

## خطایابی در شبکه های حسگر بیسیم با استفاده از یک روش مبتنی بر ترکیب الگوریتم رقابت استعماری و منطق فازی

طیبه توسلی<sup>۱\*</sup>، مسعود ماهرانی<sup>۲</sup>، عبدالرضا رسولی کناری<sup>۳</sup>

۱- کارشناسی ارشد، IT، دانشگاه غیر انتفاعی تعالی، قم

۲- کارشناسی ارشد، نرم افزار، دانشگاه پیام نور واحد بین الملل قشم

۳- دکتری، کامپیوتر، دانشگاه صنعتی، قم

### خلاصه

خطا در شبکه های حسگر بیسیم تاثیر منفی بر روی عملکرد شبکه دارد و منجر به از بین رفتن داده های حساس و همچنین منجر به مشکلات اقتصادی و امنیتی فراوانی می شوند. شناسایی علت وقوع خطا (خطایابی) در شبکه های حسگر بیسیم از اهمیت خاصی برخوردار است. خطاها در شبکه های حسگر بیسیم نیاز به تشخیص و سپس اصلاح دارند تا سلامتی شبکه تضمین گردد. مهمترین نکته در خطایابی در شبکه های حسگر بیسیم، هزینه تست اجزای شبکه است؛ بدین معنی که فرایند تست برای خطایابی و رفع آن باید با کمترین هزینه به انجام برسد. در این پایان نامه یک روش مبتنی بر الگوریتم رقابت استعماری و منطق فازی برای رفع خطاهای موجود در شبکه های حسگر بیسیم معرفی می شود. هدف از ارائه این روش حداقل سازی هزینه های خطایابی است. براساس نتایج بدست آمده از شبیه سازی ها، روش ارائه شده هزینه تست خطایابی را نسبت به روش های موجود حداقل می کند و عملکرد بهتری نسبت به روش های مشابه دیگر در یافتن علت خطاها دارد.

**کلمات کلیدی:** شبکه حسگر بیسیم، خطایابی، تست، هزینه های خطایابی

### ۱. مقدمه

بطور کلی عدم شناسایی به موقع خطاها در شبکه های کامپیوتری، منجر به صدمه دیدن و از بین رفتن مقادیر قابل توجهی از امکانات و اطلاعات خواهد شد. در نتیجه، تمایل روزافزونی در زمینه شناسایی علل خطاها چه در مجامع علمی و چه در مجامع صنعتی ایجاد شده است. به سیستمی که بتواند خطاهای موجود در سیستم (در این مقاله شبکه های حسگر بیسیم) را تشخیص داده و علت وقوع این خطاها را تعیین کند، سیستم خطایابی نامیده می شود [۳].

هر چقدر شبکه های حسگر بیسیم گسترش می یابند، پیچیدگی آن ها نیز افزایش پیدا می کند. از طرفی دیگر سرویس دهی مناسب در این شبکه ها مورد انتظار کاربران شبکه است. با توجه به اهمیت موضوع شناسایی علت خطاها (خطایابی) در شبکه های حسگر بیسیم، در این مقاله یک روش مبتنی بر الگوریتم رقابت استعماری و منطق فازی برای شناسایی مکان خطا در شبکه های حسگر بیسیم ارائه می شود. روش معرفی شده در این مقاله از طریق داده های

\* Corresponding author: کارشناسی ارشد مهندسی فناوری اطلاعات  
Email: t.tavasoli67@yahoo.com

انتباه آنها، خطاهای موجود در شبکه را تشخیص داده و سپس بوسیله الگوریتم رقابت استعماری و منطق فازی علت وقوع خطاها را شناسایی می کند [۱ - ۶].

در ادامه مقاله در بخش ۲، مروری بر روش های گذشته ارائه شده خواهیم داشت. سپس در بخش ۳ به تعریف مسئله خواهیم پرداخت. در بخش ۴ یک جدید برای شناسایی علت خطاها در شبکه های حسگر بیسیم معرفی خواهد شد. در بخش ۵ شبیه سازی و نتایج بدست آمده از شبیه سازی ها مورد بررسی قرار خواهد گرفت و در نهایت در بخش ۶ نتیجه گیری و کارهای آتی ارائه خواهد شد.

