



ارائه روشی مبتنی بر الگوریتم‌های ژنتیکی برای تعیین موقعیت بهینه عامل‌های سیار در کاربرد نظارت بر کارایی شبکه‌های فعال

مهدی دهقان
دانشکده مهندسی کامپیوتر،
دانشگاه صنعتی امیرکبیر
dehghan@ce.aut.ac.ir

حسین پدرام
دانشکده مهندسی کامپیوتر،
دانشگاه صنعتی امیرکبیر
pedram@ce.aut.ac.ir

امیرحسین حداد
دانشکده مهندسی کامپیوتر،
دانشگاه صنعتی امیرکبیر
a_haddad@itrc.ac.ir

انجام اعمال مدیریتی در آنها را به امری پیچیده و دشوار تبدیل کرده است. به دلیل مسائلی از این قبیل ایده شبکه‌های فعال در سال ۱۹۹۷ مطرح گردید [1]. شبکه‌های فعال به شبکه‌هایی اطلاق می‌گردد که در آنها نودهای میانی از قبیل مسیریابها، همانند نودهای پایانی دارای قدرت پردازشی برای اجرای برنامه‌های کاربردی باشند.

از سوی دیگر، مزایای بسیار زیادی برای استفاده از عامل‌های سیار در مدیریت شبکه ذکر گردیده است. از جمله این مزایا می‌توان به استفاده مؤثر از منابع، کاهش ترافیک شبکه، تقابل با نودها به صورت بلادرنگ اشاره کرد [2]-[4]. همچنین یک سیستم مدیریتی که از عامل‌های سیار برای انجام فرآیندهای مدیریتی استفاده می‌نماید، دارای یک ساختار توزیع شده است. این امر مقیاس‌پذیری و قابلیت انعطاف سیستم مدیریتی را تا حد زیادی افزایش می‌دهد [3]، [4].

با مطرح شدن ایده شبکه‌های فعال گامی بزرگ در راستای استفاده مؤثرتر از عامل‌های سیار در مدیریت شبکه برداشته شد. تعدد و تنوع مقالات ارائه شده برای استفاده از عامل‌های سیار در زمینه‌های مختلف مدیریت شبکه‌های فعال گویای این امر است [5]-[9]. با وجود مزایای ذکر شده برای استفاده از عامل‌های سیار در مدیریت شبکه فعال، محدودیتهایی نیز مانند محدودیت قدرت پردازشی در نودهای فعال وجود دارد [10]-[12]. در نتیجه نودهای فعال دارای توانی محدود برای میزبانی عامل‌های سیار و اجرای اعمال مدیریتی آنها می‌باشند.

در [4] از عامل‌های سیار برای انجام فرآیند نظارت بر کارایی شبکه استفاده شده است. هدف اصلی آن تعیین موقعیت عامل‌های نظارت کننده بر روی نودهای پایانی یک شبکه مقیاس بالا بوده است. از آنجایی که شبکه غیر فعال در نظر گرفته شده است محدودیت پردازشی نودهای پایانی لحاظ نشده است. در این مقاله

چکیده

نظر به گسترش روز افزون نیاز به پردازش در سطح شبکه‌ها طی چند سال اخیر ایده شبکه‌های فعال مطرح گردیده و امکان میزبانی عامل‌های سیار در نودهای میانی از قبیل مسیریابها میسر شده است. این مقاله به بررسی یک روش پیشنهادی برای انجام فرآیند نظارت بر کارایی در شبکه‌های فعال با مقیاس بالا توسط عامل‌های سیار می‌پردازد. در روش پیشنهادی زیرمجموعه‌ای از نودهای فعال برای انجام فرآیند نظارت بر کارایی شبکه توسط عامل‌های سیار انتخاب می‌شوند. معیار انتخاب مجموعه‌ای از محدودیتها از قبیل توان پردازشی نودهای فعال و حجم ترافیک مورد نظارت می‌باشد. پیدا کردن محل عاملها در قالب مساله p سکوی میانی ظرفیتی که از دسته مسائل موقعیت تسهیلات به حساب می‌آید، طرح گردیده است. به دلیل فضای جستجوی وسیع و محدودیت‌های متعدد این مساله، برای حل آن از روشی مبتنی بر الگوریتم‌های ژنتیکی استفاده شده است. روش پیشنهادی بر روی \mathcal{E} شبکه با هم بندی و تعداد نودهای متفاوت شبیه سازی شده است. نتایج به دست آمده نشان می‌دهد که این روش احتمال استفاده بیشتر از حد مجاز، از توان پردازشی نودهای فعال میزبان عامل‌های سیار را کاهش می‌دهد و منجر به کاهش زمان یافتن موقعیت بهینه برای عامل‌های سیار می‌گردد.

