

یک الگوریتم مدیریت خرابی سلسله مراتبی مبتنی بر خوشه‌بندی برای شبکه‌های حسگر بی‌سیم

شهرام بابائی^۱، احمد خادم‌زاده^۲، کامبیز بدیع^۳

^۱ دانشگاه آزاد اسلامی واحد علوم و تحقیقات، گروه کامپیوتر، تهران، ایران Sh.babaie@iaut.ac.ir

^۲ مرکز تحقیقات مخابرات ایران، تهران، ایران. Zadeh@itrc.ac.ir

^۳ مرکز تحقیقات مخابرات ایران، تهران، ایران. K_badie@itrc.ac.ir

چکیده:

به دلیل به کارگیری حسگرها در محیط‌های دور از دسترس، عوامل مخرب محیطی و عملیات خرابکارانه دشمن، ایجاد هرگونه خرابی در شبکه‌های حسگر بی‌سیم امری اجتناب ناپذیر است. اغلب روش‌های کشف خرابی مبتنی بر مقایسه محلی، در مواقعی که بیش از نصف گره‌های همسایه معیوب باشند و علت خرابی گره‌های شبکه مشترک باشد، قادر به شناسایی صحیح حسگرهای معیوب نخواهند بود. لذا در این مقاله یک رویکرد کشف خرابی سلسله‌مراتبی مبتنی بر خوشه‌بندی ارائه می‌شود که شرایطی فراهم می‌کند تا برای تعیین وضعیت حسگرهای شبکه به مقایسه داده هر حسگر با داده حسگرهای همسایه اکتفا نکرده و با بررسی داده حسگرهای غیر همسایه در لایه بالاتر، تصمیم درستی در مورد وضعیت حسگرها اتخاذ شود. همچنین به دلیل ناکارآمدی رویکردهای کشف خرابی ایستا، یک رویکرد هوشمند به منظور تعیین زمان مناسب برای اجرای الگوریتم پیشنهادی ارائه می‌شود که به صورت پویا تعداد دفعات اجرای الگوریتم را کاهش و موجب افزایش طول عمر شبکه می‌شود. نتایج شبیه‌سازی‌های انجام شده در نرم‌افزار متلب حاکی از دقت کشف خرابی بالا و نرخ اختلال نادرست پایین رویکرد پیشنهادی دارد. شبیه‌سازی‌ها در چگالی‌های مختلف و با احتمال‌های مختلف خرابی و تعداد همسایه‌های مختلف مورد ارزیابی قرار گرفته و مقیاس‌پذیری بودن آن و توانایی آن در کشف خرابی اثبات می‌شود.

کلید واژه‌ها: شبکه‌های حسگر بی‌سیم، تحمل‌پذیری خرابی، کشف خرابی سلسله‌مراتبی، دقت کشف خرابی، نرخ اختلال نادرست.

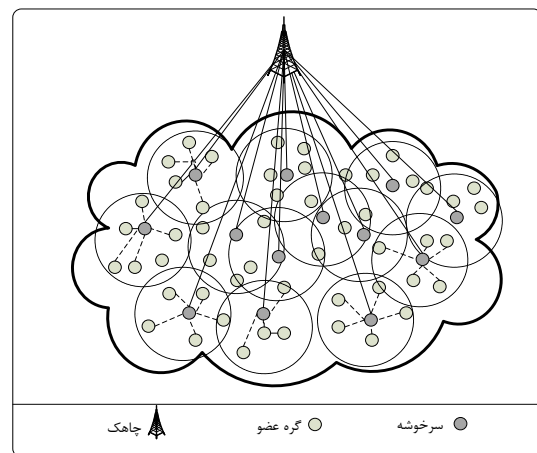
قابلیت‌های حسگری متفاوت مانند تشخیص لرزش، صوت و رطوبت می‌توانند رخداد‌های مختلف محیط را شناسایی و گزارش حاصل را به چاهک ارسال کنند [۲ و ۳]. با توجه به توزیع آسان، هزینه پایین نصب و بی‌سیم بودن ارتباطات باعث به کارگیری این شبکه‌ها از مصارف تحقیقاتی، نظارت بر محیط زیست گرفته تا کاربردهای نظامی و مراقبت‌های پزشکی شده است [۴].