## ۲. کارهای مرتبط

روش های مختلفی برای خطایابی در شبکه های حسگر بیسیم وجود دارند. هر یک از این روش ها سعی می کنند به نوعی شبکه را مدل کنند و سپس براساس مدلی که از شبکه دارند، مدیریت عیب در شبکه را به صورت کارا انجام دهند. در این بخش به برخی از روش هایی که اخیرا برای شناسایی خطا در شبکه ها ارائه شده اند می پردازیم. گرشاسی و همکاران [۳] یک روش هیوریستیک برای شناسایی عیوب در شبکه های کامپیوتری ارائه کرده اند، این روش براساس استراتژی غیرفعال می باشد. نتایج بدست آمده در این روش نشان می دهد که این روش بدون اینکه تاثیر منفی را بر روی ترافیک شبکه داشته باشد، می تواند مکان دقیق علل وقوع خطا در شبکه را شناسایی کند. در [۱۳] یک روش مبتنی بر الگوریتم ژنتیک برای خطایابی در شبکه های کامپیوتری ارائه شده است. این روش نیز بدون تاثیر منفی بر روی ترافیک شبکه، خطایابی را انجام می دهد، اما در مواردی که شبکه از نظر مقیاس گسترده می شود، زمان شناسایی علل خطاها طولانیتر می شود. بارفورد و همکاران [۱] یک الگوریتمی را ارائه کرده اند که تعدادی بسته را برای تشخیص مکان خطا، بر روی بخشی از شبکه در هر لحظه ارسال می کند که این سربار اضافی ترافیک را در مرحله تشخیص کاهش می دهد. ژائو و همکاران [۷] یک روش فعالی ارائه کرده اند که سنجش هر یک از مسیرهای شبکه را در چند مرحله زمانبندی می کند و براساس این زمانبندی، بخشی از لینک های شبکه را در هر مرحله بررسی و نظارت می کند، این روش باعث می شود ترافیک اضافی کاهش یابد و هزینه تست نیز کاهش پیدا می کند. بینگ وانگ و همکاران [۲] یک الگوریتم هیوریستیک برای خطایابی در شبکه های حسگر بیسیم ارائه کرده اند، این روش اجزائی را برای تست انتخاب می کند که بیشترین استفاده اشتراکی را در مسیرهای مختلف دارند، در این روش علاوه بر اینکه ترافیک اضافی در روی شبکه ارسال نمی شود، همچنین سعی می شود هزینه تست حداقل شود. لولو و همکاران [۸] روشی معرفی کرده اند که فرایند شناسایی مکان خطاها را در مراحل چندگانه تقسیم می کند، در هر مرحله فقط عملیات خطایابی را در بخش کوچکی از شبکه و با استفاده از تعداد اندکی بسته اضافی، انجام می دهد. این روش نیز تلاش می کند تاثیر مخرب کمتری را بر روی ترافیک بگذارد و شناسایی علت خطاها را با حداقل هزینه ممکن انجام دهد. پاتریک و همکاران [۹] چندین الگوریتم هیوریستیک را برای انتخاب اجزای کاندید به منظور تست این اجزا معرفی کرده اند که هزینه تست را حداقل می کند. هونگ نگوین و همکاران [۱۰] روشی ارائه کرده اند که با استفاده از اصول بیزی به شناسایی علت خطاها در شبکه های کامپیوتری می پردازند.

## ۳. تعریف مسئله و فرضیات تحقیق

شبکه های حسگر بیسیم از اجزا و پیوندهای ارتباطی که در بین اجزا وجود دارد، تشکیل شده اند. در واقع منظورمان از اجزای موجود در شبکه های حسگر بیسیم گره ها، لینک های ارتباطی بین آن ها، ایستگاه اصلی و منابع است. در واقع

هر یک از این اجزا را به صورت یک گره فرض می کنیم، لذا می توان اجزای شبکه را به صورت یک درخت یا یک گراف نمایش داد [۳، ۱۱]. برای خطایابی در شبکه های حسگر بیسیم در این مقاله، نیاز داریم که شکل فیزیکی شبکه (توپولوژی فیزیکی) را به یک شکل منطقی تبدیل کنیم. به عبارتی دیگر باید توپولوژی فیزیکی شبکه را به یک گراف تبدیل کرد. لذا شبکه ای در نظر می گیریم که داده های حس شده از محیط از منابع به سمت ایستگاه اصلی ارسال می شوند.

فرض ما بر این است که براساس داده های انتها به انتهای موجود در شبکه می توانیم وجود خطا در شبکه را تشخیص دهیم، این فرض براساس مرجع [۲، ۳] انتخاب شده است و به اینصورت در نظر گرفته می شود که مقدار ناکافی و غیرمتناسب از داده ها در ایستگاه اصلی نشان دهنده وجود خطا در شبکه است و مقدار داده های کافی و متناسب از داده ها در ایستگاه اصلی نشانگر این است که شبکه عملکرد نرمال و عادی را دارد. وضعیت هر یک از اجزا بوسیله تست آن ها مشخص می شود. لذا تست هر جز از شبکه دارای هزینه های زیادی است. بنابراین انتخاب جز مناسب برای تست بسیار حائز اهمیت می باشد. هر خطا در شبکه های حسگر بیسیم می تواند در شکل های مختلفی باشد، برای مثال ممکن است پیوندهای ارتباطی (Links) گره ها دچار خرابی (خطا) شوند [۳، ۱۲، ۱۳] یا خود گره های حسگر خراب شوند، لذا خطا در شبکه می تواند در اجزای مختلفی در شبکه اتفاق بیافتد. در این مقاله فرض می کنیم که در حالت کلی با یک نوع خطا روبه رو هستیم و فرض ما بر این است که گره ها در گراف دچار خطا می شوند، که این گره ها در واقع متناظر با اجزایی هست که در شکل فیزیکی شبکه قرار دارند.