کلمات کلیدی: شبکه‌های فعال، عامل‌های سیار، نظارت بر کارایی، مسائل p سکوی میانی، الگوریتم‌های ژنتیک.

۱- مقدمه

به دلیل نیاز گسترده و روز افزون به پردازش اطلاعات در شبکه‌های کامپیوتر، استفاده از توان پردازشی نودهای پایانی در شبکه کافی به نظر نمی‌رسد. از طرف دیگر افزایش تعداد نودهای شبکه‌های با مقیاس بالا، به روز کردن پروتکل‌های ارتباطی و نیز

$$\text{Min} \sum_{i=1}^n \sum_{j=1}^n a_i d_{ij} x_{ij} \quad (1)$$

با در نظر گرفتن محدودیت‌های زیر:

$$\sum_{j=1}^n x_{ij} = 1, \quad i = 1, 2, \dots, n \quad (2)$$

$$x_{ij} \leq y_j, \quad i = 1, 2, \dots, n \quad (3)$$

$$\sum_{j=1}^n y_j = p, \quad (4)$$

$$x_{ij}, y_j \in \{0, 1\}, \quad i, j = 1, 2, \dots, n \quad (5)$$

که در آن:

n : تعداد کل نودهای گراف

a_i : میزان درخواست برای گره i

d_{ij} : فاصله نود i از j

p : تعداد تسهیلاتی که به عنوان سکوی میانی انتخاب شده‌اند

a_i, d_{ij} اعداد مثبت و حقیقی هستند.

$$x_{ij} = \begin{cases} 1 & \text{if vertex } i \text{ is assigned to facility } j \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

$$y_j = \begin{cases} 1 & \text{if vertex } j \text{ chosen as facility} \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

تفاوت مسائل p سکوی میانی ظرفیتی با مسائل p سکوی میانی اضافه شدن شرط محدودیت در ظرفیت سکوهای میانی می‌باشد که آنها را برای تعریف مساله مورد بررسی در این مقاله مناسب‌تر می‌گرداند، زیرا خصوصیت سکوهای میانی در آنها به نودهای فعال که توان پردازشی محدودی برای میزبانی عامل‌های سیار و انجام فرآیند نظارتی دارند نزدیک تر است. هدف این مساله کمینه کردن ترافیک حاصل از نظارت کارآیی شبکه ارسال شده توسط عامل‌های سیار به ایستگاه مدیریتی مرکزی است. ترافیک حاصل از فرآیند نظارت به سه دسته کلی تقسیم می‌شود:

- ترافیک ارسال شده از نودهای مورد نظارت به عامل سیار مسئول نظارت آنها که با پارامتر $Traff$ نمایش داده می‌شود.
 - ترافیک ارسال شده از هر یک از عامل‌های سیار به ایستگاه مدیریتی مرکزی که با پارامتر $rtTraff$ نمایش داده می‌شود.
 - ترافیک حاصل از ارسال عامل‌های سیار به سکوهای میانی نظارتی که با پارامتر $MaTraff$ نمایش داده می‌شود.
- به این ترتیب مساله را می‌توان در قالب عبارات برنامه‌نویسی عدد صحیح به صورت زیر بیان کرد:

$$\text{Min} \left(\sum_{i=1}^n \sum_{j=1}^n Traff_{ij} d_{ij} x_{ij} + \sum_{j=1}^p d_j (rtTraff_j + MaTraff_j) \right) \quad (8)$$

با در نظر گرفتن محدودیت‌های زیر علاوه بر محدودیت‌های (۲) - (۵):

$$Load_i < Threshold \quad i = 1, 2, \dots, n \quad (9)$$

ما کار انجام شده در [4] برای استفاده در شبکه‌های فعال توسعه داده‌ایم و محدودیت پردازشی نودهای فعال برای میزبانی عامل‌های سیار را نیز به محدودیت‌های مساله افزوده‌ایم.

