دسته‌بندی حمله‌های شبکه به کمک الگوریتم‌های جستجوی محلی تطبیقی مبتنی بر مفاهیم یادگیری متالامارکی

مریم امیرحائری[۱]، زهرا احمدی[۲]، جعفر حبیبی[۳] و محمد صنیعی آباده[۴]

[۱] دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف، تهران، ایران
haeri@ce.sharif.edu
[۲] دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف، تهران، ایران
z_ahmadi@ce.sharif.edu
[۳] دانشیار، دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف، تهران، ایران
habibi@ce.sharif.edu
[۴] دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف، تهران، ایران
saniee@ce.sharif.edu

۱- مقدمه

در انجام جستجو، عموماً به دنبال برآورده کردن دو هدف اکتشاف (exploring) و بهره‌برداری (exploiting) در فضای مسأله هستیم [10]. اکتشاف به معنای اطمینان از سراسری بودن جستجو است و از آن جهت دارای اهمیت است که فضای مسأله باید به نحو قابل توجهی برای یافتن نقطه بهینه سراسری جستجو شود. در بهره‌برداری به دنبال یافتن پاسخ‌های بهتر حول یک پاسخ به دست آمده، و در واقع جستجوی محلی هستیم. الگوریتم‌های ژنتیک دارای ساختار قوی در جستجوی سراسری هستند اما برای تقویت مقوله بهره‌برداری نیاز به ترکیب با الگوریتم‌های جستجوی محلی دارند. لذا انتخاب روش‌های جستجوی محلی حائز اهمیت است؛ زیرا این روش‌ها ممکن است در مسأله‌ای مناسب بوده اما در مسأله دیگر عملکرد شایان ذکری نداشته باشند. به ترکیب الگوریتم‌های ژنتیک متعارف با الگوریتم‌های جستجوی محلی، الگوریتم‌های ممتیک گفته می‌شود. در این الگوریتم‌ها به دلیل وجود روش‌های جستجوی محلی، کارایی جستجو بالاتر می‌رود.

البته برای هر مسأله در ابتدا نمی‌توان روش جستجوی محلی مناسب‌تر را مشخص نمود. روش‌های جستجوی محلی بر اساس پیچیدگی‌های مسأله و فضا و پارامترهای آن، بسیار متفاوت عمل می‌کنند. با استفاده از چند روش جستجوی محلی در یک الگوریتم ممتیک و بکارگیری آنها در چند مرحله، می‌توان روش جستجوی محلی مناسب‌تر را برای مسأله مورد نظر پیدا نمود؛ به این روش یادگیری متالامارکی گفته می‌شود. با این روش احتمال استفاده از روش‌های جستجوی محلی مناسب مسأله را بالا می‌بریم.

الگوریتم‌های یادگیری متالامارکی براساس الگوریتم‌های تکاملی و یک روش جستجوی محلی برای بهبود تابع ارزش، عمل می‌کنند. تمامی روش‌های جستجوی محلی بجز روش گام تصادفی (روشی که در آن به تمامی memeها احتمال یکسان برای انتخاب داده می‌شود)، دارای پارامترهایی هستند که با تغییر آنها، می‌توان روش جستجو را برای یک مسأله خاص کاراتر و در حقیقت بایاس کرد. در شکل (۱) چارچوب کلی روش یادگیری متالامارکی آورده شده است.

الگوریتم‌های ممتیک با یادگیری متالامارکی، دارای انواع گوناگون هستند که در بخش بعد به معرفی انواع دسته‌های آن خواهیم پرداخت. این الگوریتم‌ها در مسائل با تعداد زیاد ماکزیمم‌های محلی (که روش‌های متعارف امکان یافتن ماکزیمم سراسری آنها را ندارند)، بعنوان گزینه کارا مورد توجه قرار گرفته‌اند.

الگوریتم ژنتیک، مانند بسیاری از تکنیک‌های جستجو و بهینه‌سازی، شامل تعدادی پارامتر عملگر است که مقادیرشان به‌طرز چشمگیری بر رفتار الگوریتم در مسأله مورد نظر تأثیر می‌گذارد. لذا برای افزایش کارایی لازم است که پارامترهای الگوریتم ژنتیک تنظیم شوند. در طول بیست سال گذشته تحقیقات زیادی بر روی تطبیق اتوماتیک پارامترهای الگوریتم ژنتیک انجام شده است. این

تحقیقات بر روی سرعت جهش، ترکیب (crossover) و تکنیک‌های تولید مجدد بوده که نتایج امیدوار کننده‌ای به همراه داشته است. مروری اجمالی بر دسته‌بندی تطبیقات در محاسبات تکاملی در [4,5] آمده است.

با این حال الگوریتم‌های ژنتیک متعارف معمولا به دلیل شکست در استفاده از اطلاعات محلی، سرعت همگرایی بسیار کمی در تعیین پاسخ نسبتاً دقیق دارند. این عامل باعث محدودیت الگوریتم‌های ژنتیکی در بسیاری از مسائل دنیای واقعی (که زمان محاسبه بسیار اهمیت دارد) می‌شود. الگوریتم‌های ممتیک درواقع رویکردهای فرامکاشفه‌ای (metaheuristic) مبتنی بر جمعیت و براساس اصل نئوداروینی تکامل طبیعی (اصل بقای اصلح) و نظریه meme داوکین (که meme را بعنوان واحدی از تکامل فرهنگی با توان اصلاح محلی می‌داند) هستند. در حالت کلی الگوریتم‌های ممتیک ترکیبی از جستجوی سراسری مبتنی بر جمعیت و رویه‌های پیشرفت محلی بوده و برای پاسخ‌گویی به بسیاری مسائل بهینه‌سازی موفق عمل کرده‌اند. البته مشکل الگوریتم‌های ممتیک آن است که برای مفید بودن، لازم است پارامترهای کنترلی بسیار زیادی را تنظیم نمود.

```
BEGIN
  Initialize: Generate an initial GA population.
  While (Stopping conditions are not satisfied)
    Evaluation of All Individuals in the Population
    For each individual in the population
      Pseudo-code to select local search methods. For example, Select LS using
      the Meta-Lamarckian Learning Strategy employed.
      Proceed with local improvement and replace the genotype in the
      population with the improved solution.
    End For
    Apply standard GA operators to create a new population; i.e., Selection,
    Mutation and Crossover.
  End While
END
```

شکل (۱): چارچوب کلی الگوریتم ممتیک با یادگیری متالامارکی.