مسئله اصلی این است که دقیقا مشخص نیست کدام جز یا اجزا دچار خرابی شده اند. برای شناسایی مکان دقیق اجزای خراب در شبکه حسگر بیسیم باید اجزایی از شبکه تست شوند. اما تست اجزا دارای هزینه های زیادی است. لذا مسئله اصلی حداقل سازی هزینه تست است. بنابراین باید گرهی را برای تست انتخاب کنیم که بیشترین احتمال خرابی را دارد. حال چگونگی تعیین این احتمال ها و اینکه کدام گره را برای تست انتخاب کنیم به طوری که هزینه خطایابی حداقل شود، یک مسئله NP-Complete است. در ادامه یک روش مبتنی بر الگوریتم رقابت استعماری و منطق فازی برای خطایابی در شبکه های حسگر بیسیم ارائه می نمائیم و جزئیات آن را تشریح می کنیم.

#### ۴. الگوریتم پیشنهادی

در این بخش به توضیح روش پیشنهادی می پردازیم که از الگوریتم رقابت استعماری و منطق فازی برای خطایابی در شبکه های حسگر بیسیم استفاده می کند.

برای بازنمایی کشورها در الگوریتم رقابت استعماری، به صورت گسسته و از روش کدینگ باینری استفاده می کنیم. در واقع هر کشور یک مسیر در شبکه حسگر بیسیم را نشان می دهد و هر عنصر از کشور به یک گره اشاره می کند. هر عنصر می تواند دو مقدار ۱ یا ۰ را داشته باشد. اگر مقدار عنصر ۱ باشد، یعنی گره متناظر با این عنصر در این مسیر برای انتقال داده ها مورد استفاده قرار نمی گیرد (در مسیریابی ها). اگر مقدار عنصر ۰ باشد، یعنی گره متناظر با این عنصر در مسیر مورد استفاده قرار می گیرد. شکل (۱) نحوه کدینگ کشورها را نشان می دهد.

	Node 1	Node 2	Node 3
Path 1	0	0	0
Path 2	1	0	1
Path 3	0	0	1

شکل (۱): مثالی از نحوه بازنمایی کشورها

برای شروع الگوریتم رقابت استعماری باید تعدادی کشور اولیه وجود داشته باشد. لذا در اینجا، مسیرهای خراب را به عنوان کشورهای اولیه الگوریتم رقابت استعماری در نظر می گیریم. با توجه به اینکه می توان با استفاده از داده های انتهایی مسیرهای خراب (مسیرهای دارای خطا) را شناسایی کرد، لذا می توان آن ها را به عنوان کشورهای اولیه در نظر گرفت. مهمترین بخش از الگوریتم های بهینه سازی، نحوه تعریف تابع معیار می باشد، که باید متناسب با مسئله تعریف گردد. در این بخش برای ارزیابی هر کشور از دو پارامتر استفاده می کنیم:

(۱) تعداد استفاده اشتراکی گره ها در جفت های مختلف (درجه هر گره)

(۲) تعداد گره هایی که در یک جفت مورد استفاده قرار می گیرند.

با استفاده از نحوه کدینگ کشورها و داده های موجود در آن ها (آرایه ها) می توان برای هر یک از گره ها یک وزن ( $w_i$ ) در نظر گرفته می شود. وزن هر گره از طریق رابطه (۱) بدست می آید. که  $N$  تعداد کل کشورها را نشان می دهد.  $w_i$  وزن گره  $i$ ام را محاسبه می کند.  $a$  تعداد کل کشورها است.  $n_{ik}$  مقدار عنصر  $i$ ام در کشور  $k$ ام از هر کشور را نشان می دهد. برای محاسبه این پارامتر برای هر کشور از رابطه (۲) استفاده می کنیم.

$$w_i = N - \sum_{k=1}^a n_{ik} \quad (1)$$

$$A = \sum_{i=1}^N n_i \quad (2)$$

در رابطه (۲) مقدار عنصر  $i$ ام را در کشور نشان می دهد.  $N$  تعداد کل گره ها را نشان می دهد. در واقع برای هر کشور یک مقدار  $A$  که نشان تعداد گره هایی است که مورد استفاده قرار گرفته است محاسبه می گردد. با توجه به توضیحات بالا، برای ارزیابی و محاسبه برازندگی هر کشور از رابطه (۳) استفاده می کنیم. که  $\alpha$  مقداری بین صفر و یک است. هر چقدر مقدار به دست آمده از رابطه فوق بیشتر باشد در اینصورت آن کشور مقدار برازندگی بیشتری دارد و شانس بیشتری برای انتخاب شدن به عنوان استعمارگر را دارد.