در ادامه راه حل پیشنهادی مساله یافتن موقعیت عامل‌های سیار توسط یک الگوریتم ژنتیک برای شبکه‌های فعال در دو حالت، با در نظر گرفتن محدودیت پردازشی نودهای فعال و بدون در نظر گرفتن این محدودیت ارائه شده است. علت اصلی استفاده از الگوریتم‌های ژنتیکی در حل این مساله فضای وسیع حالات در شبکه‌های مقیاس بالا برای رسیدن به راه حل مناسب، محدودیت‌های متنوعی که به این مساله اضافه گردیده، و پیچیدگی استفاده از روش‌های قطعی در حل آن می‌باشد.

۲- مساله موقعیت عامل‌های سیار در شبکه فعال

یک ایستگاه مدیریتی مرکزی را در نظر بگیرید که تصمیم دارد عامل‌های سیار را برای نظارت بر کارآیی نودهای شبکه فعال ارسال نماید. زیر مجموعه‌ای از نودهای فعال برای میزبانی عامل‌های سیار توسط این ایستگاه انتخاب می‌گردند. این ایستگاه با توجه به در دست داشتن اطلاعات مربوط به میانگین بار پردازشی در هر نود شبکه فعال، نودهای میزبان عامل‌های سیار را به گونه‌ای انتخاب می‌کند که:

- ترافیک حاصل از فرآیند جمع‌آوری و ارسال نتایج نظارت بر کارآیی شبکه توسط عامل‌های سیار کمینه گردد.
- نودهایی انتخاب گردند که بار پردازشی آنها بعد از میزبانی عامل سیار از یک حد آستانه بالاتر نرود.

فعال در نظر گرفتن نودهای شبکه با محدودیت توان پردازشی موجب می‌گردد مساله یافتن میزبان‌های عامل‌های سیار در دسته تغییر یافته مسائل p سکوی میانی، با عنوان مسائل p سکوی میانی ظرفیتی^۱ قرار گیرد [13]. برای شرح مساله ابتدا مساله p سکوی میانی در قالب عبارات برنامه‌نویسی عدد صحیح^۲ ارائه گردیده است [14]. نکته قابل توجه در این بیان ریاضی، امکان در نظر گرفتن همزمان نودها هم به عنوان درخواست کننده و هم به عنوان تسهیلات می‌باشد. از این نکته می‌توان برای مساله مورد بررسی در این مقاله استفاده کرد، زیرا نودهای فعال شبکه در عین حال که باید نظارت شوند، ممکن است یک جایگاه برای میزبانی عامل‌های سیار و نظارت نودهای دیگر شبکه باشند.

مساله p سکوی میانی:

¹ Capacitated p-median problem
² Integer Programming

$$Traff_{ij} > rtTraff_j > MaTraff_j \quad (10)$$

که در آن:

$Traff_{ij}$: اندیس نود نظارت شونده و i اندیس نود میزبان عامل سیار،

$rtTraff_i$: اندیس نودی که عامل سیار روی آن قرار دارد،

$MaTraff_i$: اندیس نودی که عامل سیار روی آن قرار دارد،

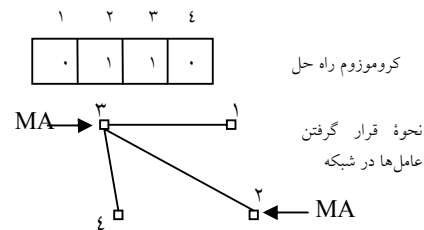
$Load_i$: بار پردازشی نود i بعد از شروع فرآیند نظارت بر کارایی.

$Threshold$: عدد ثابت حد آستانه مجاز برای بار پردازشی نودهای فعال بر حسب درصد.

نحوه محاسبه این پارامترها متناسب با نوع فرآیند نظارت بر کارایی می تواند تغییر کند. در بخش شبیه سازی یک نحوه محاسبه این پارامترها متناسب با نوع نظارت بر کارایی مفروض ذکر شده است.

