<sup>۲</sup> نیز گفته می‌شود، بخش وسیعی از مسائل مطرح در زمینه هوش مصنوعی<sup>۳</sup> را در بر می‌گیرند. مسائل هشت وزیر<sup>۴</sup>، رنگ کردن نقشه<sup>۵</sup>، تناظر گرافها<sup>۶</sup> و درک تصویر با برچسب‌دهی والتز<sup>۷</sup> همگی مثالهایی از اینگونه از مسائل هستند [1,2,3,6,7].

اجزاء اصلی مسائل ارضای محدودیت، متغیرها و محدودیتها هستند. هدف نهایی در حل اینگونه از مسائل مقداردهی به متغیرهاست به نحوی که مجموعه محدودیت‌های مساله ارضاء شوند. مقدار هر متغیر از دامنه مقادیر مجاز آن انتخاب می‌گردد. دامنه مقادیر مجاز یک متغیر، توسط محدودیت‌های یکتایی<sup>۸</sup> تعیین می‌گردد و شرایط سازگاری مقادیر متغیرهای مختلف نسبت به یکدیگر نیز توسط محدودیت‌های چندتایی<sup>۹</sup> مشخص می‌گردند [7]. البته مسائلی که ما در این مقاله به آنها خواهیم پرداخت، تنها شامل محدودیت‌های یکتایی و دوتایی هستند که به این نوع از مسائل ارضای محدودیت، مسائل

---

Constraint Satisfaction Problems (CSP)  
Consistent Labeling Problems (CLP)  
Artificial Intelligence (AI)  
8\_Queenes  
Graph Coloring  
Graph Isomorphism  
Waltz Labeling  
Unary Constraints  
N\_ary Constraints

ارضای محدودیت دودویی<sup>۱</sup> گفته می‌شود. گرچه در بخش ۱۱ خواهیم دید که محدودیتهای چندتایی به صورت ضمنی در این نوع از مسائل وجود دارند [4,5]. ابزار و الگوریتمهای مختلفی برای حل مسائل ارضای محدودیت ارائه شده‌اند. هدف از این نوشتار بررسی الگوریتم ارائه شده بر مبنای اتوماتانهای یادگیر<sup>۲</sup> در مرجع [1] و مقایسه آن با سایر الگوریتمها می‌باشد. ما در بخش ۲ به معرفی مراحل انجام این مطالعه پرداخته و سپس در بخشهای ۳ الی ۱۱ به معرفی مراحل مهم این مطالعه و نتایج به دست آمده در آنها خواهیم پرداخت و سپس در نهایت نتایج به دست آمده را در بخش ۱۲ تحلیل نموده و نتیجه گیری کلی از بحث ارائه خواهیم کرد.

## ۲- مراحل انجام مطالعه

این مطالعه با هدف آشنایی با مفهوم اتوماتان یادگیر و نحوه استفاده از آن برای حل مسائل ارضای محدودیت و نیز بررسی امکان بهبود الگوریتم ارائه شده در [1] بر مبنای اتوماتان یادگیر (که از این به بعد آن را الگوریتم اتوماتان یادگیر خواهیم نامید) تعریف و انجام شده است. لازم به ذکر است که تمامی مراحل انجام این تحقیق لزوماً در راستای بررسی الگوریتم اتوماتان یادگیر نیستند اما برای تحلیل نهایی این الگوریتم، مطالعه تمامی قسمت‌های این مقاله و تمامی مراحل انجام این مطالعه و نتایج آنها ضروریست.

در این تحقیق ابتدا در قدم اول الگوریتم اتوماتان یادگیر ارائه شده در [1] مطالعه و پیاده‌سازی گردید. در قدم دوم مسائل ارضای محدودیت هشت وزیر و درک تصویر با برچسبدهی والتز، مطالعه شده و به صورت مسائل ارضای محدودیت مورد مدلسازی قرار گرفتند. در قدم سوم الگوریتم اتوماتان یادگیر بر روی مسائل هشت وزیر و درک تصویر با برچسبدهی والتز (با یک تصویر نمونه) اعمال شد. در قدم چهارم بر اساس ایده عدم تغییر انتخابهای اتوماتانهای موفق، یک الگوریتم توزیع شده<sup>۳</sup> برای حل مسائل ارضای محدودیت طراحی و پیاده‌سازی گردید. در قدم پنجم الگوریتم اتوماتان یادگیر و الگوریتم توزیع شده با الگوریتمهای آینده‌نگر<sup>۴</sup> از نظر تعداد قدمها یا تکرارها<sup>۵</sup> مورد مقایسه قرار گرفتند. در قدم ششم مساله هشت وزیر مورد بررسی واقع شده و زمان نیاز به عقبگرد<sup>۶</sup> احتمالی مورد انتظار در آن به صورت تقریبی مورد محاسبه قرار گرفت. در قدم هفتم الگوریتمهای آینده‌نگر از نظر زمان نیاز به عقبگرد در آنها در طی حل مساله هشت وزیر مورد بررسی واقع شدند و نتایج قدم ششم در این قدم تا حدودی تایید شدند. در قدم هشتم با استفاده از نتایج قدمهای ششم و هفتم روشهایی برای تعیین احتمالات اولیه و نیز تغییر پارامترهای ثابت موجود در الگوریتم اتوماتان یادگیر، ارائه و پیاده‌سازی شده و سپس این روشها نسبت به روش استفاده از پارامترهای با مقادیر ثابت در حل مساله هشت وزیر مورد مقایسه قرار گرفتند و در نهایت در قدم نهم الگوریتمهای یادگیر وضعیتهای نامناسب<sup>۷</sup> مطالعه شده و از نظر مفهومی با الگوریتم اتوماتان یادگیر مورد مقایسه قرار گرفتند.

## ۳- الگوریتم اتوماتان یادگیر

الگوریتمهای مختلفی برای حل مسائل ارضای محدودیت ارائه شده‌اند اما الگوریتم ارائه شده بر مبنای اتوماتانهای یادگیر در [1] اولین و تقریباً آخرین کار انجام شده در این زمینه است. در این روش برای هر یک از متغیرهای مساله یک اتوماتان یادگیر در نظر گرفته می‌شود. عملی<sup>۸</sup> که یک اتوماتان انجام می‌دهد، انتخاب یکی از مقادیر موجود در دامنه مقادیر مجاز متغیر متناظر با آن است. در ابتدای حل یک مساله ارضای محدودیت، احتمال انتخاب تمامی مقادیر موجود در دامنه مقادیر یک متغیر با یکدیگر برابر هستند. در هر قدم یا تکرار الگوریتم، هر یک از اتوماتانها برحسب احتمالات متناظر با مقادیر موجود در دامنه مقادیر متغیر متناظر با آن اتوماتان، یکی از آنها را به صورت تصادفی انتخاب می‌کند و این انتخاب را به همسایگان خود اطلاع می‌دهد. هر اتوماتان با بررسی سازگاری انتخاب خود با انتخابهای همسایگان خود، قبول یا رد انتخابهای آنها را به ایشان اعلام می‌کند و سپس هر اتوماتان پس از گرفتن پاسخهای همسایگان خود، در صورت سازگاری