در تحقیقات انجام شده در [9] نشان داده شده است که الگوریتم‌های ممتیک از الگوریتم ژنتیک بهتر عمل می کند، بخصوص در حالت وجود memeهای مخصوص مسأله. در تحقیقات بهینه‌سازی ترکیبیات گسسته، عبارت "ابَرمکاشفه" (hyperheuristic) برای توصیف ایده آمیخته‌سازی تعدادی meme مختلف با هم، بگونه‌ای که meme اعمال شده در هر نقطه تصمیم‌گیری ممکن است متفاوت باشد، وضع شد. در واقع ابرمکاشفه، مکاشفه‌ای برای انتخاب memeها تعریف می‌شود. هم‌چنین ایده استفاده از چند meme-ای و انتخاب تطبیقی memeها در هر نقطه انتخاب ارائه شده است [2,6]. هم‌چنین به‌طور هم‌زمان الگوریتم‌های هم‌تکاملی (co-evolving) چندین meme نیز معرفی شد [11,12]. تاکنون خلاصه‌ای از کارهای انجام شده در بخش تطبیق memeها در مسائل بهینه‌سازی ترکیباتی بیان شد. اما در حیطه بهینه‌سازی پیوسته، نشان داده شده است که انتخاب memeها در مسائل معیارسنجی گوناگون تأثیر بسزایی بر کارایی الگوریتم‌های ممتیک دارد [1]. هم‌چنین Keane و Ong عبارت "یادگیری متالامارکی" را برای معرفی ایده انتخاب چندmeme ای تطبیقی در جستجوی متالامارکی بکار بردند [10].

در این مقاله الگوریتم‌های ممتیک متفاوت از دیدگاه انتخاب memeها تحلیل و با یکدیگر مقایسه می‌شوند. در بخش بعد، یک دسته‌بندی بر روی الگوریتم‌های ممتیک انجام می‌دهیم و الگوریتم‌های ممتیک مورد بررسی را تشریح می‌کنیم. سپس در بخش ۳، یک تعریف اجمالی از مسأله مورد بررسی را بیان داشته و در بخش ۴ به تشریح روش‌های پیاده‌سازی الگوریتم‌ها می‌پردازیم. در بخش ۵ نتایج اجرای عملی و مقایسه الگوریتم‌های جستجوی محلی متفاوت آورده شده و در پایان نیز مقایسه‌ای بین الگوریتم‌های ممتیک با سایر روش‌های پیاده‌سازی شده از نظر کارایی به همراه یک جمع‌بندی کلی ارائه می‌نماییم.

۲- دسته‌بندی الگوریتم‌های ممتیک

در این بخش یک دسته‌بندی از تطبیق memeها در الگوریتم‌های ممتیک تطبیقی (الگوریتم‌هایی که در آنها بتوان براساس یادگیری، روش جستجوی محلی را متناسب با مسأله مورد نظر منطبق کرد) ارائه می‌شود.

همان‌طور که پیشتر اشاره شد، واژه "اَبَرمکاشفه" بعنوان یک استراتژی مدیریت انتخاب meme در هر لحظه لازم در [3] معرفی شده است. این مدیریت به ویژگی‌های memeها و محدوده فضای پاسخ در حال بازیابی بستگی دارد. در اَبَرمکاشفه از memeهای چندگانه در جستجوی تکاملی استفاده می‌شود. یک دسته‌بندی اَبَرمکاشفه‌ها در مسائل زمان‌بندی به‌صورت زیر است:

۱- تصادفی: در این گروه، اولین حالت روش "تصادفی ساده" است که در هر نقطه تصمیم‌گیری یک meme به‌صورت کاملا تصادفی و با احتمال ثابت انتخاب می‌شود. از این روش می‌توان بعنوان یک خط پایه برای مقایسه با سایر استراتژی‌ها استفاده کرد. در روش "نزول تصادفی" ابتدائاً انتخاب memeها به‌صورت تصادفی بوده و یک meme تا زمانی که پیشرفت محلی ایجاد کند، مورد استفاده خواهد بود. در حقیقت در این روش، نظر بر آن است که memeای که تاکنون بهبود ایجاد کرده، در آینده هم خوب عمل می‌کند. پس از آن‌که meme دیگر بهبودی حاصل نکرد، مجدداً به‌صورت تصادفی meme دیگری انتخاب خواهیم نمود. در روش "نزول تصادفی جایگشتی" مشابه روش نزول تصادفی عمل می‌کنیم، با این تفاوت که جایگشت انتخاب memeها از ابتدا مشخص شده است.

۲- حریصانه: این گروه مشابه تکنیک کورکورانه عمل می‌کند. بدین صورت که در هر مرحله، تمام memeها بر روی فرد آزمایش شده و memeای انتخاب می‌شود که بیشترین پیشرفت را داشته باشد. عیب آشکار این روش، هزینه محاسباتی بسیار بالای آن است.