۳- روش پیشنهادی برای تعیین محل بهینه میزبان های عامل سیار

ما در روش پیشنهادی از الگوریتم ژنتیک برای تعیین محل بهینه میزبان های عامل سیار استفاده می نماییم. راه حل در این الگوریتم، یک کروموزوم در قالب رشته بیتی است که تعیین کننده محل میزبان های عامل سیار است. طول این رشته بیتی برابر تعداد نودهای گراف نمایانگر توپولوژی شبکه می باشد. بتهای یک کروموزوم نودهای میزبان عامل سیار را مشخص می نمایند. در اینجا فرض شده است که هر عامل سیار تنها قادر به نظارت بر کارایی نود میزبان و نودهای همسایه آن در ماتریس همبندی شبکه می باشد. ساختار کد گذاری هر کروموزوم مشابه ساختار مورد استفاده در [17] می باشد. در شکل ۱ نمونه کروموزوم و چگونگی تعیین محل میزبان های عامل های سیار توسط آن را مشاهده می کنید.



شکل ۱: نحوه تعیین محل عامل های سیار توسط یک کروموزوم

راه حل پیشنهادی برای دو حالت بیان شده است: حالتی که تنها هدف کمینه کردن ترافیک نظارتی است و حالتی که هدف کمینه کردن ترافیک نظارتی است با رعایت این محدودیت که بار پردازشی هیچ نودی از یک حد آستانه تعریف شده بیشتر نگردد. مفروضات در نظر گرفته شده برای حل مساله به شرح زیر هستند:

- هر عامل سیار تنها قادر به نظارت بر کارایی نودهای همسایه خود در گراف مجاورت شبکه است.
- تمامی نودها باید توسط عامل های سیار نظارت شوند.

۳-۱- پارامترهای الگوریتم ژنتیک

جهش کروموزومها

در این مقاله بعد از آزمایش الگوریتم ژنتیک برای نرخ های مختلف جهش، بهترین جوابها با استفاده از نرخ جهش ۰,۰۳ به دست آمد. جداول نتایج شبیه سازی ارائه شده در بخش بعد با استفاده از همین نرخ جهش به دست آمده اند.

تقاطع کروموزومها

روشهای تقاطع متنوعی برای حل این مساله مورد آزمایش قرار گرفتند، که در نهایت تقاطع دو نقطه ای^۳ به نتیجه های بهتری جوابها منجر شد.

استفاده از مهاجرت

در روش پیشنهادی از ایده مهاجرت نیز استفاده شده است. تعداد زیر جمعیتها ۲ در نظر گرفته شده است و بعد از هر ۲۰ نسل بهترین کروموزوم از هر یک از زیر جمعیتها جایگزین فردی در زیر جمعیت دیگر می گردد.

تابع سازگاری^۴

توابع سازگاری متناسب با مساله محل میزبان های عامل سیار که در فصل قبل ذکر شد، یکبار بدون در نظر گرفتن محدودیت (۹) از این به بعد به آن تابع سازگاری نوع اول اطلاق می گردد، و یکبار با در نظر گرفتن آن (از این به بعد به آن تابع سازگاری نوع دوم اطلاق می گردد) تعریف گردیده اند.

تابع سازگاری نوع اول

$$Fit1 = Ma2RootTraff / MaxTraff + \quad (11)$$

$$Nodes2MaTraff / MaxTraff + OverlapRate$$

تابع سازگاری نوع دوم

$$Fit2 = Ma2RootTraff / MaxTraff + \quad (12)$$

$$Nodes2MaTraff / MaxTraff +$$

$$OverlapRate + OverloadedNodes / N$$

که در آنها:

$Ma2RootTraff$: ترافیک های ارسالی از عامل های سیار به ایستگاه

مدیریتی مرکزی،

$Nodes2MaTraff$: ترافیک های ارسالی از نودهای تحت نظارت به

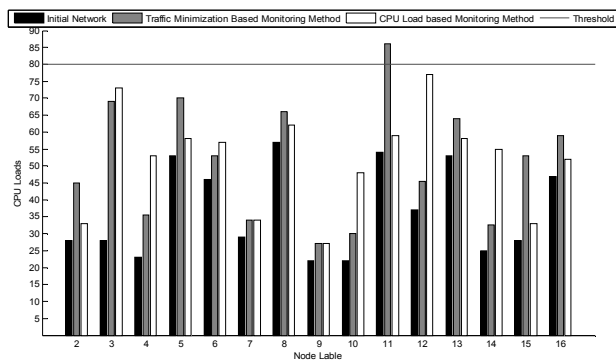
عامل های سیار،

³ Two Point Crossover

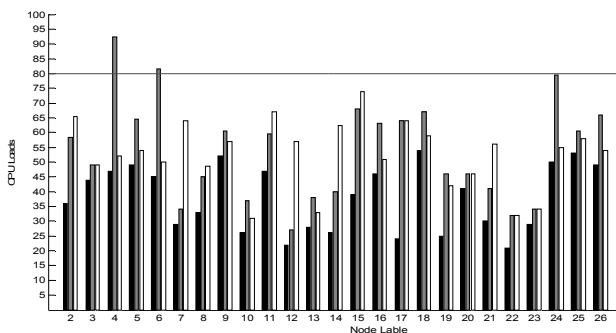
⁴ Fitness function

می‌گردند. اندازه هر پارامتر ۱۶ بایت در نظر گرفته شده است که بدین ترتیب، هر نود میزان ۱۶۰ کیلو بایت به عامل سیار مسئول نظارت خود ارسال می‌کند. میزان ترافیک ارسال شده از عامل‌های سیار به ایستگاه مرکزی مدیریت بازی هر نود که مسئول نظارت آن هستند ۱،۶ کیلو بایت به دست می‌آید که حجم ترافیک اطلاعات جمع بندی شده ۱۰۰ پارامتر ۱۶ بایتی در بازه زمانی ۱۰۰ دقیقه نظارت بر کارایی است. اندازه عامل سیار در شبیه سازی بر اساس اندازه عامل‌های سیار در چهار چوب ساخت Grasshopper ۰،۵ کیلو بایت فرض شده است [18].

در شبکه‌های آزمایشی بار پردازشی مصرفی نودهای شبکه‌ها به صورت تصادفی اعدادی بین ۲۰ تا ۶۵ درصد در نظر گرفته شده‌اند. بازی نظارت بر هر نود توسط نود فعال میزبان عامل سیار ۵٪ به بار پردازشی مصرف شده آن اضافه گردیده، بعلاوه میزبانی عامل سیار منجر به مصرف ۵٪ دیگر از توان پردازشی نود می‌شود. به عنوان نمونه، شکل ۳ نمودار بار پردازشی هر یک از نودهای شبکه فعال پس از میزبانی عامل‌های سیار در شبکه‌های آزمایشی با ۱۵ و ۲۵ نود را نشان می‌دهد.



(a)



(b)

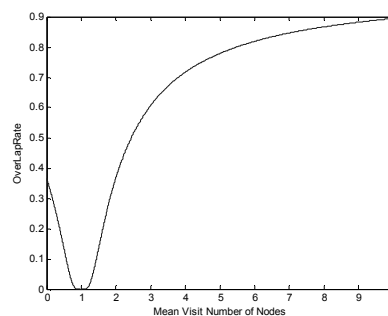
شکل ۳: در (a) و (b) نمودار میله‌ای مربوط به بار پردازشی نودهای ۲ شبکه با هم بندی و تعداد نودهای متفاوت ارائه گردیده است. میله‌ها به ترتیب از چپ به راست نمایانگر موارد روبرو هستند: بار پردازشی مصرفی اولیه، بار پردازشی مصرفی در حالتی که تنها هدف کمینه کردن ترافیک است، بار پردازشی مصرفی در حالتی که محدودیت بار پردازشی در نظر گرفته شود

OverloadedNodes: تعداد نودهایی که بار پردازشی آنها بعد از قرار گرفتن عامل‌های سیار بیش از حد آستانه مجاز است، *MaxTraff*: ترافیک ارسالی از نودهای شبکه به ایستگاه مدیریت مرکزی در حالتی که از عامل‌های سیار استفاده نشود، *OverlapRate*: مقداری که نشان دهنده میزان مناسب بودن موقعیت عامل‌های سیار با توجه به نودهایی است که نظارت می‌شوند. در صورتی که بخشی از نودها پایش نشده باشند، یا بیش از یک بار نظارت شده باشند، این مقدار مانع از کاهش مقدار تابع سازگاری می‌گردد. مقدار این پارامتر براساس فرمول زیر به دست می‌آید:

$$OverlapRate = \exp(-1 / |1 - \sum_{i=1}^N (visitedNodes_i / N)|) \quad (13)$$

visitedNodes_i: تعداد دفعات نظارت شدن نود *i*ام،

N: تعداد نودهای فعال شبکه.