انتخابش با تمامی انتخابهای همسایگان خود و گرفتن جواب قبولی از تمامی همسایگانش، احتمال متناظر با این انتخاب را بر اساس یک پارامتر تشویق<sup>۱</sup> ثابت به نام  $a$ ، افزایش داده و احتمال متناظر با سایر مقادیر موجود در دامنه را بگونه‌ای کاهش می‌دهد که مجموع احتمالات متناظر با تمامی مقادیر برابر یک باقی بماند. در صورتی که یک آتوماتان حداقل از یکی از همسایگان خود جواب رد بگیرد، احتمال متناظر با این انتخاب را بر اساس یک پارامتر تنبیه<sup>۲</sup> ثابت به نام  $b$  کاهش می‌دهد و احتمال متناظر با سایر مقادیر را به نوعی افزایش می‌دهد که مجموع احتمالات متناظر با تمامی مقادیر برابر یک باقی بماند. اگر  $m$  تعداد مقادیر موجود در دامنه مقادیر آتوماتان  $n$ ام باشد و  $n$  شماره قدم فعلی و  $n+1$  شماره قدم بعدی باشد و  $j$  شماره مقدار انتخاب شده در مرحله  $n$ ام توسط آتوماتان  $n$ ام باشد، در این صورت احتمال انتخاب مقدار  $k$ ام از دامنه مقادیر آتوماتان یادگیر  $n$ ام در شروع قدم  $n+1$ ، از روی روابط زیر تعیین می‌گردد:

$$P_{jk}(n+1) = \begin{cases} P_{jk}(n) + a \times (1 - P_{jk}(n)) & k=j \text{ و موفق} \\ (1-a) \times P_{jk}(n) & k \neq j \text{ و موفق} \\ (1-b) \times P_{jk}(n) & k=j \text{ و ناموفق} \\ (b/(m-1)) + (1-b) \times P_{jk}(n) & k \neq j \text{ و ناموفق} \end{cases}$$

#### ۴- مسائل مورد بررسی در این مقاله

ما از بین مسائل مختلف ارضای محدودیت، دو مساله هشت وزیر و درک تصویر با برچسبدهی والتز را برای بررسی الگوریتمهای مختلف انتخاب کرده‌ایم که در بخشهای زیر به معرفی آنها می‌پردازیم.

##### ۴-۱ مساله هشت وزیر

مساله هشت وزیر یکی از مسائل معروف و کلاسیک ارضای محدودیت می‌باشد. در این مساله هر سطر از صفحه شطرنج معادل یک متغیر یا وزیر در نظر گرفته می‌شود و ستونها و یا خانه‌های موجود در هر سطر مقادیر قابل انتخاب برای هر وزیر یا متغیر متناظر با آن سطر در نظر گرفته می‌شوند. هدف از حل این مساله یافتن ستونهایی در هر سطر از صفحه شطرنج برای متغیر متناظر با آن است بگونه‌ای که در صورت جایگذاری وزیرها در خانه‌های یافت شده، هیچ وزیری دیگری را تهدید نکند.

##### ۴-۲ مساله درک تصویر با برچسبدهی والتز

مساله برچسبدهی به یالهای یک تصویر دو بعدی از یک شیء سه بعدی به روش والتز، یکی دیگر از مسائل ارضای محدودیت است. در این مساله برای اشکال دو بعدی که در هر راس آنها حداکثر سه یال همدیگر را قطع نمایند<sup>۳</sup>، برای هر یک از یالها یکی از علائم  $>$ ،  $+$ ،  $-$  و  $<$  انتخاب می‌گردد بطوریکه دو علامت  $>$  و  $<$  به معنی مرز کل یا قسمتی از جسم متناظر سه بعدی با فضای پیرامونی بوده و علامت  $+$  به معنی برآمدگی و علامت  $-$  به معنی فرورفتگی می‌باشد. به عنوان مثال در شکل (۱-الف) یک مکعب نمایش داده شده است که یکی از گوشه‌های آن کمی سائیده یا بریده شده است. در صورت برچسبدهی به یالهای آن، بایستی شکل (۱-ب) نتیجه شود. تحلیلهای انجام گرفته بر روی این مساله نشان می‌دهد که در صورت برچسبدهی به یالها، تنها ۱۸ حالت ممکن برای رئوس با اشکال مختلف وجود خواهد داشت. بنابراین هدف از حل این مساله می‌تواند انتخاب یکی از حالات ۱۸ گانه برای تک تک رئوس باشد بگونه‌ای که یالهای مشترک بین رئوس همجوار علامت مشترکی گرفته باشند [2].

##### ۵- اعمال الگوریتم آتوماتان یادگیر بر روی مسائل ارضای محدودیت

ما از بین مسائل مختلف ارضای محدودیت، دو مساله هشت وزیر و درک تصویر با برچسبدهی والتز بر روی تصویر نشان داده شده در شکل (۱) را برای تست الگوریتم آتوماتان یادگیر مورد استفاده قرار دادیم که در زیر به ارائه نتایج به دست آمده می‌پردازیم.

## ۵-۱ مساله هشت وزیر

ما ابتدا برای مساله هشت وزیر الگوریتم آتوماتان یادگیر را با پارامترهای  $a$  و  $b$  با مقادیر مختلف از ۰ تا ۱ و با طول گامهای ۰.۲۵ و با ۱۵ بار اجرا در هر مورد و تعداد حداکثر تکرار ۲۰۰۰۰ قدم اعمال کردیم که نتیجه این عمل در شکل (۲) نشان داده شده است. همانطور که در شکل (۲) مشاهده می‌شود، بهترین نتایج از لحاظ درصد جوابهای همگرا و میانگین مناسب تعداد تکرارها وقتی به دست می‌آید که مقدار پارامتر  $a$  بزرگ بوده و مقدار پارامتر  $b$  کوچک بوده و نزدیک به صفر باشد. برای وضوح بیشتر، ما بار دیگر الگوریتم آتوماتان یادگیر را بر روی مساله هشت وزیر با پارامترهای  $a$  و  $b$  با مقادیر مختلف به ترتیب از ۰.۲ الی ۱ با طول گامهای ۰.۲ و ۰.۵ الی ۰.۲ با طول گامهای ۰.۵ و با ۲۵ بار اجرا در هر مورد و تعداد حداکثر تکرار ۵۰۰۰ اعمال کردیم که نتیجه این عمل در شکل (۳) نشان داده شده است. نتایج به دست آمده در شکل (۳) بسیار شبیه به نتایج به دست آمده در صفحه ۱۴ از [1] با پارامترهای مشابه است. همانطور که ملاحظه می‌شود، نتایج مناسب وقتی به دست می‌آید که  $a$  نزدیک به ۱ بوده و  $b$  نیز بین ۰.۱ و ۰.۲ باشد.