۱. مقدمه

شبکه حسگر بی‌سیم شامل تعداد زیادی گره حسگر بی‌سیم بوده که در محیط‌هایی که انسان نمی‌تواند حضور داشته باشد یا برای انسان خطرناک هستند پخش شده و به نظارت محیط و مخابره رخداد‌های تشخیص داده شده می‌پردازند [۱]. حسگرها با

چاهک از طریق تعدادی گام کوتاه و بواسطه سرخوشه‌های دیگر برقرار خواهد شد. از آنجاییکه انرژی مصرفی با فاصله بین فرستنده و گیرنده رابطه مستقیم دارد؛ رویکردهای چندگامه مصرف انرژی پایین تر و رویکردهای تک‌گامه دارای تأخیر انتها به انتهای کمتر و نرخ گم شدن بسته کمتری دارند [۷]. الگوریتم‌های متعددی برای خوشه‌بندی شبکه‌های حسگر بی‌سیم ارائه شده است که از آن جمله می‌توان به EEHC، HEED و DWEHC اشاره کرد که نحوه خوشه‌بندی و چگونگی تعیین سرخوشه‌های این الگوریتم‌ها با هم متفاوت هستند. یکی از معروف‌ترین پروتکل خوشه‌بندی در شبکه‌های حسگر بی‌سیم، الگوریتم Leach می‌باشد [۸]. در این الگوریتم، خوشه‌های شبکه به صورت تصادفی، وفقی و خودسازمانده تشکیل می‌شوند. انتخاب سرخوشه‌ها در هر دوره به صورت تصادفی صورت می‌گیرد و هیچ‌گونه رویکرد از قبل مشخص شده‌ای در تعیین سرخوشه‌ها نقشی ندارد. رویکرد وفقی به این صورت که گره‌هایی که در دوره‌های قبلی سرخوشه بوده‌اند، در دور بعدی نمی‌توانند برای سرخوشه بودن کاندیدا شوند. به این ترتیب انتظار می‌رود که در پایان تعداد مشخصی از دوره‌ها، تمامی گره‌ها سرخوشه شده باشند. در ضمن گره‌های شبکه در این پروتکل بدون هیچ کمکی از عامل خارجی یا گره خاصی از شبکه، عمل خوشه‌بندی را انجام می‌دهند که مشخص کننده خودسازمانده بودن این الگوریتم می‌باشد. وجود این ویژگی متضمن مقیاس‌پذیر بودن این الگوریتم می‌باشد. محققان تحقیقات گسترده‌ای را در جهت بهبود عملکرد این الگوریتم ارائه داده‌اند که متأسفانه در اغلب پروتکل‌های ارائه شده ویژگی خود-سازمانده بودن این الگوریتم از بین رفته است. در الگوریتم پیشنهادی بعلاوه پیچیدگی پایین، از این الگوریتم به‌مراه یک پارامتر شایستگی که در ادامه معرفی می‌شود؛ برای خوشه‌بندی استفاده شده است. علاوه بر محدودیت انرژی، ایجاد خرابی در گره‌های

محدودیت انرژی گره‌های حسگر از مشکلات عمده این شبکه‌ها محسوب می‌شود. یکی از موثرترین روش‌ها برای کاهش انرژی مصرفی گره‌ها، مدیریت هدایت مناسب بسته‌های شبکه می‌باشد. از مفیدترین رویکردهای پیشنهادی می‌توان به تکنیک خوشه‌بندی اشاره کرد. برای این منظور کل گره‌های شبکه به خوشه‌های مجزا از هم با کمترین همپوشانی ممکن، تقسیم می‌شوند. در هر خوشه یک گره سرخوشه به عنوان مسئول ارتباط با چاهک انتخاب می‌شود. به این ترتیب هر گره داده حس شده خود را به سرخوشه فرستاده و سرخوشه بعد از تجمیع و حذف داده‌های زاید و تکراری، آنها را به چاهک می‌فرستد [۵ و ۶]. بنابراین با کاهش تعداد دفعات ارسال و دریافت، طول عمر شبکه افزایش می‌یابد. شکل ۱ خوشه‌بندی فرضی برای یک شبکه حسگر بی‌سیم را نشان می‌دهد.