شکل ۴: مقادیر مختلف *OverlapRate* به بازی میانگین‌های مختلف برای دفعات نظارت نودهای فعال. همانطور که در شکل می‌بینید در حالتی که این میانگین ۱ است، یعنی هر نود توسط یک عامل سیار نظارت می‌گردد، مقدار *OverlapRate* کمینه است.

الگوریتم ژنتیک برای هر دو تابع سازگاری اجرا و نتایج آن در فصل بعد ارائه گردیده است.

۴- نتایج شبیه سازی

در این بخش به بررسی نتایج به دست آمده از شبیه سازی الگوریتم ژنتیکی بر روی ۴ شبکه آزمایشی که دارای ۱۵، ۲۵، ۳۵ و ۵۰ نود می‌باشند، می‌پردازیم. توپولوژی شبکه‌ها به صورت تصادفی تولید در قالب ماتریس مجاورت آنها تولید گردیده است. پردازنده سیستمی که شبیه سازی بر روی آن انجام گردیده است، از نوع *Centrino 1.5GHz* با *512MByte* حافظه می‌باشد. این شبیه سازی‌ها با استفاده از نرم‌افزار *Matlab* انجام گردیده‌اند. در این شبیه سازی فرض شده است که یک فرآیند نظارتی برای اندازه‌گیری میانگین ۱۰۰ پارامتر کارایی در طول مدت ۱ ساعت و ۴۰ دقیقه، توسط عامل‌های سیار برای کل نودهای شبکه انجام می‌گردد. در این فرآیند هر ۱ دقیقه اطلاعات کارایی جمع آوری

نزدیک به بهینه در مساله موقعیت عامل‌های سیار برای الگوریتم ژنتیکی یک مزیت به حساب می‌آید.

در ستونی ترافیک نظارتی بیشینه، مقادیر بیشینه ترافیک در صورت عدم استفاده از عامل‌های سیار برای هر یک از شبکه‌ها محاسبه گردیده است. همانطور که در جدول می‌توان مشاهده نمود میزان ترافیک حاصل از این روش نظارتی که در ستون آخر جدول نمایش داده شده است، برای هر دو تابع سازگاری به طور قابل ملاحظه‌ای کمتر از ترافیک حاصل از نظارت بر کارایی در حالت متمرکز و غیر توزیع شده می‌باشد.

نکته دیگری که در جدول می‌توان مشاهده نمود افزایش تدریجی نودهای نظارت نشده با افزایش تعداد نودهای شبکه می‌باشد. این امر به دلیل افزایش وزن معیار کاهش ترافیک نسبت به معیار نظارت نودها توسط عامل‌های سیار می‌باشد.

جدول ۱: نتایج میانگین ۱۰ بار اجرای الگوریتم پیشنهادی برای دو تابع سازگاری،

نوع اول (GA1) و نوع دوم (GA2)

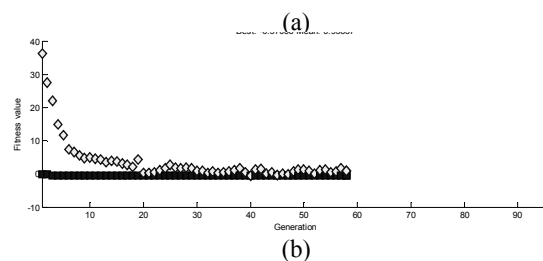
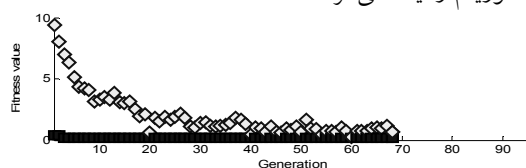
	تعداد نودها	نودهای نظارت نشده %	نودهای دارای بار پردازشی بیش از حد مجاز %	مدت زمان هم‌گرایی (ثانیه)	ترافیک نظارتی بیشینه	ترافیک نظارتی Kbytes
GA1	15	0	12	5.79	6400	2962.1
GA2	15	0	0	4.71	6400	3284.8
GA1	25	0	16	23.64	8320	4959.4
GA2	25	0	0	20.48	8320	5445.1
GA1	35	0	14	45.02	10400	8140
GA2	35	6	0	37.4	10400	9267
GA1	50	0	13	72.06	14720	13263
GA2	50	12	0	57.23	14720	11342