## ۵-۲ مساله درک تصویر با برچسب‌دهی والتز

ما برای مساله درک تصویر با برچسب‌دهی والتز، تصویر نشان داده شده در شکل (۱) را انتخاب نموده و الگوریتم آتوماتان یادگیر را با پارامترهای  $a$  و  $b$  با مقادیر مختلف از ۰ تا ۱ و با طول گامهای ۰.۲۵ و با ۱۵ بار اجرا در هر مورد و تعداد حداکثر تکرار ۲۰۰۰۰ قدم اعمال کردیم که نتیجه این عمل در شکل (۴) نشان داده شده است. همانطور که در شکل (۴) مشاهده می‌شود، به جز در حالتی که  $b=0$  باشد، در بقیه حالتها با درصدهای متفاوتی الگوریتم همگرا بوده است اما درصدهای همگرایی و نیز میانگین تکرارها خیلی به هم نزدیک هستند. درصدهای بالای همگرایی در گوشه تحتانی سمت راست واقع شده و کمترین مقدار میانگین تکرارها در  $a=1$  و  $b=0.25$  قرار دارد. برای وضوح بیشتر ما سقف تعداد تکرار را به ۵۰۰۰۰ افزایش دادیم و الگوریتم آتوماتان یادگیر را دوباره با همان پارامترهای بالا بر روی این مساله اعمال کردیم که نتیجه این عمل در شکل (۵) نشان داده شده است. همانطور که در شکل (۵) مشاهده می‌شود، به جز در حالتی که  $b=0$  باشد، باز هم در بقیه حالتها با درصدهای متفاوتی الگوریتم همگرا بوده است. اما به دلیل سقف تعداد تکرارهای بالاتر نسبت به شکل (۴)، میانگینها و درصدها بالا رفته‌اند، ولی بازهم درصدهای همگرایی و نیز میانگین تعداد تکرارها خیلی به هم نزدیک هستند. توزیع درصدهای بزرگ این بار بیشتر است و درصدهای بالای ۸۰ را در نقاط مختلف جدول می‌توان دید. گرچه چگالی آنها به سمت گوشه تحتانی سمت راست می‌باشد. بعلاوه کمترین مقدار میانگین تعداد تکرارها بازهم در موقعیت  $a=1$  و  $b=0.25$  واقع شده است.

## ۶- ارائه و پیاده‌سازی الگوریتمی توزیع شده

در الگوریتم آتوماتان یادگیر پس از انتخاب و تشویق یا تنبیه متناظر با مقدار انتخاب شده توسط یک آتوماتان، در قدم یا تکرار بعدی نیز انتخاب مقدار (احتمالا جدید) صورت می‌گیرد. ایده‌ای که به ذهن می‌رسد این است که انتخابهایی که موفق بوده‌اند و به عبارتی مقدارهای انتخابی آتوماتانهایی که با تمامی انتخابهای همسایگان خود سازگار بوده‌اند تغییر داده نشوند. بر اساس همین ایده، الگوریتم توزیع شده جدید طراحی و پیاده‌سازی گردید. ما در زیر به معرفی این الگوریتم می‌پردازیم.

این الگوریتم در هر قدم از اجرای خود دارای هفت مرحله است که ما در زیر به معرفی این مراحل خواهیم پرداخت. در ابتدای شروع این الگوریتم، دامنه مقادیر مجاز متغیرها همان دامنه تعریف شده اولیه است اما در طی اجرای الگوریتم، احتمال دارد که یک مقدار از دامنه یک متغیر برای مدتی و به دلایلی حذف شده و سپس در صورت نیاز به عقبگرد، دوباره به آن اضافه شود. ضمناً وضعیت تمامی متغیرها در شروع اجرای الگوریتم برابر عبارت **ناموفق** می‌باشد.

### مرحله اول: انتخاب یا عقبگرد

در این مرحله هر متغیر دارای وضعیت **ناموفق**، یکی از مقادیر موجود در دامنه مقادیر خود را به صورت تصادفی انتخاب و آن را به همسایگان خود اطلاع می‌دهد. اما در صورتی که دامنه مقادیر آن قبلاً تهی شده باشد، به یکی از

متغیرها که دارای وضعیت موفق باشد، دستور تغییر وضعیت به ناموفق و تغییر مقدار در شروع قدم بعدی صادر می‌کند.

#### مرحله دوم: بررسی سازگاری

در این مرحله هر متغیر دارای وضعیت ناموفق، سازگاری انتخاب خود را با انتخابهای همسایگان ناموفق خود، بررسی می‌کند. اگر با همه آنها سازگار باشد، وضعیت خود را موفق فرض می‌کند. ولی اگر با یکی از مقادیر انتخابی همسایگان ناموفق خود ناسازگار باشد، وضعیت خود را ناموفق فرض می‌کند. هر متغیر در پایان این مرحله، وضعیت نهایی خود را به همسایگان خود اطلاع می‌دهد.

#### مرحله سوم: آینده‌نگری

در این مرحله هر متغیر آخرین وضعیت خود و تمامی همسایگان خود را می‌داند. بنابر این با فرض اینکه متغیرهای اخیراً موفق شده، در قدم بعدی مقدار فعلی خود را تغییر نخواهند داد، مقادیر موجود در دامنه مقادیر خود را که با انتخاب فعلی این نوع از متغیرها ناسازگار باشند، معین نموده و حذف می‌نماید تا در قدم بعدی آن مقادیر را انتخاب ننماید.

#### مرحله چهارم: آینده‌نگری بیشتر

در این مرحله متغیرهای دارای وضعیت ناموفق، سازگاری تمامی مقادیر باقیمانده در دامنه مقادیر خود را با انتخابهای فعلی متغیرهای همسایه ناموفق خود، بررسی می‌کنند. اگر تمامی مقادیر باقیمانده در دامنه مقادیر یک متغیر دارای وضعیت ناموفق با مقدار فعلی یکی از این متغیرهای همسایه ناموفق، ناسازگار باشند، دستور حذف مقدار فعلی از دامنه مقادیر مجاز این همسایه ناموفق، به وی صادر می‌گردد.

#### مرحله پنجم: حذف و بررسی امکان نیاز به عقبگرد

در این مرحله متغیرهای دارای وضعیت ناموفق که از طرف یک یا چند تن از همسایگان خود دستور حذف مقدار فعلی خود را دریافت نموده باشند، مقدار فعلی خود را از دامنه مقادیر خود حذف می‌کنند و سپس تعداد مقادیر باقیمانده در دامنه مقادیر خود را بررسی می‌کنند. اگر این تعداد برابر صفر باشد، نیاز به عقبگرد وجود دارد. بنابر این به یکی از متغیرهای دارای وضعیت موفق، دستور تغییر وضعیت به ناموفق و تغییر مقدار در شروع قدم بعدی صادر می‌گردد.

#### مرحله ششم: حذف و عقبگرد

در این مرحله متغیرهای دارای وضعیت موفق که دستور تغییر وضعیت را (چه در مرحله اول و چه در مرحله پنجم از این قدم) دریافت کرده باشند، مقدار فعلی خود را از دامنه مقادیر خود حذف نموده و وضعیت خود را به ناموفق تغییر داده و این امر را همراه با شماره آخرین قدمی که از آن به بعد ثابت مانده بودند، به سایر متغیرها اطلاع می‌دهند.

#### مرحله هفتم: ترمیم

در این مرحله همه متغیرها حذفیات مقادیر از دامنه مقادیر خود که به دلیل عدم سازگاری با مقادیر قبلی متغیرهای ذکر شده در مرحله ششم حذف شده‌اند و یا در مراحل چهارم و پنجم و یا ششم از قدمهای بین آخرین قدم ثابت ماندن متغیرهای ذکر شده در مرحله ششم تا بحال حذف شده‌اند را بی اثر ساخته و این مقادیر را به دامنه خود باز می‌گردانند.

پس از انجام هفت مرحله فوق در یک قدم، تعدادی از متغیرها در وضعیت موفق باقی می‌مانند و تعدادی نیز دارای وضعیت ناموفق خواهند بود. متغیرهایی که وضعیت آنها موفق باشد، در شروع قدم بعدی انتخاب دیگری را انجام نمی‌دهند و ثابت باقی می‌مانند. اما آنهایی که دارای وضعیت ناموفق باشند، در قدم بعدی انتخاب تصادفی دیگری را انجام خواهند داد.