۳- تابع انتخاب: در این گروه سعی در انتخاب معیاری یکپارچه براساس دانش کنونی از محدوده فضای پاسخ در حال بازیابی، برای برآورد میزان تأثیر یک meme داریم. این تابع انتخاب ترکیبی از سه جزء است. جزء اول بیان‌کننده پیشرفت‌های اخیر ایجاد شده با یک meme؛ جزء دوم توصیف کننده پیشرفت ایجاد شده با جفت‌های متوالی memeها است؛ و جزء نهایی زمان سپری شده از آخرین استفاده meme را نشان می‌دهد. پنج روش معرفی شده در این دسته در ادامه تشریح می‌شوند. در روش "انتخاب مستقیم" در هر نقطه تصمیم‌گیری meme با بهترین مقدار تابع انتخاب، برگزیده خواهد شد. در روش "انتخاب رتبه‌بندی شده"، memeها براساس مقدار تابع انتخاب رتبه‌بندی می‌شوند و memeهای با رتبه بالاتر به صورت فردی آزمایش شده و تنها memeای که بیشترین پیشرفت را با استفاده از یادگیری لامارکی داشته باشد، انتخاب می‌شود. در روش "انتخاب رولتی" یک meme، M_e، با استفاده از احتمال نسبی به کل پیشرفت‌ها که عبارت است از $\frac{F(Me)}{\sum_{e=1}^{\eta} F(Me)}$ و η تعداد کل memeهای مورد توجه و F تابع انتخاب هستند؛ انتخاب می‌شود. روش "انتخاب تجزیه‌ای" به هر جزء تابع انتخاب به صورت مجزا توجه می‌کند. به‌این صورت که تمامی memeها اعمال شده و بهترین پیشرفت محلی براساس تک تک اجزاء تابع انتخاب و مقدار خود آن ذخیره می‌شود. سپس meme با بهترین پیشرفت از میان آنها انتخاب می‌شود (در واقع از میان حداکثر چهار meme با رتبه‌های بالا در هریک از اجزاء). هم‌چنین ممکن است که از یک لیست تحریمی برای محدودیت memeها در هر نقطه تصمیم‌گیری استفاده شود که به آن "جستجوی تحریمی" گفته می‌شود.

هم‌چنین یک مکانیزم توارث ساده برای جستجوی ترکیبی گسسته پیشنهاد شده است [2,6]. در این روش هر فرد با اطلاعات ژنتیکی و ممتیکی خود ساخته و نمایش می‌شود. اطلاعات ممتیکی در بخش ژنتیکی رمزگذاری شده تا memeها جهت استفاده در جستجوی محلی در همسایگی پاسخ بکار گرفته شوند. Smith بر روی الگوریتم‌های ممتیک هم‌تکاملی با مکانیزم‌های مشابه مدیریت انتخاب memeها، کار کرده است [11,12]. در این الگوریتم‌های خودتطبیق، هم‌زمان اطلاعات ژنتیکی و انتخاب memeها طی جستجو تکامل بخشیده می‌شوند.

یک دسته‌بندی الگوریتم‌های ممتیک براساس گونه تطبیق و سطح دانش مورد استفاده (سطح تطبیق) در [9] ارائه شده است که در جدول (۱) نمایش داده‌ایم. گونه تطبیق در واقع توجه بر بازخورد است. در اینجا بازخورد به‌صورت پیشرفت بدست آمده از memeهای انتخاب شده بر روی کروموزم‌های مورد جستجو تعریف می‌شود و شامل سه دسته است:

۱- ایستا: هنگامی که از هیچ بازخوردی در طول جستجوی تکاملی استفاده نشود. روش تصادفی ساده، که پیشتر توضیح داده شد، جزء این دسته است.

۲- تطبیقی: هنگامی که در هر نقطه تصمیم‌گیری از الگوریتم در انتخاب memeها از بازخورد استفاده شود. این تطبیق ممکن است کمّی یا کیفی باشد. در تطبیق‌های کیفی مقدار دقیق بازخورد از اهمیت کمی برخوردار است و کیفیت meme دارای اهمیت است. تاجایی که meme در فرایند یادگیری محلی پیشرفت داشته باشد، در مرحله بعد مورد استفاده قرار می‌گیرد. روش‌های نزول

تصادفی، نزول تصادفی جایگشتی و جستجوی تحریمی جزو این دسته‌اند. اما روش‌های حریصانه، انتخاب مستقیم، انتخاب رتبه‌بندی شده، انتخاب رولتی و تجزیه به زیرمسائل برپایه بازخورد کمّی از memeها هستند.

۳- خود تطبیق: در واقع پیاده‌سازی ایده خودتطابقی memeها است و نمایش ممتیک memeها به‌صورت بخشی از فرد کد شده و از تکامل استاندارد پیروی می‌کند.

سطح تطبیق به سطح دانش بدست آمده از memeها در گذشته گفته می‌شود و به سه سطح خارجی، محلی و سراسری تقسیم می‌شود.

۱- خارجی: به حالاتی که از هیچ دانشی در انتخاب memeها استفاده نشود و memeها از استخری از کل memeهای مخصوص مسأله انتخاب شوند، گفته می‌شود.

۲- محلی: در این سطح از دانش ساده‌ای از گذشته استفاده می‌شود. برای مثال الگوریتم‌های حریصانه، نزول تصادفی و نزول تصادفی جایگشتی برپایه پیشرفت‌های بدست آمده در حال یا دقیقاً گام پیشین تصمیم‌گیری می‌کنند. روش‌های تجزیه به‌زیرمسائل،

جدول(۱): دسته های مختلف الگوریتم ممتیک.

نوع تطبیق		سطح تطبیق		
		خارجی	محلی	سراسری
ایستا		یادگیری متالامارکی پایه یا تصادفی ساده		
تطبیقی	تطبیقی کمّی		نزول تصادفی	جستجوی تحریمی
	تطبیقی کمّی		حریصانه	انتخاب رتبه بندی شده انتخاب مستقیم انتخاب رولتی
خود تطبیق			چند ممی MAهای هم‌تکاملی	

چندmeme ای و هم‌تکاملی، تنها براساس دانش بدست آمده از k نزدیک‌ترین افراد یا والدین موجود در جستجوهای قبل، انتخاب می‌کنند؛ به همین دلیل در دسته تطبیق‌های سطح محلی در نظر گرفته می‌شوند.