$$f_i = (1 - \alpha)A + \alpha w_i \quad (3)$$

همانطور که گفته شد، برای شروع الگوریتم رقابت استعماری باید تعدادی امپراتوری اولیه وجود داشته باشد. تعداد  $N_{country}$  کشور اولیه را ایجاد می کنیم، سپس  $N_{imp}$  تا از بهترین اعضای این جمعیت (کشورهای دارای کمترین مقدار برازندگی بهتر) را به عنوان امپریالیست انتخاب می کنیم. باقیمانده  $N_{col}$  تا از کشورها، مستعمراتی را تشکیل می دهند که هر کدام به یک امپراطوری تعلق دارند. برای تقسیم مستعمرات اولیه بین امپریالیست ها، به هر امپریالیست، تعدادی از مستعمرات را که این تعداد، متناسب با قدرت آن است، می دهیم.

به منظور تعریف سیاست جذب در الگوریتم رقابت استعماری، از عملگر مشابه با عملگر ترکیب در الگوریتم ژنتیک استفاده می کنیم. به منظور اعمال سیاست جذب، هر کشور با استعمارگر خود ترکیب می شود و کشور جدیدی بوجود می آید. کشور جدید بوجود آمده جایگزین کشور قبلی می گردد. کشورها براساس احتمال ترکیب (احتمال جذب)، برای اعمال عملگر ترکیب بر روی آن ها (ترکیب با استعمارگر)، انتخاب می شوند. در این مقاله از روش ترکیب تک نقطه ای برای عملگر ترکیب استفاده می کنیم. بر اساس روش ترکیب تک نقطه ای، تعدادی مستعمره براساس احتمال ترکیب برای انجام عمل

ترکیب با استعمارگر انتخاب می شوند شکل (۲) مثالی از نحوه انجام عمل ترکیب به منظور اعمال سیاست جذب را نشان می دهد.

نقطه برش

استعمارگر	0	1	1	0	1	0	0	1
مستعمره	0	1	0	0	1	0	1	1
مستعمره جدید	0	1	0	0	1	0	0	1

شکل (۲): مثالی از نحوه انجام عمل ترکیب به منظور اعمال سیاست جذب

پس از اینکه سیاست جذب انجام شد، قدرت امپراتوری ها محاسبه می گردد. قدرت یک امپراطوری برابر است با قدرت کشور استعمارگر، به اضافه درصدی از قدرت کل مستعمرات آن. منظور از قدرت یک کشور مقدار برازندگی آن است که نحوه محاسبه آن مورد بررسی قرار گرفت. لذا براساس مقدار برازندگی هر کشور قدرت هر امپراتوری براساس رابطه (۴) محاسبه می گردد. که در آن  $T.C_n$  هزینه کل امپراطوری  $n$ ام و  $\alpha$  عددی مثبت است که معمولاً بین صفر و یک و نزدیک به صفر در نظر گرفته می شود. کوچک در نظر گرفتن  $\alpha$  باعث می شود که هزینه کل یک امپراطوری، تقریباً برابر با هزینه حکومت مرکزی آن (کشور امپریالیست)، شود و افزایش  $\alpha$  نیز باعث افزایش تاثیر میزان هزینه مستعمرات یک امپراطوری در تعیین هزینه کل آن می شود.

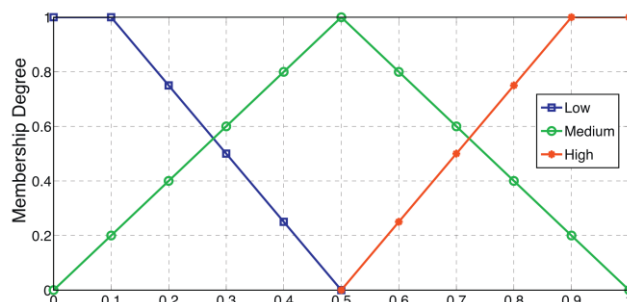
$$T.C_n = f(imperialist_n) + \alpha(f(colonies of empire_n)) \quad (4)$$

بعد از هر بار عمل جذب، مقدار برازندگی کشورهای هر امپراتوری محاسبه می گردد. اگر برازندگی کشور مستعمره از برازندگی کشور استعمارگر در همان امپراتوری بیشتر باشد در این صورت جای مستعمره و استعمارگر در آن امپراطوری جابه جا خواهد شد. برای مدل کردن رقابت استعماری، فرض می کنیم که امپراطوری در حال حذف، ضعیف ترین امپراطوری موجود است (امپراتوری که دارای برازندگی کمتری نسبت به امپراتوری های دیگر است). بدین ترتیب، در تکرار الگوریتم، یکی یا چند تا از ضعیف ترین مستعمرات ضعیف ترین امپراطوری را برداشته و برای تصاحب این مستعمرات، رقابتی را میان کلیه امپراطوری ها ایجاد می کنیم. مستعمرات مذکور، لزوماً توسط قویترین امپراطوری، تصاحب نخواهند شد، بلکه امپراطوری های قویتر، احتمال تصاحب بیشتری دارند.