شکل ۱: خوشه‌بندی شبکه‌های حسگر بی‌سیم

از مزایای خوشه‌بندی می‌توان به سهولت مدیریت شبکه، سهولت مدیریت خرابی، افزایش امنیت، کاهش ترافیک و غیره اشاره کرد. برای ارتباط سرخوشه با چاهک می‌توان از رویکردهای تک‌گامه یا چندگامه استفاده کرد. در رویکردهای تک‌گامه، هر سرخوشه با یک ارتباط مستقیم با چاهک ارتباط برقرار می‌کند. در حالی که در رویکردهای چندگامه، ارتباط سرخوشه با

حسگر و شکست در ارتباطات شبکه از مهم‌ترین چالش‌های شبکه‌های حسگر بی‌سیم محسوب می‌شود. با توجه به دلایل ذیل مدیریت خرابی گره‌های شبکه حسگر بی‌سیم ضروری بنظر می‌رسد. گره‌های حسگر از لحاظ فیزیکی کاملاً شکننده بوده و نیز در محیط‌های خشن و دور از دسترس به کار گرفته می‌شوند، بنابراین ایجاد خرابی چه به صورت عمدی توسط دشمن و چه به صورت غیر عمدی توسط عوامل مخرب محیطی امری رایج می‌باشد. تعداد زیاد حسگرها و توزیع آنها در محیط‌های دور از دسترس، باعث شده که نظارت دستی و فیزیکی به حسگرها عملاً غیرممکن باشد. لذا مدیریت خرابی به صورت خود-سازمانده از ملزومات این شبکه‌ها محسوب می‌شود. از آنجاییکه گره‌های معیوب شبکه، داده نادرست تولید می‌نمایند؛ در این صورت اطلاعات به دست آمده از شبکه مطابق با واقعیت نبوده و واکنش‌های نامناسب را در پی خواهد داشت. همچنین گره‌های معیوب، موجب اختلال در مسیریابی بسته‌های شبکه خواهد شد [۹]. لذا با شناسایی دقیق حسگرهای معیوب و با کنار گذاشتن سریع آنها می‌توان از عملکرد مطلوب شبکه اطمینان حاصل کرد. حسگرها از باتری برای تأمین انرژی لازم برای فعال ماندن استفاده می‌کنند. معمولاً باتری حسگرها غیر قابل شارژ و غیر قابل تعویض می‌باشند. بنابراین تخلیه باتری حسگرها موجب اختلال در عملکرد صحیح آن و شبکه خواهد شد. گره‌های حسگر حساس به قیمت بوده و نمی‌توان در ساخت آنها از قطعات الکترونیکی با کیفیت بالا استفاده کرد؛ لذا با به کارگیری قطعات ارزان، مشکل از کارافتادگی این حسگرها بیش از پیش بدیهی به نظر می‌رسد.

اثر عوامل مخرب مذکور می‌تواند موجب ایجاد خطا در انتقال اطلاعات شود که از آن به عنوان خطای داده^۱ نامبرده می‌شود. برای مقابله با خطاهای داده از رویکردهای مختلف کنترل خطا مانند CRC^۲، LDPC^۳ و ... استفاده می‌شود [۱۰]. علاوه بر خطای

داده، اثر عوامل مخرب ممکن است موجب تغییر رفتار عادی گره‌های حسگر شده که به عنوان خطای وظیفه‌مندی^۴ و تابعی گره حسگر شناخته می‌شود و گره مورد نظر را معیوب می‌نامیم. وجود هر گونه اشکال در یک گره حسگر که موجب غیرفعال شدن کلی آن شود به عنوان خرابی سخت شناخته می‌شود و معمولاً به صورت دائم در گره باقی می‌ماند. در بعضی موارد اشکال ایجاد شده در مؤلفه‌های داخلی گره حسگر، ارتباط آنرا با بقیه گره‌های شبکه قطع نکرده و موجب نادرست بودن داده‌های تولیدی آن خواهد شد و به آن خرابی نرم گفته می‌شود [۱۱]. در این گونه موارد اثر خرابی ممکن است به صورت گذرا یا متناوب در گره نمایان شود. لذا باید تمهیداتی اندیشیده شود که شبکه حسگر بی‌سیم به صورت خودسازنده بروز هرگونه خرابی در گره‌های حسگر را پوشانده و فعالیت خود را مشابه وضعیت مطلوب ادامه دهد.