۳- سراسری: زمانی که از دانش گذشته به‌طور کامل در انتخاب memeها استفاده شود. الگوریتم‌هایی چون انتخاب مستقیم، انتخاب رتبه‌بندی شده، انتخاب رولتی، انتخاب تجزیه‌ای و جستجوی تحریمی، به‌دلیل استفاده از دانش کامل بدست آمده بر روی memeها، جزو این دسته به‌حساب می‌آیند.

۳- شرح مسأله

به منظور پیاده‌سازی الگوریتم‌های ممتیک برای حل مسأله دسته‌بندی و تشخیص نفوذ به شبکه، از الگوریتم‌های ژنتیکی بعنوان بستر حل مسأله استفاده کرده‌ایم. الگوریتم‌های ژنتیکی بکار برده شده براساس قوانین فازی و استنتاج به‌کمک الگوریتم قانون برنده پیاده-سازی شده‌اند. افراد جمعیت در این الگوریتم ژنتیکی، قوانین فازی هستند. هر قانون دارای ۴۱ ویژگی است و این ویژگی‌ها اعضای مجموعه فازی ارائه شده در شکل (۲) هستند.

	S : Small MS : Medium Small M : Medium ML : Medium Large L : Large
	DC : don't care

شکل (۲): مجموعه‌های فازی نخستین استفاده شده در این پژوهش.

بنابراین اساس کار، یک سیستم فازی تکاملی است که قوانین فازی اگر-آنگاه را به‌طریقه افزایشی یاد می‌گیرد و الگوریتم تکاملی در هر زمان یک قانون دسته‌بندی را بهینه می‌کند. مکانیزم یادگیری، وزن آن دسته از نمونه‌های آموزشی که توسط قانون جدید به‌درستی دسته‌بندی شده‌اند را کاهش می‌دهد. در نتیجه، در دور بعد تولید قانون، بر روی قوانینی تمرکز می‌شود که اشتباهاً دسته‌بندی و یا اصلاً دسته‌بندی نشده‌اند. در هر تکرار، قانونی که بتواند توزیع کنونی نمونه‌ها را بهتر از قانون‌های دیگر موجود دسته‌بندی کند، انتخاب و به مجموعه نهایی قانون‌ها اضافه می‌شود. ایده اصلی در استفاده از مکانیزم تقویتی، جمع‌آوری فرض‌های متعدد حاصل بکار بردن یک الگوریتم یکسان بر روی توزیع‌های مختلف داده‌های آموزشی، و تبدیل آنها به یک دسته‌بندی مرکب یکتا است. در چارچوب یادگیری ذکر شده، از تابع ارزش استفاده نموده‌ایم که طریقه محاسبه آن در روابط (۱) تا (۳) آورده شده است.

$$f_p = \frac{\sum_{k|c^k=c_i} w^k \mu_{R_i}(x^k)}{\sum_{k|c^k=c_i} w^k} \quad (۱)$$

$$f_N = \frac{\sum_{k|c^k\neq c_i} w^k \mu_{R_i}(x^k)}{\sum_{k|c^k\neq c_i} w^k} \quad (۲)$$

$$fitness(R_j) = w_P\, f_P - w_N\, f_N \quad (۳)$$

در اینجا f_P: نرخ نمونه‌های آموزشی که به‌صورت صحیح توسط قانون R_i دسته‌بندی شده‌اند (دسته‌بندی درست)؛ f_N: نرخ نمونه‌های آموزشی که به‌صورت اشتباه توسط قانون R_i دسته‌بندی شده‌اند (دسته‌بندی اشتباه)؛ w^k، وزن نشان‌دهنده تعداد تکرار (frequency) نمونه x^k در مجموعه آموزشی؛ w_P، وزن کلاس‌بندی صحیح و w_N، وزن کلاس‌بندی اشتباه می‌باشد.

طرح عمده‌ای از سیستم فازی تکاملی تکراری در ادامه ارائه شده است:

- گام ۱: یک جمعیت اولیه از قانون‌های اگر-آنگاه بر اساس وزن نمونه‌های آموزشی تولید کنید. (مقداردهی اولیه)
- گام ۲: قوانین فازی اگر-آنگاه جدیدی با استفاده از عملگرهای ژنتیکی تولید کنید. (تولید)
- گام ۳: انتخاب یک meme و اجرای یک جستجوی محلی بر روی افراد شایسته جمعیت.
- گام۴: بخشی از مجموعه کنونی را با قوانین تازه تولید شده جایگزین کنید. (جایگزینی)
- گام ۵: چنانچه شرایط توقف برنامه مشاهده شود، حلقه درونی الگوریتم را خاتمه دهید؛ در غیر این صورت به گام ۲ بروید. (چک کردن شرایط پایان حلقه درونی)
- گام ۶: چنانچه شرایط توقف برنامه مشاهده شود، حلقه بیرونی الگوریتم را خاتمه دهید؛ در غیر این صورت به گام بعدی بروید. (چک کردن شرایط پایان حلقه بیرونی)

- گام ۷: وزن جدید هر یک از نمونه‌های آموزشی که توسط قانون فازی جدید پوشش داده می‌شوند را تنظیم کرده و به گام ۱ بروید. (تنظیم وزن)

۴- شرح طراحی meme

در طراحی memeها نکات مختلفی از سوی طراح لحاظ می‌شود، برخی از ملزومات طراحی را می‌توان از سوال‌های زیر دریافت:

- آیا مدل بالدوینی یا لامارکی در این مسأله روش ارجح است؟
- فضای مسأله به چه صورت است؟ بهترین نسبت بین جستجوی محلی و جستجوی سراسری چیست؟
- چه وقت و در کجا باید از جستجوی محلی در مراحل تکامل استفاده شود؟
- کدام افراد در جمعیت باید بهبود داده شوند و این افراد چگونه باید انتخاب شوند؟
- برای هر روش جستجو از لحاظ محاسباتی چه میزان می‌توان هزینه کرد؟ چه مقدار محاسبات می‌توانیم انجام دهیم؟

۴-۱- طراحی یک meme کارآمد

دورنمای (landscape) مسأله ما دارای ناهمواری‌های بسیار است که سرتاسر آن پر از بیشینه‌ها و کمینه‌های محلی است. از این رو فضای مسأله خود دارای پیچیدگی‌های بسیاری است که باعث افزایش محاسبات می‌شود. بنابراین انتخاب ساختار meme از اهمیت خاصی برخوردار است. بکار بردن memeهایی با ساختار پیچیده نه تنها باعث یافتن جواب‌های بهتر نخواهد شد؛ بلکه زمان محاسبات را نیز بسیار افزایش می‌دهد.