۵- نتیجه گیری و کارهای آینده

در به کار گیری سیستم‌های مبتنی بر عامل‌های سیار در شبکه‌های فعال باید مساله محدودیت پردازشی نودهای فعال را در نظر گرفت. زیرا همانطور که در بخش نتایج نشان داده شد، در نظر گرفتن توان پردازشی نا محدود به خصوص برای نودهای فعال (که وظیفه اصلی آنها اجرای عامل‌ها نیست) منجر به ارائه راه حل‌های غیر واقعی می‌گردد. از سوی دیگر در شبکه‌های مورد بررسی در این مقاله لحاظ کردن این محدودیت منجر به کاهش زمان جستجو

میزان بار پردازشی بر روی هر یک از نودهای شبکه پس از اجرای الگوریتم ژنتیک برای دو تابع سازگاری نوع اول و نوع دوم و تعیین محل عامل‌های سیار در شکل ۳ ارائه گردیده است. خط افقی کشیده شده نمایانگر سطح آستانه‌ای است که حداکثر بار پردازشی مجاز نودهای فعال را نشان می‌دهد. همانطور که در شکل مشخص است، در تمامی شبکه‌های مورد بررسی در صورت لحاظ نکردن محدودیت بار پردازشی، امکان گذشتن از حد آستانه مجاز وجود دارد.

شکل ۴ روند هم‌گرایی روش پیشنهادی را برای رسیدن به جواب قابل قبول در شبکه آزمایشی متشکل از ۱۵ نود با استفاده از دو نوع تابع سازگاری نشان می‌دهد. همانطور که در شکل مشاهده می‌شود، استفاده از تابع سازگاری نوع دوم منجر به هم‌گرایی سریع تر الگوریتم ژنتیک می‌گردد.



شکل ۴: شکل (a) نتایج اجرا با تابع سازگاری نوع اول و شکل (b) نتایج اجرا با تابع سازگاری نوع دوم می‌باشد. در نمودار بالایی شکل‌های (a) و (b) نحوه حرکت میانگین توابع سازگاری برای کل جمعیت (لوزی‌های روشن) و بهترین مقدار توابع سازگاری در یک جمعیت (مربع‌های تیره) نشان داده شده است. تعداد نودهای شبکه در این شبیه سازی ۱۵ عدد بوده است.

در جدول ۱ نتایج آماری مربوط به ۱۰ اجرای الگوریتم برای دو نوع تابع سازگاری ارائه گردیده است. همانطور که در جدول مشهود است، در نظر گرفتن حد آستانه مجاز برای بار پردازشی نودهای فعال، منجر به افزایش ترافیک نظارتی می‌گردد. از طرف دیگر در شبکه‌های مورد بررسی زمان اجرای الگوریتم ژنتیکی برای تابع سازگاری نوع دوم تا حد قابل ملاحظه‌ای نسبت به تابع سازگاری نوع اول کمتر است. دلیل این امر محدودتر شدن جواب‌ها در این حالت است. کوتاه‌تر بودن زمان پیدا کردن جواب

R. Kazi, P. Morreale, "Mobile Agents for Active Network Management", Stevens Institute of Technology, IEEE, 1999.

[8] A. Galis, D. Griffin, W. Eaves, G. Pavlou, S. Covaci, R. Broos ó "Mobile Intelligent Agents in Active Virtual Pipes" ó in "Intelligence in Services and Networks" ó Springer Verlag Berlin Heidelberg, April 2000.

[9] Rumeel Kazi, Patricia Moreale, "Mobile Agents for Active Network Management", Stevens Institute of Technology, IEEE communications surveys, 1999.