برای مقابله با بروز خرابی در هر سیستمی دو رویکرد مختلف ارائه شده است. رویکرد اول تکنیک‌های پرهیز از وقایع نامطلوب^۵ می‌باشد. در این تکنیک‌ها با به کارگیری قطعات الکترونیکی کیفیت بالا در ساخت زیرسیستم‌ها، استفاده از تجهیزات پیشرفته در طراحی و ساخت آنها، آزمایش مرحله به مرحله در روند ساخت و رعایت دقیق قوانین طراحی، سعی می‌شود که تا حد امکان از ایجاد وقایع نامطلوب جلوگیری شود. باید توجه داشت برای عملیاتی کردن این تکنیک‌ها، شناسایی عوامل مخرب تأثیرگذار در وظیفه‌مندی مطلوب سیستم، میزان تأثیرگذاری هر کدام از این عوامل مخرب، شناخت دقیق وظیفه‌مندی تک تک اجزای سیستم و ... امری لازم و ضروری می‌باشد. اما باید توجه داشت که به دلیل افزایش شدید هزینه کلی شبکه با رعایت دقیق اصول ذکر شده و همچنین به دلیل بروز وقایع غیرقابل پیش‌بینی در محیط عملیاتی این شبکه‌ها، شاید نتوان فقط با تکیه بر این اصول به اتکاپذیری بالایی در این شبکه‌ها دست یافت. بنابراین رویکرد دوم مقابله با خرابی‌ها معروف به

رویکردهای تحمل‌پذیری خرابی مطرح می‌شوند. رویکردهایی که سعی می‌کنند بدون نیاز به کنترل کننده خارجی و به صورت خودسازمانده وجود هر گونه خرابی را پوشش داده و شبکه به وظیفه‌مندی مطلوب خود ادامه دهد [۱۲].

برای ایجاد یک سیستم تحمل‌پذیر خرابی از چهار نوع افزونگی، سخت‌افزاری، نرم‌افزاری، اطلاعاتی و زمانی استفاده می‌شود [۱۲]. دو نوع اول افزونگی به دلیل افزایش شدید هزینه و محدودیت سخت‌افزاری گره‌های حسگر، برای این شبکه‌ها مناسب نمی‌باشند. ولی دو نوع بعدی افزونگی به نوعی در رویکردهای مختلف تحمل‌پذیر خرابی این شبکه‌ها مورد استفاده قرار می‌گیرند. از رویکردهای متعددی که برای بهبود تحمل‌پذیری خرابی این شبکه‌ها ارائه شده‌اند می‌توان به تکنیک‌های ذخیره اطلاعات در گره‌های میانی شبکه^۶، روش‌های کنترل خطا، تکنیک‌های استفاده از چندین مسیر^۷ و ... اشاره کرد. تکنیک‌های ذخیره اطلاعات در گره‌های میانی موجب می‌شود تا در صورت گم شدن بسته در حین انتقال، گره مبدأ مجبور به ارسال مجدد آن شود و این گره‌های ذخیره کننده میانی هستند که بسته گم شده را مجدداً ارسال می‌کنند. واضح است که از افزونگی اطلاعاتی برای بهبود تحمل‌پذیری خرابی استفاده می‌شود؛ ولی محدودیت حافظه در گره‌های حسگر مشکلات خاصی را در نحوه انتخاب، تعداد و موقعیت این گره‌ها ایجاد می‌کند [۱۳]. روش‌های کنترل خطا با اضافه کردن بیت‌های کنترلی و استفاده از افزونگی اطلاعاتی سعی در کشف و در صورت امکان تصحیح خطاهای رخ داده دارند. با توجه به اینکه با افزایش تعداد بیت‌های کنترلی قابلیت کشف، تصحیح و انرژی مصرفی گره‌ها افزایش می‌یابد؛ تعداد بیت‌های کنترلی و نحوه الحاق آن به بسته‌های اطلاعاتی از چالش‌های اساسی تکنیک‌های مذکور می‌باشد [۱۴]. در تکنیک‌های چند مسیره با ارسال اطلاعات از چندین مسیر سعی در پوشش خرابی‌های احتمالی در گره‌ها و شکست‌های

احتمالی در پیوندها می‌شود. این تکنیک‌ها با ترکیبی از افزونگی اطلاعاتی و زمانی، تأثیر گره‌های معیوب را در عملکرد مطلوب شبکه کاهش می‌دهند [۱۵].