برای طراحی meme، مسائل زیادی را مطالعه و memeهایی را که برای حل آنها طراحی شده بود، مورد بررسی قرار دادیم. مسأله مطرح شده در [7,8]، مسأله ای با پیچیدگی مشابه فضای مسأله ما بود. در این مقاله برای طراحی meme از ساختار ساده و سبکی استفاده شده است که در فضای پیچیده مسأله به خوبی عمل می‌کند و این امکان را می‌دهد تا با محاسبات کمتر، memeهای بیشتری روی مسأله اعمال شود. ما از ایده این مقاله استفاده کردیم و سعی کردیم تا meme ای برای مسأله خود طراحی کنیم که هم ساده و سبک باشد و هم با نیازها و ملزومات فضای مسأله هماهنگی داشته باشد.

فضا و دورنمای مسأله مورد بررسی پر از بیشینه های محلی است، بنابراین از آنجا که احتمال قرارگیری در بیشینه محلی و عدم امکان پیشرفت بسیار زیاد است؛ در روش جستجوی محلی نیازمند مکانیزمی هستیم تا بتوانیم از روی این بیشینه‌های محلی پرش کنیم و به قسمت‌های دیگر فضای مسأله برسیم. بهترین مکانیزمی که می‌تواند در این امر ما را یاری نماید، استفاده از تصادف و پرش-های تصادفی است. پیش از این هم مشاهده کرده‌ایم که در چنین مسأله‌ای پرش‌های تصادفی می‌تواند بسیار راهگشا باشد. برای همین در ساختار meme و یا به‌عبارت دیگر، جستجوهای محلی طراحی شده، به تصادف اهمیت زیادی دادیم و memeها را به‌صورت ساختارهایی احتمالی طراحی نموده‌ایم. ساختار meme طراحی شده در شکل (۳) آورده شده است.

تعداد تغییراتی که در BRAMS داده می شود، تعداد خانه‌هایی از کاراکترهای BRAMS است که تغییر می‌کنند. در خانه‌های BRAMS مشخص می‌شود که یک المان از قانون قرار است به چه مقداری تغییر کند. این مقادیر به صورت زیر هستند:

{Don’t care, Small, Medium-Small, Medium, Medium-Large, Large}

یک BRAMS دارای یک آرایه ۴۱ خانه‌ای است، که مشخص می‌کند این meme چند خانه را تغییر می‌دهد. آرایه P، آرایه‌ای ۶ درایه‌ای از مقادیر اعداد حقیقی بین ۰ و۱ است و هر درایه آن مشخص‌کننده احتمال تغییر مقدار یک خانه BRAMS به یکی از ۶ مقدار بالا است. مثلا اگر مقدار درایه ۱ آرایه P برابر با ۰.۲ باشد یعنی احتمال تغییر خانه‌ای از BRAMS به مقدار Large برابر ۰.۲ است.

با این ساختار، استخری از ۳۰ meme که هر meme یک روش جستجوی محلی ساده است، ایجاد کردیم تا در ادامه از آنها استفاده کنیم. هم‌چنین امتیاز هر meme در ابتدا صفر است و همه meme ها مشابه یکدیگر هستند؛ ولی در طی مراحل اجرای برنامه و حل مسأله، memeها براساس نحوه عملکردشان امتیازات متفاوتی بدست می‌آورند.

۵- نتایج اجرای عملی

در این پژوهش الگوریتم ارائه شده را بر روی مجموعه داده تشخیص نفوذ KDD Cup 1999 اعمال نمودیم. هر شئ در این مجموعه داده یک ارتباط شبکه را نشان می‌دهد. هر شئ در یک فضای ۴۱ بُعدی تعریف شده است و متعلق به یکی از ۵ کلاس: نرمال، prob، denial of service (DoS)، unauthorized access to root (U2R) و یا unauthorized access from root machine (R2L) می‌باشد.

شکل (۳): ساختارmeme طراحی شده برای حل مسائل.

جدول (۲): مشخصات پارامترها درپیاده‌سازی کامپیوتری.

پارامتر	مقدار
اندازه جمعیت (N_{pop})	۱۰۰
نرخ جایگزینی مقدار غیرمهم (P_{DC})	۰.۹
احتمال ترکیب (P_X)	۰.۹
احتمال جهش (P_M)	۰.۴
وزن مثبت در محاسبه شایستگی (W_P)	۰.۱
وزن منفی در محاسبه شایستگی (W_N)	۰.۹
حداکثر تغییرات مجاز بر روی قانون اولیه	۵
درصد جایگزینی (P_R)	۱۰
حداکثر تعداد نسل‌ها در الگوریتم اصلی	۵۰

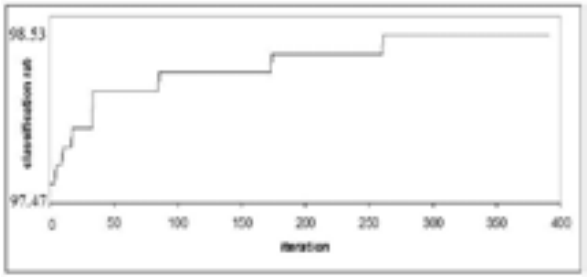
اشیائی که در کلاس نرمال قرار دارند ارتباطات شبکه‌ای بی‌خطر محسوب می‌گردند، در صورتی‌که اشیای متعلق به چهار کلاس دیگر همگی انواع مختلفی از حمله‌های شبکه هستند. مجموعه آموزشی شامل ۴۹۴،۰۲۱ ارتباط و تعداد داده‌های تستی ۳۱۱،۰۲۹ است.