#### ۷- مقایسه با الگوریتمهای آینده‌نگر

الگوریتمهای بررسی جلورو<sup>۱</sup> (FC)، آینده‌نگر جزئی<sup>۲</sup> (PL)، آینده‌نگر کامل<sup>۳</sup> (FL) و آینده‌نگر بهبود یافته<sup>۴</sup> (MFL) که به الگوریتمهای آینده‌نگر معروفند، الگوریتمهای سیستماتیک هستند که در هر قدم از اجرای خود، برای یکی از متغیرها مقداری را انتخاب نموده و سپس این انتخاب را بر روی دامنه مقادیر متغیرهای مقدار دهی نشده تاثیر می‌دهند و برخی از مقادیر را از دامنه مقادیر متغیرهای مقدار دهی نشده حذف می‌کنند که به این امر، آینده‌نگری گفته می‌شود. اما میزان این آینده‌نگری در این چهار الگوریتم متفاوت می‌باشد بطوریکه در FC کمترین بوده و در MFL بیشترین می‌باشد. هر چقدر میزان آینده‌نگری بیشتر باشد الگوریتم در تعداد قدمهای کمتری به جواب می‌رسد [3,7].

الگوریتم توزیع شده ارائه شده در بخش قبل نیز در مراحل سوم و چهارم خود تا حدودی آینده‌نگری را انجام می‌دهد. بنابر این ما تصمیم گرفتیم که از لحاظ تعداد قدمهای اجرا و تعداد بررسیهای سازگاری، این الگوریتم را با الگوریتمهای آینده‌نگر مقایسه نماییم. بدین منظور متوسط تعداد قدمهای اجرا و تعداد بررسیهای سازگاری در ده بار اجرای این الگوریتم در طی یافتن جوابهایی برای مسائل ۴، ۵، ۶، ۷، ۸ و ۹ وزیر را با چهار الگوریتم آینده‌نگر FC، PL، FL و MFL در یافتن اولین جواب برای این مسائل، با یکدیگر مقایسه کردیم که نتیجه این عمل به ترتیب در دو شکل (۶) و (۷) نشان داده شده است. همانطور که در شکل (۶) ملاحظه می‌شود، الگوریتم توزیع شده از لحاظ تعداد متوسط قدمها در ده بار اجرا با این الگوریتمها قابل مقایسه بوده و حتی در مساله ۸ وزیر بهترین تعداد قدمها را داشته است. بعلاوه همانطور که در شکل (۷) ملاحظه می‌شود، از نظر تعداد بررسیهای سازگاری، بهبود قابل ملاحظه‌ای در الگوریتم توزیع شده نسبت به الگوریتمهای آینده‌نگر به دست آمده است. از طرف دیگر در [1] ادعا شده است که الگوریتم اتوماتان یادگیر از الگوریتم FC بهتر است. بنابرین ما تصمیم گرفتیم که الگوریتم اتوماتان یادگیر را از نظر تعداد قدمهای اجرا و تعداد بررسیهای سازگاری، با الگوریتمهای آینده‌نگر مورد مقایسه قرار دهیم. بدین منظور متوسط تعداد قدمهای اجرا و تعداد بررسیهای سازگاری در ده بار اجرای این الگوریتم در طی یافتن جوابهایی برای مسائل ۴، ۵، ۶، ۷، ۸ و ۹ وزیر را با چهار الگوریتم آینده‌نگر FC، PL، FL و MFL در یافتن اولین جواب برای این مسائل، با یکدیگر مقایسه کردیم که نتیجه این عمل به ترتیب در دو شکل (۸) و (۹) نشان داده شده است. همانطور که در دو شکل (۸) و (۹) ملاحظه می‌شود، تعداد قدمها و تعداد بررسیهای سازگاری الگوریتم اتوماتان یادگیر به طور فاحشی از چهار الگوریتم فوق بیشتر است. البته پارامتر a برابر ۰.۹۳ و پارامتر b برابر ۰.۱۲ در نظر گرفته شده‌اند که با توجه به شکل (۳) مقادیر مناسبی می‌باشند.

#### ۸- بررسی مساله هشت وزیر برای تعیین زمان احتمالی نیاز به عقبگرد

ما مساله هشت وزیر را از این جهت که چه مواقعی می‌تواند نیاز به عقبگرد در حین حل آن به وجود بیاید، مورد بررسی قرار دادیم و به نظر می‌رسد که این بررسی بایستی برای هر مساله ارضای محدودیت مورد حل دیگر نیز صورت گیرد. در این بررسی ابتدا تعداد مقادیری که یک مقدار از یک متغیر با آنها ناسازگار می‌باشد، محاسبه و تعیین شده است. در شکل (۱۰) این تعدادها برای تمامی مقادیر از دامنه مقادیر تمامی متغیرهای مساله هشت وزیر نشان داده شده‌اند. همانطور که در شکل (۱۰) مشاهده می‌شود هر چقدر به سمت مرکز صفحه شطرنج نزدیکتر می‌شویم، تعداد عدم سازگاریها بیشتر می‌شود. تعداد متوسط عدم سازگاریها برای هر خانه از صفحه شطرنج فوق برابر است با مجموع کل اعداد بالا تقسیم بر ۶۴ که برابر است با  $15.74 = 1008/64$ . بنابراین متوسط تعداد مقادیری که یک مقدار از هر وزیر از وزیران دیگر را تهدید می‌کند برابر است با  $2.25 = 15.74/7$  و حال  $4 \approx 3.56 = 8/2.25$  تعداد وزیرانی را نشان می‌دهد که در صورتی که مقادیر موفق و سازگاری نسبت به یکدیگر انتخاب نموده باشند، احتمال دارد که حداقل یکی از وزیران ناموفق، دیگر انتخابی در دامنه مقادیر خود نداشته باشد و به عبارت دیگر نیاز به عقبگرد وجود داشته باشد.

#### ۹- بررسی الگوریتمهای آینده‌نگر در حل مساله هشت وزیر از نظر زمان نیاز به عقبگرد

ما الگوریتمهای آینده‌نگر FC، PL، FL و MFL را بر حسب متوسط تعداد متغیرهایی که پس از مقداردهی به آنها نیاز به عقبگرد به وجود می‌آید و نیز تعداد عقبگردها در یافتن اولین جواب مساله هشت وزیر مورد بررسی قرار دادیم. جدول

نشان داده شده در شکل (۱۱) مقادیر به دست آمده برای هر یک از این الگوریتمها را نشان می‌دهد. همانطور که در شکل (۱۱) مشاهده می‌شود تعداد متغیرهای مقداردهی شده در هنگام عقبگرد در حالت متوسط، عدد ۴ به دست آمده در بخش قبل را تایید می‌کند. اعداد فوق نشان می‌دهند که از بالا به پائین، الگوریتمها آینده‌نگری بیشتری را انجام داده و با کاستن از انتخابهای اشتباه در آینده، تعداد عقبگردها را کاهش داده و در قدمهای کمتری نیاز به عقبگرد را تشخیص می‌دهند.

#### ۱۰- روشهای مختلف تغییر پارامترهای الگوریتم مبتنی بر اتوماتانهای یادگیر

نتایج ارائه شده در بخش ۹ نشان می‌دهند که در حین حل مساله هشت وزیر پس از مقدار دهی موفق به ۳ الی ۴ وزیر، احتمال نیاز به عقبگرد وجود دارد، بنابر این بایستی از این مرحله به بعد با احتیاط عمل شود. ما با بررسی الگوریتم اتوماتان یادگیر این ایده به ذهنمان رسید که پارامترهای تنبیه و تشویق را بر مبنای:

- مشخصات مساله و مقدار انتخاب شده برای هر متغیر
- میزان موفقیت به دست آمده
- میزان احتمال نیاز به عقبگرد

تعیین نمائیم. ما در بخشهای زیر به بررسی تاثیر سه مورد فوق در طراحی روشهای تغییر پارامترهای الگوریتم اتوماتان یادگیر می‌پردازیم.