در اثر رقابت های امپریالیستی و سیاست جذب، عناصر کشورها تغییر پیدا می کنند. سپس هر کشور مورد بررسی قرار می گیرد. بررسی به این منظور صورت می گیرد احتمال خراب بودن گره ها تعیین شود. شرط لازم برای خاتمه الگوریتم این است که در  $n$  دفعات متوالی هیچ بهبودی در قدرت امپراتوری ها وجود نداشته باشد و یا به یک امپراتوری واحد برسیم. پس از اینکه الگوریتم رقابت استعماری خاتمه یافت، بهترین امپراتوری انتخاب می شود و همه کشورهای آن امپراتوری براساس رابطه (۵) بررسی شده و احتمال خراب بودن گره ها تعیین می گردد. در رابطه (۵) که  $P_i$  احتمال خراب بودن گره  $n$ ام را نشان می دهد.  $C$  تعداد کل کشورهای بهترین امپراتوری (یا تنها امپراتوری) است که شامل کشور استعمارگر و کشورهای مستعمره است.  $N$  تعداد کل گره های حسگر است، درواقع طول هر کشور است.  $n_j$  نیز نشان دهنده مقدار عنصر  $n$ ام در کشور  $n$ ام است.

$$P_i = \frac{\sum_{i=1}^C \sum_{j=1}^N n_j / N}{C} \quad (5)$$

پس از تعیین احتمال خرابی هر یک از گره ها، از منطق فازی برای تعیین احتمال تست گره ها استفاده می کنیم و براساس احتمال تست، هر یک از گره ها را برای پیدا کردن گره خراب تست می کنیم. لذا می توان مقادیر دو پارامتر احتمال خرابی گره ها و هزینه تست گره ها را به صورت فازی تعریف کرد. یعنی می توان از سه متغیر فازی برای فازی سازی این پارامترها استفاده کرد. در این مقاله ما برای هر یک از پارامترها (متغیرها) سه حالت فازی در نظر گرفته ایم. طبق قوانین فازی سه وضعیت (بالا، متوسط و پایین) برای احتمال خرابی گره ها در نظر گرفته می شود، سه وضعیت (کم، متوسط و زیاد) برای هزینه تست گره ها در نظر گرفته می شود. براساس متغیرهای ورودی فازی که در شکل (۳) نشان داده شده اند. می توان قوانین اگر-آنگاه (*if-then*) را به صورت جدول (۱) ایجاد کرد. گرهی که دارای مقدار احتمال بیشتری است، مسلماً احتمال خرابی آن بیشتر است. گرهی که دارای مقدار احتمال بیشتری است، مسلماً شانس بیشتری برای تست شدن خواهد داشت. در اینجا برای تعیین اینکه کدام گره تست شود از روش انتخاب چرخ رولت استفاده شده است. اگر گره انتخاب شده خراب باشد در اینصورت تعمیر می شود. اگر گره خراب نباشد (سالم باشد)، الگوریتم پیشنهادی دوباره تکرار خواهد شد تا گره دیگری برای تست انتخاب شود. الگوریتم تا زمانی همه گره های خراب شناسایی شوند و شبکه عمکرد عادی داشته باشد، تکرار خواهد شد.



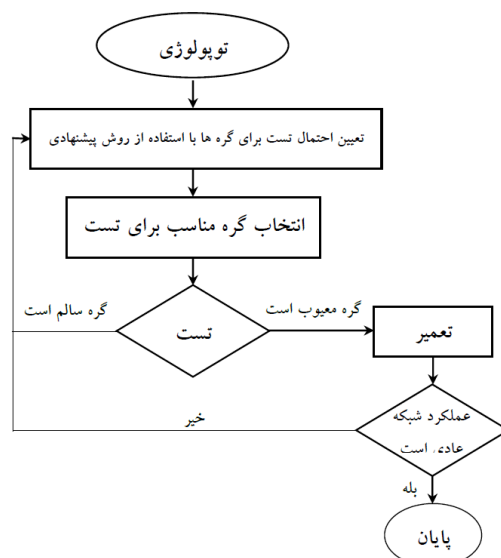
شکل (۳): متغیرهای ورودی فازی در نظر گرفته شده

جدول (۱): قوانین اگر-آنگاه برای محاسبه احتمال تست گره های شبکه

شانس انتخاب (مقدار احتمال)	احتمال خرابی گره ها	هزینه تست هر گره
۰.۹	بالا	کم
۰.۸	بالا	متوسط
۰.۷	بالا	زیاد
۰.۶	متوسط	کم
۰.۵	متوسط	متوسط
۰.۴	متوسط	زیاد
۰.۳	بالا	کم
۰.۲	بالا	متوسط
۰.۱	بالا	زیاد

شکل (۴) فلوچارت روش پیشنهادی را نشان می دهد.