مجموعه داده KDD Cup 1999 تنها مجموعه داده در دسترس و با مقیاس بزرگ است که برای ارزیابی ابزارهای تشخیص نفوذ مورد استفاده قرار می‌گیرد. شرح مبسوطی از این مجموعه داده در [13] موجود است. ما زیرمجموعه‌ای ۱۰٪ ای از مجموعه داده KDD Cup 1999 را بعنوان مجموعه داده آموزشی در نظر گرفته‌ایم. مجموعه داده تست نیز همان مجموعه داده‌ای است که در رقابت‌های KDD Cup 1999 برای ارزیابی الگوریتم‌های کلاس‌بندی مورد استفاده قرار می‌گیرد. در این پژوهش مجموعه داده‌های آموزشی و تست نرمالیزه شده‌اند به این معنا که هر مقدار عددی در مجموعه داده به عددی بین ۰.۰ تا ۱.۰ نرمالیزه شده است. جدول (۲) مشخصات پارامترها را در پیاده‌سازی EFS-MA نشان می‌دهد.

برای حل این مسأله دسته‌های مختلف الگوریتم ژنتیک را بکار برده‌ایم. این دسته‌ها را به تفکیک در جدول (۱) آورده‌ایم. باید توجه داشت که در نمودارهای ارائه شده برای نرخ دسته‌بندی، نمودارها را از مقدار ۹۷.۴۷٪ رسم نموده‌ایم؛ زیرا این مقدار با الگوریتم ژنتیک متعارف نیز قابل حصول است اما الگوریتم‌های ممتیک این مقدار را تا مقادیر نمایش داده شده، افزایش می‌دهند.

۵-۱- روش ایستا

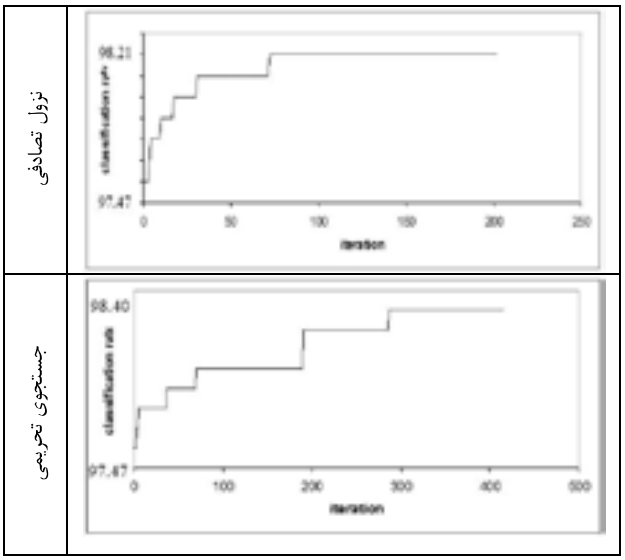
در این روش در هر گام انتخاب، memeها به‌صورت تصادفی انتخاب می‌شوند. در این شرایط همه memeها شانس یکسانی برای انتخاب شدن دارند. حُسن این روش در آن است که به همه شانس انتخاب شدن داده می‌شود ولی امکان تطبیق انتخاب memeها با مسأله را نداریم. نرخ دسته‌بندی که با اجرای این الگوریتم به آن رسیده‌ایم را در شکل (۴) آمده است.



شکل (۴): نرخ دسته‌بندی در روش ایستا.

۵-۲- روش تطبیقی کیفی

روش تطبیقی کیفی از لحاظ سطح تطبیق شامل دو سطح محلی و سراسری است. در سطح محلی با پیاده‌سازی الگوریتم نزول تصادفی در هر گام انتخاب meme، اگر meme مرحله قبل بخوبی عمل کرده و تابع برازش را ارتقا داده باشد، در این مرحله همان meme را انتخاب می‌کنیم. در غیر این صورت یک meme به صورت تصادفی انتخاب می‌شود. در سطح سراسری با پیاده‌سازی الگوریتم جستجوی تحریمی در هر گام انتخاب، memeها را به‌صورت تصادفی انتخاب می‌کنیم اما memeهایی که تابع برازش را ارتقا ندهند، در یک لیست تحریم قرارداده و از آنها شانس انتخاب شدن را می‌گیریم. نتایج این دسته از الگوریتم‌ها در شکل (۵) نشان داده شده است.



شکل (۵): نرخ دسته‌بندی الگوریتم ممتیک در روش تطبیقی کیفی.