#### ۱-۱۰ مشخصات مساله و مقدار انتخاب شده برای هر متغیر

ما در شکل (۱۰) جدولی از عدم سازگاریها برای خانههای شطرنج مساله هشت وزیر نمایش دادیم. عدد نشان داده شده در سطر  $i$ ام و ستون  $j$ ام را  $C_{ij}$  در نظر بگیرید. اولین موضوعی که به ذهن می‌رسد این است که احتمال اولیه انتخاب مقادیر را نه به صورت مساوی بلکه برحسب  $C_{ij}$  آن تعیین کنیم. بدین صورت که آنهایی که  $C_{ij}$  کمتری داشته باشند، احتمال اولیه آنها بیشتر باشد. اگر احتمال اولیه را  $p_{ij}(0)$  و  $M$  را تعداد مقادیر در دامنه مقادیرمتغیر  $i$ ام در نظر بگیریم، در این صورت  $p_{ij}(0)$  را می‌توان به صورت زیر محاسبه نمود:

$$p_{ij}(0) = \frac{1/C_{ij}}{\sum_{k=1}^M 1/C_{ik}}$$

سپس برای هر یک از خانهها یا مقادیر، دو پارامتر  $A_{ij}$  و  $B_{ij}$  را به صورت زیر تعریف می‌کنیم:

$$A_{ij} = \frac{\sum_{k=1}^M C_{ik} - C_{ij}}{\sum_{k=1}^M C_{ik}} \quad \text{و} \quad B_{ij} = \frac{\sum_{k=1}^M 1/C_{ik} - 1/C_{ij}}{\sum_{k=1}^M 1/C_{ik}}$$

پارامترهای  $A_{ij}$  و  $B_{ij}$  خودشان پارامترهای تنبیه یا تشویق نیستند، بلکه پارامترهای تشویق و تنبیه ضریبی از به ترتیب  $A_{ij}$  و  $B_{ij}$  خواهند بود. این امر باعث می‌شود که خانهها یا مقادیری که  $C_{ij}$  کوچکتری داشته باشند، بیشتر تشویق شده و کمتر تنبیه شوند و خانههایی که  $C_{ij}$  بیشتری داشته باشند، کمتر تشویق شده و بیشتر تنبیه شوند.

#### ۱۰-۲ میزان موفقیت به دست آمده

اگر پارامتر  $a$  در الگوریتم اتوماتان یادگیر را به عنوان میزان امیدواری برای رسیدن به جواب در نظر بگیریم، به نظر می‌رسد که ثابت بودن آن صحیح نباشد. بدین منظور ما سعی کردیم پارامتر  $a$  را بر حسب موفقیت‌هایی که در یک قدم به دست می‌آیند و نیز مشخصات مساله و مقدار انتخاب شده برای هر متغیر به صورت جداگانه تعیین نمائیم.

اگر پارامتر  $S_i$  برابر تعداد همسایگان (و خود) اتوماتان  $i$ ام بوده و  $R_i$  برابر تعداد همسایگان موفق اتوماتان  $i$ ام (و خود) اتوماتان در صورت موفق بودن) باشد، پارامتر  $X_i$  را برابر  $\frac{R_i}{S_i}$  در نظر می‌گیریم. این پارامتر میزان موفقیت اتوماتان  $i$ ام و

همسایگان او را در یک قدم نشان می‌دهد. اگر پارامتر  $Y_i$  را برابر میزان امید آتوماتان  $\lambda$  برای رسیدن به جواب در نظر بگیریم، در ساده‌ترین حالت می‌توان پارامتر  $Y_i$  را مساوی با پارامتر  $X_i$  در نظر گرفت. نمودار رابطه بین پارامترهای  $X_i$  و  $Y_i$  در این حالت در شکل (۱۲) نشان داده شده است. حال اگر آتوماتان  $\lambda$  در یک قدم، مقدار  $\lambda$  را از دامنه خود انتخاب کرده باشد و این انتخاب موفق بوده باشد، پارامتر تشویق  $a_i$  را می‌توان از رابطه  $a_i = Y_i \times A_{ij}$  محاسبه نمود. اما اگر انتخاب مقدار  $\lambda$  برای آتوماتان  $\lambda$  موفقیت آمیز نباشد و پارامتر  $T_i$  برابر با تعداد همسایگان ناسازگار با انتخاب مقدار  $\lambda$  باشد، پارامتر تنبیه  $b_i$  را می‌توان از رابطه  $b_i = Y_i \times \left( \frac{S_i}{S_i - Y_i \times T_i} \right) \times B_{ij}$  محاسبه نمود. البته لازم به ذکر است که این رابطه به صورت تجربی به دست آمده است.

با در نظر گرفتن مطالب گفته شده در بخش ۱۰-۱ و مطالب فوق، الگوریتم جدید آتوماتان یادگیر طراحی و پیاده‌سازی شده و با الگوریتم آتوماتان اولیه با پارامترهای ثابت  $a$  و  $b$  مورد مقایسه قرار گرفت. این مقایسه بدین صورت انجام شد که با تعیین دامنه تعییرات  $a$  بین ۸ و ۱ با طول قدمهای ۰.۲ و  $b=۱.۲$  و در هر قدم با ۱۰ بار اجرای هر دو الگوریتم، نمودار مقدار متوسط تعداد تکرارهای هر دو الگوریتم در حل مساله هشت وزیر تهیه شد که این نمودار در شکل (۱۳) نشان داده شده است. علامت  $\{1\}$  نشان دهنده الگوریتم اولیه و علامت  $\{2\}$  نشان دهنده الگوریتم طراحی شده با پارامترهای متغیر است. البته تغییر پارامتر  $a$  بین ۸ و ۱ تنها مربوط به الگوریتم  $\{1\}$  بوده و الگوریتم  $\{2\}$  مقدار پارامترهایش را خودش تعیین می‌کند.