[10] Celestine Brou et al. , " Future Active IP Networks (FAIN) ", GMD company, Initial Specification of Case Study Systems, May 2001.

[11] Florian Baumgartner, Torsten Braun, and Bharat Bhargava, iDesign and Implementation of a Python-Based Active Network Platform for Network Management and Controlî, Design and Implementation of a Python-Based Active Network Platform for Network Management and Control, pages 177-190, Springer-Verlag, 2002.

[12] L. Kencl, J. L. Boudec, iAdaptive Load Sharing for Network Processorsî, In Proceedings of INFOCOM 02, IEEE, June 2002.

[13] E. S. Correa, M. T. A. Steiner, A. A. Freitas, C. Carnieri, iA genetic algorithm for solving a capacitated p-median problemî, Numerical Algorithms 35, pages 373ñ388, Kluwer Academic Publishers, 2004.

[14] C. Reville, R. Swain, Central facilities location, Geographical Analysis 2 (1970) 30ñ42.

[15] S. Shepherd, A. Sumali, iA Genetic Algorithm Based Approach to Optimal Toll Level and Location Problemsî, Networks and Spatial Economics (4), pages 161-179, Kluwer Academic Publishers, 2004.

[16] O. Alp, E. Erkut, Z. Drezner, iAn Efficient Genetic Algorithm for the p-Median Problemî, Annals of Operations Research 122, pages 21ñ42, Kluwer Academic Publishers, 2003.

[17] J. H. Jaramillo, J. Bhadury, R. Batta, iOn the use of genetic algorithms to solve location problemsî, Computers & Operations Research 29, pages 761-779, Elsevier Science, 2002.

[18] Grasshopper Development System, Light Edition v2.2.1, Programmer's Guide, IKV++ GmbH, [URL:<http://www.grasshopper.de>], Berlin, 2001.

برای محل میزبان‌های عامل سیار می‌گردد. زیرا با این کار فضای جستجو مساله نیز کاهش می‌یابد.

در راه حل ارائه شده در این مقاله عامل‌های سیار تنها قادر هستند که نودهای اطراف خود را نظارت کنند. این محدودیت از یک طرف منجر به ساده شدن نحوه کد گذاری کروموزوم‌های می‌گردد، اما از سوی دیگر امکان رسیدن به جواب‌های نزدیک به بهینه در مساله را کاهش می‌دهد. یکی از کارهای آینده‌ای که در ادامه این کار انجام خواهیم داد، حل این مساله در حالتی است که عامل‌های سیار چنین محدودیتی ندارند. برای این منظور نیاز به تغییر ساختار کد گذاری کروموزوم‌های استفاده شده در الگوریتم ژنتیکی می‌باشد.

۶- مراجع

[1] D. L. Tennenhouse, J.M. Smith, W. D. Sincoskie, D. J. Wetherall, and G.J. Minden. "A survey of active network research", IEEE Communications Magazine, pages 80 - 86, January 1997.

[2] S. Green, L. Hurst, B. Nangle, P. Cunningham, F. Sommers and R. Evans, "Software Agents: A review", Technical Report, Department of Computer Science, Trinity College, Deblein, Irland, 1998.

[3] D. Chess, C. G. Harrison and A. Kershenbaum, iMobile Agents: Are They a Good Idea?î G. Vigna (ed.), Mobile Agents and Security, LNCS 1419, Springer Verlag, 1998.

[4] Antonio Liotta, iTowards Flexible and Scalable Distributed Monitoring with Mobile Agentsî, Doctor of Philosophy Dissertation, University of London, July 2001.

[5] D. Gavalas, D. Greenwood, M. Ghanbari, and M. O'Mahony, "An infrastructure for distributed and dynamic network management based on mobile agent technology." In Proceedings of the IEEE International Conference on Communications (ICC99), 1362-1366, 1999.

[6] Daniel Rossier, Rudolf Scheurer, "An Ecosystem-inspired Mobile Agent Middleware for Active Network Management", Swisscom Corporate Technology, University of Fribourg, Switzerland, 2003.

[7] Breugst, M. Magedanz, "Mobile Agents ñ Enabling Technology for Active Intelligent Network Implementation", IEEE Network, Vol. 12, No. 3, May 1998.