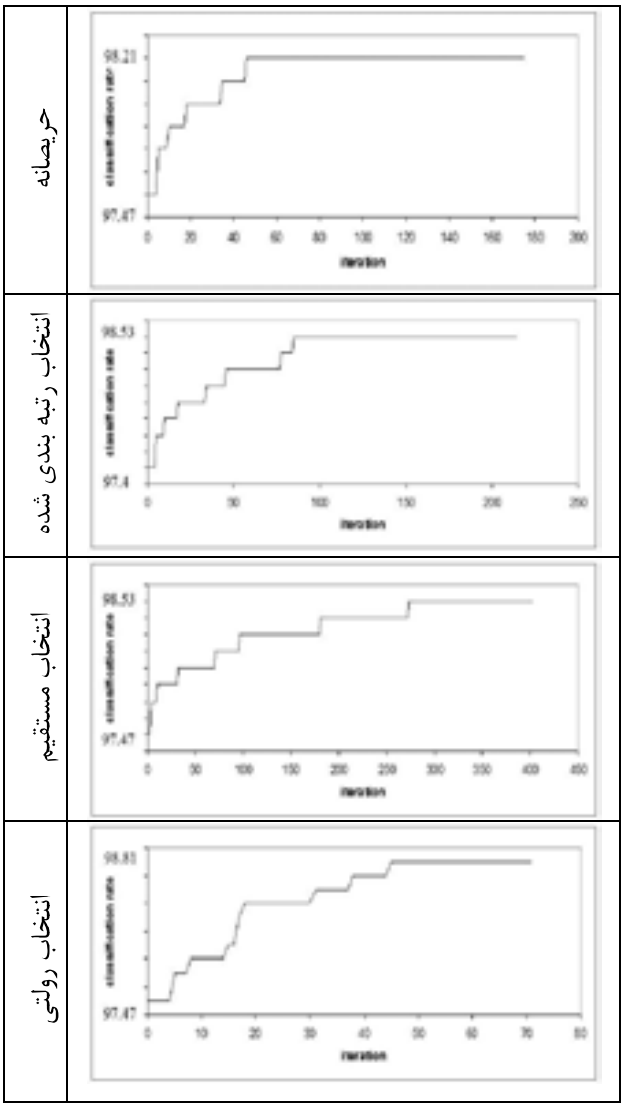
۵-۳- روش تطبیقی کمّی

در این روش برای انتخاب memeها از امتیازی که به آنها می‌دهیم، استفاده می‌شود. تابع امتیازی که برای امتیازدهی به memeها استفاده کرده‌ایم، مجموع تغییرات تابع برازش در هر بار اعمال یک meme است. با استفاده از این تابع امتیازدهی در هر مرحله انتخاب meme، با دانستن امتیاز memeها آنها را انتخاب می‌کنیم. در شکل (۶) نرخ دسته‌بندی این الگوریتم‌ها را می‌بینیم.

در جدول (۳) دسته‌های مختلف الگوریتم ممتیک را از لحاظ نرخ دسته‌بندی وسرعت مقایسه کرده‌ایم. برای مقایسه سرعت، از تعداد تکرارها برای رسیدن به نرخ دسته‌بندی که تمامی الگوریتم‌ها به آن رسیده‌اند، استفاده شده است.

۶- مقایسه نتایج بدست آمده از الگوریتم ممتیک با سایر روش‌های دسته‌بندی

برای مقایسه نتایج کار خود از الگوریتم ممتیک تطبیقی کمّی انتخاب رولتی استفاده کرده‌ایم. کارایی دسته‌بندی EFS-MA (ترکیب روش فازی با الگوریتم ممتیک) برای هر یک از کلاس‌های مسأله تشخیص نفوذ، اندازه‌گیری شده و با کارایی دسته‌بندی‌های پایه مختلف از جمله Naive Bayes (NB)، pruning C4.5، *k*-Nearest Neighbor (*k*- NN) و Support Vector Machine (SVM) مقایسه شده است.



شکل (۶): نرخ دسته‌بندی الگوریتم ممتیک در روش تطبیقی کمّی.

جدول (۳): مقایسه انواع الگوریتم های ممتیک.

الگوریتم	نرخ دسته بندی (CR)	تکرار های هر الگوریتم تا رسیدن به CR پایه
تصادفی ساده	۹۸.۵۳	۹۰
نزول تصادفی	۹۸.۲۱	۷۲
جستجوی تحریمی	۹۸.۴۳	۱۹۱
حریصانه	۹۸.۲۱	۴۶
انتخاب رتبه بندی شده	۹۸.۵۳	۵۰
انتخاب مستقیم	۹۸.۵۳	۹۷
انتخاب رولتی	۹۸.۸۱	۱۷

در k-NN پارامتر k مقدار ۵ را داشته است و SVM با استفاده از متد شناخته شده "حداقل بهینه سازی سریع متوالی" (Fast sequential minimal optimization) و با یک هسته چندجمله‌ای مورد آموزش قرار گرفته است. جدول (۴) نتایج محاسبه *یادآوری*، *دقت* و *مقدار F* را برای دسته‌بندی‌های ذکر شده نشان می‌دهد. طبق این جدول، سیستم ترکیبی ارائه شده در این پژوهش برای اکثر کلاس‌های مسأله تحت بررسی، جزو ۳ روش دسته‌بندی برتر است. بنابراین می‌توان نتیجه گرفت که EFS-MA روشی مطمئن در ایجاد سیستم‌های کلاس‌بندی با کارایی بالا محسوب می‌شود.

جدول (۴): یادآوری، دقت و مقدار F برای چند دسته‌بندی مختلف.

Class	Algorithm	EFS-MA	EFS-AIS	SVM	5-NN	NB	C4.5
NORMAL	یادآوری	۹۷.۲	۹۶.۷	۹۷.۹	۹۵.۸	۵۵.۴	۹۸.۳
	دقت	۷۵.۱	۷۵.۷	۷۳.۴	۷۴.۱	۴۳.۳	۷۴.۷
	معیار F	۸۴.۷	۸۴.۹	۸۳.۹	۸۳.۶	۴۸.۶	۸۴.۹
PRB	یادآوری	۸۸.۹	۸۷.۵	۸۶.۲	۸۱.۶	۹۰.۴	۸۱.۸
	دقت	۷۱.۳	۷۰.۲	۷۷.۷	۵۵.۴	۶۴.۱	۵۲.۲
	معیار F	۷۹.۱۳	۷۷.۹	۸۱.۷	۶۶	۷۵	۶۳.۷
DOS	یادآوری	۹۸.۱	۹۶.۴	۹۷.۵	۹۷	۸۲.۷	۹۶.۶
	دقت	۹۹.۹	۹۹.۹	۹۹.۸	۹۹.۴	۹۴	۹۹.۶
	معیار F	۹۸.۹	۹۸.۱	۹۸.۷	۹۸.۱	۸۸	۹۸.۳
U2R	یادآوری	۲۳.۴	۲۲.۵	۱۰	۱۴.۹	۱۳.۱	۱۴.۴
	دقت	۳.۵	۳.۳	۵۳.۴	۵.۴	۲	۹.۳
	معیار F	۶.۰۸	۵.۸	۱۶.۹	۸	۳.۵	۱۱.۳
R2L	یادآوری	۳۳.۵	۲۹.۹	۳.۵	۶.۹	۶۲.۷	۱.۴
	دقت	۸۰.۸	۹۳.۱	۶۲.۳	۶۶.۹	۴۲.۷	۳۰.۳
	معیار F	۴۷.۳۶	۴۵.۲	۶.۷	۱۲.۵	۵۰.۸	۲.۷