#### ۱۰-۳ میزان احتمال نیاز به عقبگرد

همانطور که در بخشهای ۸ و ۹ به آنها اشاره شد، در مساله هشت وزیر پس از تعیین مقدار موفق برای ۳ الی ۴ وزیر یا متغیر، امکان نیاز به عقبگرد وجود دارد. بنابر این بایستی در تشویق کردن محتاط بوده و از تنبیه نیز اجتناب نمود، چرا که عدم سازگاری می‌تواند به دلیل انتخاب اشتباه (گرچه سازگار) سایر متغیرها یا آتوماتانها باشد. بنابراین رابطه  $Y_i = X_i$  بین میزان موفقیت به دست آمده و میزان امیدواری برای رسیدن به جواب صحیح نمی‌باشد. اگر پارامتر  $H_i$  را برابر آستانه نیاز به عقبگرد در نظر بگیریم (که این مقدار در مساله هشت وزیر برابر  $۴/۸ = ۰.۵$  می‌باشد)، بایستی مقدار  $Y_i$  در مواقعی که  $X_i$  بین صفر و  $H_i$  باشد افزایش یافته و مواقعی که  $X_i$  بین  $H_i$  و یک باشد، کاهش یابد. بدین منظور ما دو نمودار نمایش داده شده در شکل (۱۴) را به صورت تجربی بین  $X_i$  و  $Y_i$  در نظر گرفته و الگوریتمهای متناظر با این دو نمودار را به طور جداگانه با الگوریتم اولیه آتوماتان یادگیر به همان صورتی که در بخش ۱۰-۲ گفته شد با یکدیگر مقایسه کردیم. نمودار مقایسه متناظر با شکل (۱۴-الف) نمودار نشان داده شده در شکل (۱۵-الف) بوده و نمودار مقایسه متناظر با نمودار شکل (۱۴-ب) نمودار نشان داده شده در شکل (۱۵-ب) می‌باشد. در شکل (۱۵-الف) علامت  $\{3\}$  متناظر با الگوریتم متناظر نمودار شکل (۱۴-الف) بوده و در شکل (۱۵-ب) علامت  $\{4\}$  متناظر با الگوریتم متناظر نمودار شکل (۱۴-ب) می‌باشد. همانطور که در این دو نمودار مشاهده می‌شود، گرچه الگوریتمهای  $\{3\}$  و  $\{4\}$  محاسبات زیادی را انجام می‌دهند و تلاش زیادی برای طراحی این الگوریتمها صرف شده است، ولی بهبود قابل ملاحظه‌ای نسبت به الگوریتم آتوماتان یادگیر اولیه با پارامترهای ثابت  $a$  بزرگ و  $b$  کوچک ارائه نمی‌دهند و تقریباً با آن معادل از آب درآمدند.

#### ۱۱- الگوریتمهای یادگیر دیگر برای حل مسائل ارضای محدودیت

در هر یک از الگوریتمهای مختلف حل مسائل ارضای محدودیت، به نوعی می‌توان یادگیری را مشاهده کرد. به عنوان مثال، هر یک از الگوریتمهای عقبگرد استاندارد<sup>۱</sup>، بررسی عقبرو<sup>۲</sup> و نشانگذاری عقبرو<sup>۳</sup> به نوعی از تکرار اعمالی که در گذشته انجام داده‌اند اجتناب می‌کنند، خصوصاً دو الگوریتم بررسی عقبرو و نشانگذاری عقبرو، به ترتیب شکستها و پیروزیهای گذشته را به یاد می‌سپارند که این امر را می‌توان به نوعی یادگیری در نظر گرفت. با اینحال هدف از یادگیری در حل مسائل ارضای محدودیت، بیشتر یادگیری محدودیتهای جدید می‌باشد. به عنوان مثال مساله ۴ وزیر را در نظر بگیرید.



در این مساله مقدار یک از وزیر اول با مقادیر یک و دو از وزیر دوم ناسازگار است. اما اگر پس از دادن مقدار یک به وزیر اول برای وزیر دوم مقدار سه انتخاب شود، برای وزیر سوم هیچ انتخابی باقی نمی‌ماند. بنابر این می‌توان فرض کرد که مقدار یک از وزیر اول با مقدار سه از وزیر دوم ناسازگار است، چیزی که در ابتدا به صورت صریح بیان نشده است. الگوریتمهای مختلف یادگیر وضعیتهای نامناسب برای یادگیری چنین محدودیتهایی معرفی شده‌اند. محدودیتهایی که از وضعیتهای نامناسب فرا گرفته می‌شوند، می‌توانند یکتایی، دوتایی و یا حتی چندتایی باشند، در عین حال که خود مساله ارضای محدودیت، می‌تواند یک مساله ارضای محدودیت دودویی باشد. امکان چندتایی بودن محدودیتهای جدید، یادگیری و حفظ و بررسی سازگاری بر طبق آنها را مشکل می‌سازد [4,5].

اگر بخواهیم الگوریتمهای یادگیر وضعیتهای نامناسب را با الگوریتم آتوماتان یادگیر از نظر چندتایی بودن محدودیتهای فراگرفته شده مقایسه کنیم، می‌بینیم که الگوریتم آتوماتان یادگیر تنها به دنبال یادگیری محدودیتهای یکتایی است، بدین معنی که با کاهش احتمال انتخاب یک مقدار از دامنه مقادیر یک متغیر، عملاً به سوی حذف آن از دامنه مقادیر و تعیین یک محدودیت یکتایی جدید پیش می‌رود.

## ۱۲- خلاصه و نتیجه‌گیری و پیشنهادات ادامه کار

مسائل ارضای محدودیت، الگوریتمها و تکنیهای متنوعی را به چالش طلبیده‌اند که آتوماتان یادگیر یکی از آنهاست. الگوریتم آتوماتان یادگیری که در این مقاله به آن پرداختیم، اولین کاری بود که در این زمینه انجام شده بود. ما از این الگوریتم برای حل دو مساله هشت وزیر و درک تصویر با برچسب‌دهی والتز استفاده کردیم. با ایده‌های مطرح شده در این الگوریتم، یک الگوریتم توزیع شده برای حل مسائل ارضای محدودیت طراحی و پیاده‌سازی کردیم که انتخابهای موفق را حفظ نموده و به تصحیح انتخابهای ناموفق می‌پرداخت و در صورت لزوم به صورت کاملاً توزیع شده عقبگرد می‌نمود. سپس الگوریتم آتوماتان یادگیر و نیز الگوریتم توزیع شده را با الگوریتمهای آینده‌نگر از لحاظ تعداد قدمهای اجرا و تعداد بررسیهای سازگاری در رسیدن به جوابهای مسائل ۴، ۵، ۶، ۷، ۸ و ۹ وزیر مورد مقایسه قرار دادیم و دیدیم که الگوریتم توزیع شده از لحاظ تعداد قدمها با الگوریتمهای آینده‌نگر قابل مقایسه بوده و تعداد بررسیهای سازگاری آن نسبت به الگوریتمهای آینده‌نگر بسیار کم است. ولی تعداد قدمهای اجرا و تعداد بررسیهای سازگاری الگوریتم آتوماتان یادگیر نسبت به الگوریتمهای آینده‌نگر بسیار زیاد است. سپس مساله هشت وزیر را از نظر زمان احتمالی نیاز به عقبگرد تحلیل نموده و به این نتیجه رسیدیم که در پی مقداردهی موفق به نصف وزیران، احتمال نیاز به عقبگرد وجود دارد و نتایج تجربی در رابطه با حل این مساله با الگوریتمهای آینده‌نگر این امر را تا حدودی به اثبات رساند و سپس با استفاده از همین موضوع، روشهایی برای تغییر پارامترهای تشویق و تنبیه الگوریتم آتوماتان یادگیر ابداع کردیم. ولی متأسفانه نتوانستیم به بهبود قابل ملاحظه‌ای در تعداد قدمهای اجرای این الگوریتم دست یابیم و اما چرا؟ جواب به سوال فوق مشکل است اما به نظر نگارنده جواب به این سوال را بایستی در دو امر جستجو کرد:

**اول:** بررسی نتایج مقایسه الگوریتم آتوماتان یادگیر و الگوریتم توزیع شده با الگوریتمهای آینده‌نگر.

**دوم:** اینکه الگوریتم آتوماتان یادگیر با پارامترهای ثابت، زمانی زودتر به جواب می‌رسد که مقدار  $a$  نزدیک به یک بوده و مقدار  $b$  نزدیک به صفر باشد، چیزی که در پی اعمال الگوریتم بر روی مساله هشت وزیر و درک تصویر با برچسب‌دهی والتز مشاهده نمودیم.