شکل (۴): فلوچارت روش پیشنهادی

## ۵. شبیه سازی و ارزیابی نتایج

در این بخش به پیاده سازی روش ارائه شده پرداخته و نتایج بدست آمده از شبیه سازی ها مورد تجزیه و تحلیل قرار می گیرد. برای ارزیابی عملکرد الگوریتم ها آن ها را در نرم افزار متلب پیاده سازی کرده و در سناریوهای مختلفی از شبکه اجرا می کنیم. الگوریتم ها را بر اساس هزینه تست مورد بررسی و ارزیابی قرار می دهیم. عملکرد الگوریتم پیشنهادی را با چند روش مشابه دیگر که اخیرا ارائه شده اند [۲، ۳، ۷]، مورد مقایسه قرار می دهیم و نتایج را مورد تجزیه و تحلیل قرار می دهیم.

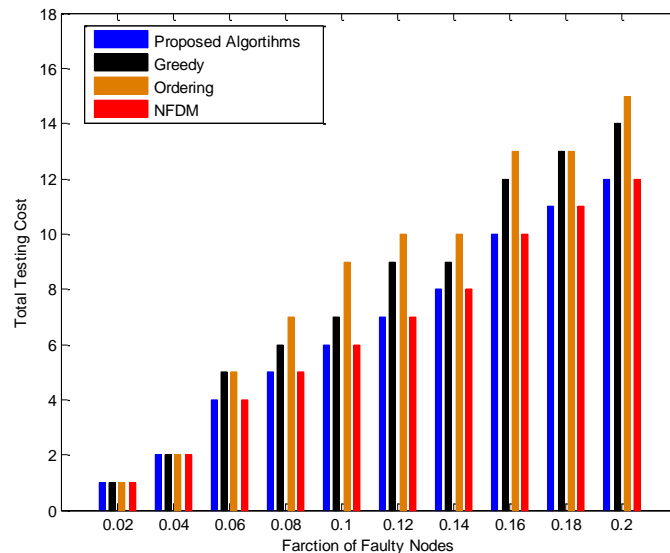
مدل شبکه به صورت معماری مبتنی بر کلاینت-سرور (در شبکه های کامپیوتری) در نظر گرفته شده است. این مدل که شامل سه بخش منابع، ایستگاه اصلی و گره های شبکه است. در شبیه سازی ها ما مسیریابی پویا در نظر می گیریم و فرض می کنیم که از هر کلاینت به سرور ممکن است چندین مسیر وجود داشته باشد. همچنین در شبیه سازی ها هزینه تست هر جزء را متفاوت در نظر می گیریم. علاوه بر هزینه تست که برای هر گره در نظر گرفته می شود، هر گره دارای احتمالی است که میزان خرابی آن را نشان می دهد. در ابتدای شروع الگوریتم فرض می کنیم که احتمال خرابی گره های شبکه را نمی دانیم و لذا احتمال همه گره ها را قبل از شروع الگوریتم برابر در نظر می گیریم.

برای بررسی و ارزیابی الگوریتم پیشنهادی در شرایط مختلف و بررسی عملکرد آن در مقیاس ها و شرایط مختلف شبکه، سه سناریو مختلف برای انجام آزمایشات در نظر گرفته شده است. هدف از در نظر گرفتن سناریوهای مختلف این است که عملکرد الگوریتم ها را در شرایط مختلف ارزیابی کنیم و مقیاس پذیر بودن الگوریتم پیشنهادی را ارزیابی کنیم. جدول (۲) سه سناریوی مختلف (برای انجام آزمایشات) را نشان می دهد.

جدول (۲): سناریوی مختلف در نظر گرفته شده برای انجام آزمایشات

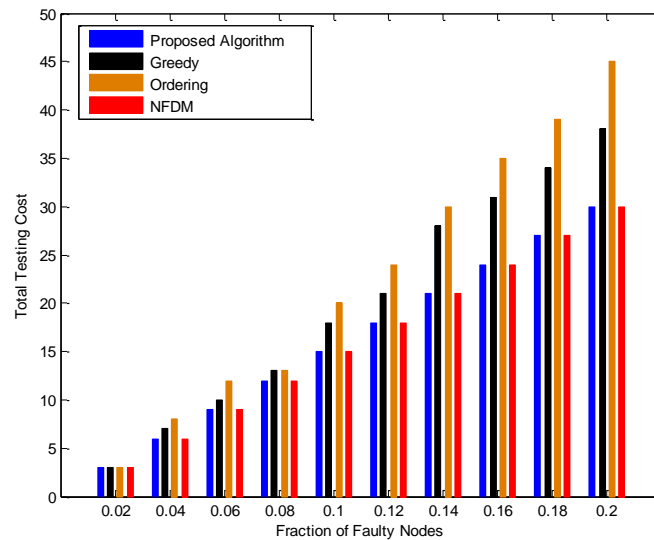
سناریوها	تعداد گره ها	تعداد منابع
سناریو اول	۶۰	۸۰
سناریو دوم	۱۵۰	۸۰
سناریو سوم	۱۵۰	۱۵۰