۷- نتیجه

در این مقاله، دسته‌های مختلف الگوریتم‌های ممتیک مورد مطالعه قرار گرفته و با استفاده از روش یادگیری متالامارکی، الگوریتم یادگیری دو مرحله‌ای نوین برای استخراج قوانین فازی کلاس‌بندی ارائه شده است. سیستم یادگیری ارائه شده، یک سیستم فازی تکاملی را با الگوریتم‌های ممتیک جهت جستجوی محلی در هر نسل از الگوریتم ژنتیک و ارتقای والدین، ترکیب و سیستم کلاس‌بندی مناسبی برای تشخیص حمله‌های خرابکارانه پیدا می‌نماید. قابلیت‌های سیستم فازی ترکیبی حاصل با استفاده از مجموعه داده شناخته شده KDD Cup 1999 مورد بررسی قرار گرفته است. با اجرای پیاده‌سازی‌ها بر روی این مجموعه داده، به این نتیجه رسیدیم که افزایش مرحله جستجوی محلی مبتنی بر الگوریتم ممتیک به الگوریتم اصلی، روند یادگیری سیستم فازی تکاملی اولیه را به‌طور مؤثری افزایش می‌دهد. بعلاوه، نتایج نشان می‌دهند که در مقایسه با چند تکنیک سنتی مانند C4.5، Naïve Bayes، k-NN و SVM، راهکار ترکیبی ارائه شده برای اکثر کلاس‌های مسأله کلاس‌بندی تشخیص نفوذ به دقت‌های بهتری دست پیدا می‌کند. بنابراین، قوانین دسته‌بندی فازی حاصل می‌توانند در ایجاد یک سیستم مطمئن تشخیص نفوذ مورد استفاده قرار بگیرند.

بررسی ترکیب‌های روش‌های فرامکاشفه‌ای دیگر مانند بهینه‌سازی از طریق الگوریتم ممتیک خودتطبیق و یا روش تجزیه به زیرمسائل و تأثیر آنها بر کارایی سیستم‌های فازی تکاملی می‌تواند موضوع جالبی برای تحقیقات آینده باشد. علاوه بر آن، استفاده از سیستم‌های فازی تکاملی چندمنظوره (multi-objective) برای بدست آوردن یک روش دسته‌بندی جامع برای مسائل کلاس‌بندی موضوع تحقیقی جالب دیگری است.

تشکر و قدردانی

این تحقیق با حمایت مرکز تحقیقات مخابرات ایران انجام پذیرفته است.

مراجع

[1] Hart, W. E., *Adaptive Global Optimization With Local Search*, Ph.D. Thesis, University of California, San Diego, 1994.

[2] Krasnogor, N., *Studies on the Theory and Design Space of Memetic Algorithms*, Ph.D. Thesis, University of the West of England, Bristol, U.K., 2002.

[3] Cowling, Peter, Kendall, Graham, Soubeiga, Eric, *"A Hyperheuristic Approach to Scheduling a Sales Summit"*, in *PATAT 2000*, Springer Lecture Notes in Computer Science, Konstanz, Germany, pp. 176–190, August 2000.

[4] Eiben, A. E., Hinterding, R., Michalewicz, Z., *"Parameter Control in Evolutionary Algorithm"*, IEEE Transactions on Evolutionary Computation, Vol. 3, pp. 124–141, July 1999.

[5] Hinterding, R., Michalewicz, Z., Eiben, A. E., *"Adaptation in Evolutionary Computation: A Survey"*, IEEE International Conference on Evolutionary Computation, Piscataway, NJ: IEEE Press, pp. 65–69, April 1997.

[6] Krasnogor, N., Blackburne, B., Hirst, J. D., and Burke, E. K. N., *"Multimeme Algorithms for the Structure Prediction and Structure Comparison of Proteins"*, in Parallel Problem Solving From Nature, Lecture Notes in Computer Science, 2002.

[7] Krasnogor, N., Gustafson, S., *"A Study on the Use of "Self-generation"in Memetic Algorithms"*, Natural Computing, Vol. 3, No. 1, pp. 53–76.

[8] Krasnogor, N., Smith, J., *"Multimeme Algorithms for the Structure Prediction and Structure Comparison of Proteins"*, 2004.

[9] Ong, Y. S., Lim, M. H., Zhu, N., and Wong, K. W., *"Classification of Adaptive Memetic Algorithms:A Comparative Study"*, IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics, Vol. 36, No. 1, pp. 141-152, February 2006.

[10] Ong, Yew Soon, Keane, Andy J., *"Meta-Lamarckian Learning in Memetic Algorithms"*, IEEE Transactions on Evolutionary Computation, Vol. 8, No. 2, pp. 99-110, April 2004.

[11] Smith, J. E., et al., *"Co-evolution of Memetic Algorithms: Initial Investigations"*, Parallel Problem Solving From Nature—PPSN VII, Springer, Lecture Notes in Computer Science, Vol. 2439, pp. 537–548, 2002.

[12] Smith, J. E., *"Co-evolving Memetic Algorithms: A Learning Approach to Robust Scalable Optimization"*, IEEE Congress on Evolutionary Computation. Piscataway, NJ: IEEE Press, Vol. 1, pp. 498–505, 2003.

[13] KDD-cup data set, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.