یک آتوماتان پس از دهها قدم و تنها پس از یک موفقیت، احتمال متناظر با آخرین مقدار انتخاب شده را به طور وحشتناکی افزایش می‌دهد. این ذوق زدگی چه معنایی می‌تواند داشته باشد؟ آیا این عمل مصداق فراموشکاری است یا یادگیری؟ به نظر نگارنده این ذوق زدگی چیزی جز حرکت به سوی الگوریتم توزیع شده پیاده‌سازی شده نیست. بدین معنی که انتخابهای موفق بایستی حفظ شوند و تنها انتخابهای ناموفق اصلاح شوند. اما در عین انتخاب مقدار ۰.۹۳. برای پارامتر  $a$  و ۰.۱۲. برای پارامتر  $b$ ، تعداد قدمهای الگوریتم آتوماتان یادگیر نسبت به الگوریتم توزیع شده و الگوریتمهای

آینده‌نگر بسیار زیاد است. یعنی اینکه محیط عمل آتوماتانها هنوز طوفانی است. در حالت کلی ضعفهای الگوریتم آتوماتان یادگیر را می‌توان در سه عنوان زیر خلاصه کرد:

#### الف) تحلیل ضعیف وضعیت موجود

به علت عدم وجود محیط کاملا و واقعا تصادفی در حل مسائل ارضای محدودیت با آتوماتانهای یادگیر و خصوصا در مسائلی که محدودیت‌های دوتایی قرینه باشند و هر یک از آتوماتانها بتوانند سازگاری با این محدودیتها را بررسی کنند، نیازی به در نظر گرفتن مدل عمل و عکس‌العمل در حل این نوع از مسائل نیست و آتوماتانها می‌توانند بررسیهای سازگاری را پس از گرفتن انتخابهای سایرین انجام داده و وضعیت موجود را تا حد ممکن مورد تجزیه و تحلیل قرار دهند و حتی آینده‌نگری انجام داده و وضعیتهای نامناسب را تشخیص دهند. به نظر نگارنده ترکیب آینده‌نگری و یادگیری نتایج بهتری خواهد داشت.

#### ب) نگهداری اطلاعات کم

در الگوریتم توزیع شده اطلاعات متنوعی از جمله آخرین قدم ثابت ماندن یک متغیر، و ناسازگاریهای کشف شده ضبط و نگهداری می‌شوند و تا زمانی که نیاز به عقبگرد وجود نداشته باشد همه این اطلاعات حفظ می‌شوند. اما در الگوریتم آتوماتان یادگیر، برای هر مقدار تنها یک احتمال نگه داشته می‌شود که آن هم در پی یک موفقیت به طور بی‌سابقه‌ای تغییر می‌کند.

#### ج) یادگیری تنها محدودیت‌های یکتایی

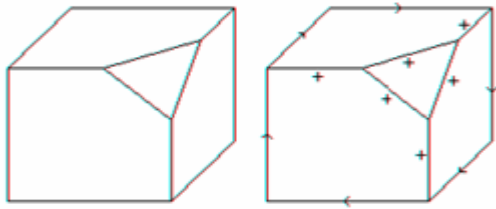
چون این الگوریتم وضعیت موجود را زیاد تحلیل نمی‌کند، بنابراین وضعیتهای نامناسب را نمی‌تواند تشخیص دهد، پس توانایی یادگیری محدودیت‌های چندتایی را نخواهد داشت.

به عنوان نتیجه گیری کلی، محیط طوفانی و آتوماتانهایی خودسر که بدون نگه داشتن اطلاعات کافی و یادگیریهای زیاد و بدون تحلیل مناسب وضعیت موجود، همیشه در حال جوش و خروش و تغییر مقادیر خود هستند، برای حل مسائل ارضای محدودیت که نیاز به محیطی آرام برای ارضای محدودیتها دارند، مناسب نیستند. فعالیت در زمینه‌های زیر راهگشای مشکلات اشاره شده در بالا خواهد بود:

- ترکیب آینده‌نگری با یادگیری و تحلیل تاثیر آن از لحاظ هزینه‌ای که صرف آن خواهد شد و نتیجه‌ای که به دست خواهد آمد.
- یادگیری محدودیت‌های چند تایی و یا ترکیب یادگیری وضعیتهای نامناسب با آتوماتانهای یادگیر.

### ۱۳- مراجع

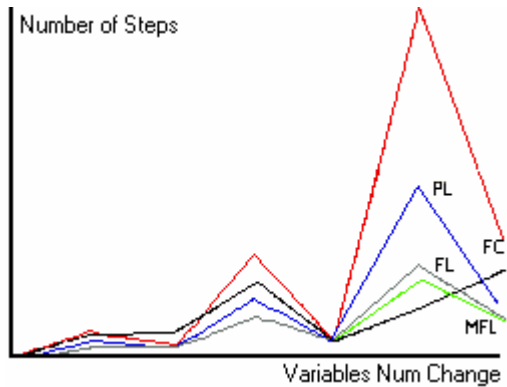
- 1 : F. Ricci. Constraint Reasoning with Learning Automata. Technical Report. Institute Per la Ricerca Scientifica. Marth 14. 1994.
- 2 : E. Rich, K. Knight. Artificial Intelligence. 2<sup>nd</sup> Ed. McGrawHill. 1991.
- 3 : R. M. Haralick, G. L. Elliot. Increasing Tree Search Efficiency for Costraint Satisfaction Problems. Artificial Intelligence 14 (1980). Pg 263-313.
- 4 : E. T. Richards, B. Richards. Nogood Learning for Constraint Satisfaction. IC Parc, Imperial College, London. {etr,ebr}@doc.ic.ac.uk
- 5 : D. Frost, R. Dechter. Dead\_end Driven Learning. Department of information and computer science, University of California, Irvine, CA 92717. {dfrost,dechter}@ics.uci.edu.



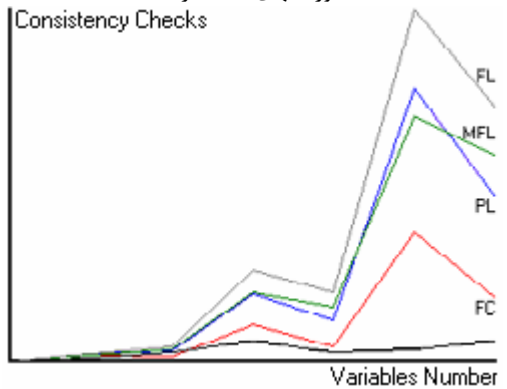
6 : بیگی. ح، میبیدی. م. حل مساله تناظر گراف توسط اتوماتانهای یادگیر. پنجمین کنفرانس بین‌المللی سالانه انجمن کامپیوتر ایران. دانشگاه شهید بهشتی. ص ۴۰۲ الی ۴۱۵.

7 : قاسم ثانی. غ، نمازی، م. روشی برای حل مسائل برچسبدهی سازگار. پنجمین کنفرانس بین‌المللی سالانه انجمن کامپیوتر ایران. دانشگاه شهید بهشتی. ص ۲۱۶ الی ۲۲۵.

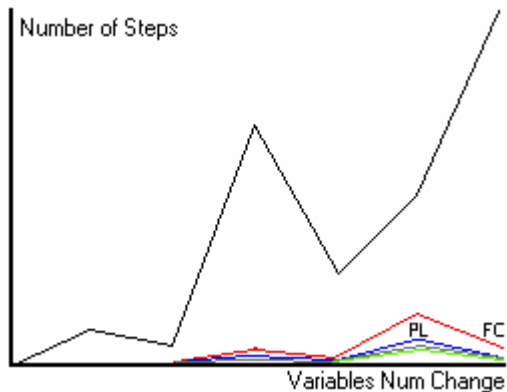
شکل(۵): نتایج اجرای الگوریتم اتوماتان یادگیر بر روی مساله



درک تصویر با ۱۵ بار اجرا در حداکثر ۵۰۰۰۰ قدم.  
شکل(۶) : مقایسه تعداد قدمهای الگوریتم توزیع شده با الگوریتمهای آینده‌نگر.