شکل (۵) نتایج بدست آمده برای هزینه تست مجموع در سناریوی اول را نشان می دهند. در نمودارها، محور عمودی (محور Y) هزینه تست مجموع را نشان می دهد و محور افقی درصد گره های خراب را نشان می دهد. با توجه به نتایج نشان داده شده در شکل (۵) هزینه تست مجموع الگوریتم پیشنهادی نسبت به الگوریتم Greedy و Ordering در سناریو اول بهتر است. همچنین نتایج نشان می دهد که هزینه تست مجموع در سناریوی اول برای الگوریتم پیشنهادی و NFDM برابر است. لذا این نتایج نشانگر این است که الگوریتم پیشنهادی می تواند هزینه تست مجموع را حداقل کند. در نتایج نشان داده در نمودارها، هرچقدر تعداد و درصد گره های خراب افزایش پیدا می کند، هزینه تست مجموع نیز افزایش پیدا می کند. دلیل این امر این است که با افزایش تعداد گره های خراب در شبکه نیاز به تعداد تست های بیشتری داریم تا گره های خراب شناسایی گردند. همچنین با افزایش تعداد منابع هزینه تست کاهش پیدا می کند. دلیل این امر این است که با افزایش تعداد منابع، اطلاعات بیشتری از شبکه را در اختیار داریم و می توانیم انتخاب و تست گره های شبکه را دقیقتر و با هزینه کمتری انجام دهیم، اما زمانی که تعداد منابع کمتر است و چون اطلاعات کمتری از گره های شبکه در اختیار داریم، لذا انتخاب و تست گره های شبکه با دقت کمتری صورت می گیرد.



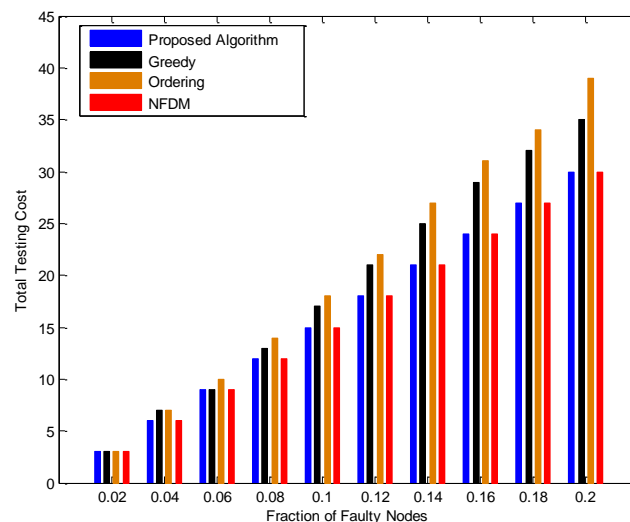
شکل (۵): هزینه تست مجموع در سناریوی اول





شکل (۶): هزینه تست مجموع در سناریوی دوم

شکل (۶) نتایج بدست آمده از هزینه تست مجموع را در سناریوی دوم نشان می دهد. در سناریوی دوم تعداد گره ها را افزایش داده ایم، هدف از در نظر گرفتن این سناریو این است که عملکرد الگوریتم را زمانی که تعداد منابع کمتری در شبکه وجود دارد و اطلاعات کمتری می توان از گره های شبکه داشت، مورد بررسی قرار داد. با توجه به نتایج بدست آمده، مشاهده می شود که الگوریتم پیشنهادی می تواند هزینه تست مجموع را نسبت به الگوریتم های دیگر کاهش دهد. با توجه به نتایج، هزینه تست الگوریتم پیشنهادی و الگوریتم NFDM تقریباً برابر است. دلیل این امر این است که این دو الگوریتم می توانند خطایابی را با حداقل هزینه ممکن انجام دهند.



شکل (۷): هزینه تست مجموع در سناریوی سوم

شکل (۷) نتایج بدست آمده از هزینه تست مجموع را در سناریوی سوم نشان می دهد. در سناریوی سوم تعداد گره ها و تعداد منابع را افزایش داده ایم و آن ها را برابر با ۱۵۰ در نظر گرفته ایم، همچنین تعداد منابع و تعداد گره ها برابر در نظر گرفته شده است. با توجه به نتایج بدست آمده در سناریوی سوم، مشاهده می شود که هزینه تست مجموع بدست آمده از الگوریتم پیشنهادی کمترین مقدار ممکن است. براساس نتایج بدست آمده در سه سناریوی مختلف، هزینه تست مجموع الگوریتم پیشنهادی نسبت به الگوریتم های دیگر به مراتب بهتر است و الگوریتم پیشنهادی توانسته است هزینه تست مجموع را حداقل مقدار ممکن برساند.