شکل(۷) : مقایسه تعداد بررسیهای سازگاری الگوریتم توزیع شده با الگوریتمهای آینده‌نگر.



شکل(۱): مکعب گوشه بریده.

b	a	0	.25	.5	.75	1
0	15730 6%	0 0%	33 6%	0 0%	23 13%	
.25	0 0%	2471 6%	19276 6%	10769 46%	6083 86%	
.5	2014 20%	17483 13%	5591 13%	1801 13%	3981 6%	
.75	11396 13%	4 6%	0 0%	8595 6%	7983 20%	
1	5742 6%	6482 13%	0 0%	3333 6%	18600 6%	

شکل(۲): نتایج اجرای الگوریتم اتوماتان یادگیر بر روی مساله ۸ وزیر با ۱۵ بار اجرا در حداکثر ۲۰۰۰۰ قدم.

b	a	.2	.4	.6	.8	1
.05	1719 4%	1076 76%	1767 72%	1418 76%	1356 96%	
.1	2174 8%	2616 16%	1149 88%	737 100%	598 100%	
.15	2263 4%	1967 4%	3270 20%	1647 84%	1215 100%	
.2	0 0%	0 0%	3106 16%	2405 20%	1340 92%	

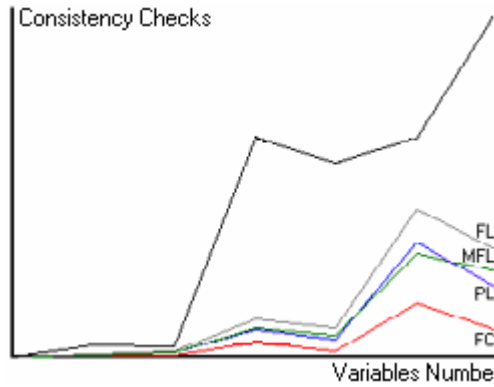
شکل(۳): نتایج اجرای الگوریتم اتوماتان یادگیر بر روی مساله ۸ وزیر با ۲۵ بار اجرا در حداکثر ۵۰۰۰ قدم.

b	a	0	.25	.5	.75	1
0	6337 6%	0 0%	0 0%	0 0%	0 0%	
.25	4332 13%	6750 20%	1436 13%	4156 40%	2830 33%	
.5	4681 6%	5541 40%	5109 26%	5154 33%	4109 26%	
.75	7821 26%	4407 26%	4132 20%	5433 40%	6715 40%	
1	5020 13%	4790 13%	5984 13%	3704 13%	6356 46%	

شکل(۴): نتایج اجرای الگوریتم اتوماتان یادگیر بر روی مساله درک تصویر با ۱۵ بار اجرا در حداکثر ۲۰۰۰۰ قدم.

b	a	0	.25	.5	.75	1
0	19789 33%	0 0%	0 0%	0 0%	0 0%	
.25	15309 66%	22353 66%	17875 80%	13620 60%	12438 73%	
.5	15169 53%	16917 80%	20142 73%	22550 80%	16534 73%	
.75	32557 53%	24676 40%	17764 80%	28819 86%	18029 73%	
1	16614	14550	26589	25465	20678	

شکل (۸): مقایسه تعداد قدمهای اجرای الگوریتم اتوماتان یادگیر با الگوریتمهای آینده‌نگر.



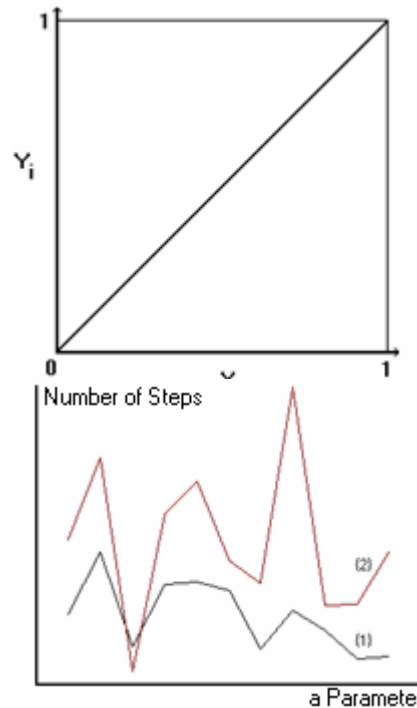
شکل (۹): مقایسه تعداد بررسیهای سازگاری الگوریتم اتوماتان یادگیر با الگوریتمهای آینده‌نگر.

14	14	14	14	14	14	14	14
14	16	16	16	16	16	16	14
14	16	18	18	18	18	16	14
14	16	18	20	20	18	16	14
14	16	18	20	20	18	16	14
14	16	18	18	18	18	16	14
14	16	16	16	16	16	16	14
14	14	14	14	14	14	14	14

شکل (۱۰): تعداد عدم سازگاریهای تک تک خانه‌های صفحه شطرنج.

الگوریتم	متوسط تعداد متغیرهای مقداردهی شده	تعداد عقبگرد
FC	5.37	35
PL	4.58	17
FL	3.64	14
MFL	3.38	13

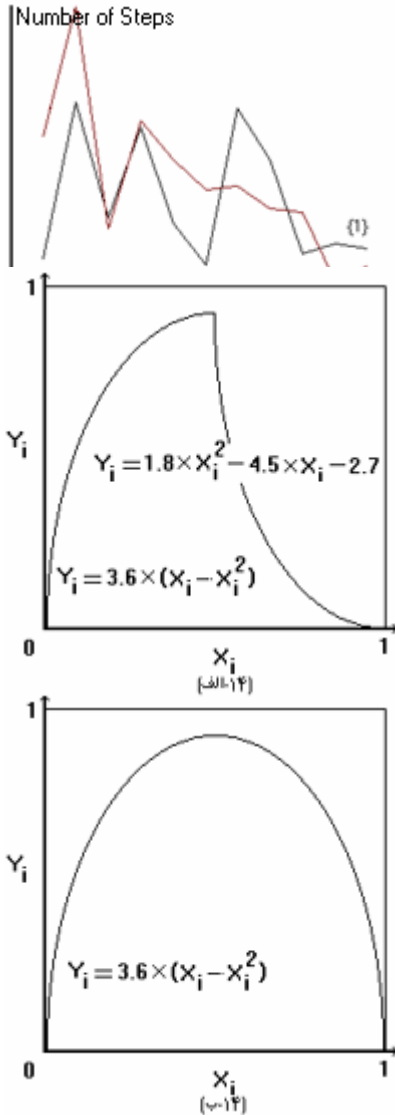
شکل (۱۱): زمان و تعداد عقبگردهای الگوریتمهای آینده‌نگر در یافتن اولین جواب مساله هشت وزیر.



شکل (۱۲): نمودار  $Y_i = X_i$ .

شکل (۱۳): مقایسه الگوریتم اولیه اتوماتان یادگیر با حالت  $Y_i = X_i$ .

شکل (۱۴): دو نمودار رابطه جدید بین  $Y_i$  و  $X_i$ .



شکل (۱۵): مقایسه الگوریتم اتوماتان اولیه و الگوریتمهای متناظر با نمودارهای شکل (۱۴).