جدول (۳) میانگین هزینه تست به ازای هر گره را در الگوریتم های مختلف را نشان می دهد. با توجه به نتایج نشان داده در جدول (۳) مشاهده می شود که هزینه تست به ازای هر گره در الگوریتم پیشنهادی کمتر از سه الگوریتم دیگر است. لذا با توجه به نتایج بدست آمده از شبیه سازی ها، مشاهده می شود که الگوریتم پیشنهادی بهتر از سه الگوریتم دیگر و با هزینه کمتر می تواند علت خطاها (گره های خراب) را در شبکه شناسایی کند.

جدول (۳): میانگین هزینه تست به ازای هر گره

NFDM	Ordering	Greedy	الگوریتم پیشنهادی
1.10	1.26	1.15	1.02

#### ۶. نتیجه گیری

در این مقاله یک روش جدید مبتنی بر الگوریتم رقابت استعماری و منطق فازی برای خطایابی در شبکه های حسگر بیسیم ارائه شد. روش ارائه شده بدون اینکه تاثیر منفی را بر روی ترافیک شبکه بگذارد، براساس داده های انتهایی شبکه خطاها را تشخیص داده و از طریق الگوریتم رقابت استعماری و منطق فازی علت خطاها را شناسایی می کند. براساس نتایج بدست آمده، روش ارائه شده در این مقاله عملکرد بهتری نسبت به روش های موجود دیگر دارد. کارهای عمده پیش رو برای ادامه کار می توانند به صورت زیر باشند:

۱. بررسی الگوریتم پیشنهادی براساس معیار زمان
۲. ارائه نسخه ای از الگوریتم پیشنهادی متناسب با محیط های ناهمگن و بلادرنگ
۳. ارائه نسخه توزیع شده از الگوریتم پیشنهادی
۴. بررسی عملکرد الگوریتم پیشنهادی تحت سناریوها و پارامترهای دیگر

1. Barford P, Duffield N, Amos R and Joel S. 2009. Network performance anomaly detection and localization. 28th IEEE International Conference on Computer Communications. Rio de Janeiro, April 19-25, pp:1377-1385.
2. Bing W, Wei W, Hieu D, Wei Z, Krishna R. 2012. Fault localization using passive end-to-end measurements and sequential testing for wireless sensor networks. IEEE TRANSACTIONS ON MOBILE COMPUTING, 11:439-452.
3. Garshasbi M.S and Jamali Sh. 2014. A New Fault Detection Method Using End-to-End Data and Sequential Testing for Computer Networks. International Journal Information Technology and Computer Science. 01, 93-100.
4. Coates M, Nowak R. 2000. Network loss inference using unicast end-to-end measurement. Modeling and Management, 23:19-28.
5. Maitreya N, Adarshpal SS, Errol L. 2008. Efficient probe selection algorithms for fault diagnosis. Telecommun Syst, 37: 109-125.
6. Małgorzata S, Adarshpal S. 2004. A survey of fault localization techniques in computer networks. Science of Computer Programming 53:165-194.
7. Zhao S, Daachi B, Djouani K. 2012. Application of fuzzy inference systems to detection of faults in wireless sensor networks. Neurocomputing, 94:111 -120.
8. LOLO M and Otman B. 2009. A new probing scheme for fault detection and Identification. IEEE International Conference on Electro/Information Technology. Windsor, USA, June 7-9, pp:90-95.
9. Patrick P, Amitava M, Mrinal K, Mita N. 2011. Minimal monitor activation and fault localization in optical networks. Optical Switching and Networking, 8:46-55.
10. Hong S, Adarshpal S. 2004. Non-deterministic fault localization in communication systems using belief networks. IEEE/ACM Transactions on Networking, 12:809-822.
11. Steinder M, Sethi A. 2002. Distributed fault localization in hierarchically routed networks. Management Technologies for E-Commerce and E-Business Applications, 125:195-207.

۱۲. گرشاسبی م.ص، جمالی ش. ۱۳۹۳. ارائه یک روش مبتنی بر الگوریتم ژنتیک برای مکانیابی اجزای معیوب در شبکه های کامپیوتری. اولین کنفرانس ملی ریاضیات صنعتی.
۱۳. گرشاسبی م.ص، جمالی ش. ۱۳۹۳. ارائه یک روش هیوریستیک برای مکانیابی خطا در شبکه های کامپیوتری. اولین کنفرانس ملی ریاضیات صنعتی.