همه رویکردهای مذکور به نوعی سعی دارند تأثیر منفی گره‌های معیوب را کاهش دهند و هیچ‌گونه تلاشی برای حذف این گره‌ها انجام نمی‌دهند. یکی دیگر از رویکردهای رایج جهت بهبود تحمل‌پذیری خرابی این شبکه‌ها، مدیریت خرابی می‌باشد. در این رویکردها سعی می‌شود با کشف خرابی، موقعیت‌یابی و شناسایی دقیق گره‌های معیوب و حذف آنها، شبکه به وضعیت مطلوب برگردانده شود. مراجع مختلف مراحل متفاوتی را برای مدیریت خرابی ضروری دانسته‌اند. در [۱۶] مدیریت خرابی در پنج مرحله پیشگیری از خرابی^۸، تشخیص خرابی^۹، جداسازی خرابی^{۱۰}، تعیین نوع خرابی^{۱۱} و ترمیم خرابی^{۱۲} انجام می‌شود. در [۱۷] مدیریت خرابی در سه مرحله کشف خرابی، تشخیص نوع خرابی و بهبود شبکه انجام می‌شود. در [۱۸] مدیریت خرابی در دو مرحله کشف خرابی و بهبود شبکه خلاصه می‌شود. با مقایسه این سه مرجع مشاهده می‌شود که دو گام کشف خرابی و ترمیم شبکه از گام‌های اساسی و وجه تشابه این مراجع می‌باشد که ما در رویکرد پیشنهادی نیز بیشتر بر روی این دو گام متمرکز خواهیم شد.

گره‌ها به دلایل مختلف دچار خرابی می‌شوند. بالطبع اثر این عوامل مختلف بر روی حسگرها متفاوت خواهد بود. اثر خرابی‌ها از نظر میزان ماندگاری، نوع تأثیرگذاری، سطوح خرابی و از نظر کمیت خرابی قابل تقسیم‌بندی می‌باشند. خرابی‌ها از نظر ماندگاری به سه گروه خرابی‌های گذرا، متناوب و دائمی تقسیم بندی می‌شوند. نوع‌های بعدی خرابی در ادامه معرفی می‌شوند.

در حالت کلی اثر اشکال می‌تواند به صورت خرابی سخت و خرابی نرم در گره‌ها نمایان شود. شناسایی گره‌هایی که دچار خرابی سخت باشند بمراتب راحت و با یک پیغام درخواست و پاسخ صورت می‌گیرد. ولی

تصمیمات گرفته شده بررسی و وضعیت قطعی گره‌ها تعیین می‌شود. برای ارزیابی کارایی الگوریتم پیشنهادی، میزان دقت کشف خرابی و نرخ اخطار نادرست آن در شرایط مختلف مورد بررسی قرار گرفته است. شبیه‌سازی‌های انجام شده در نرم‌افزار متلب در چگالی‌های مختلف و با تعداد گره‌های همسایه مختلف مورد ارزیابی قرار گرفته است تا مقیاس‌پذیر بودن الگوریتم پیشنهادی قابل اثبات باشد. در ضمن اغلب الگوریتم‌های مدیریت خرابی، روند کشف خرابی را با فرکانس ثابت و در بازه زمان‌های مشخص انجام می‌دهند. در الگوریتم پیشنهادی یک رویکرد هوشمند به منظور تعیین زمان مناسب برای اجرای الگوریتم پیشنهادی معرفی می‌شود که به صورت پویا تعداد دفعات اجرای الگوریتم را تغییر می‌دهد. نتایج حاصل از به کارگیری این رویکرد بر روی انرژی مصرفی گره‌های شبکه در قسمت شبیه‌سازی این مقاله بررسی می‌شوند.

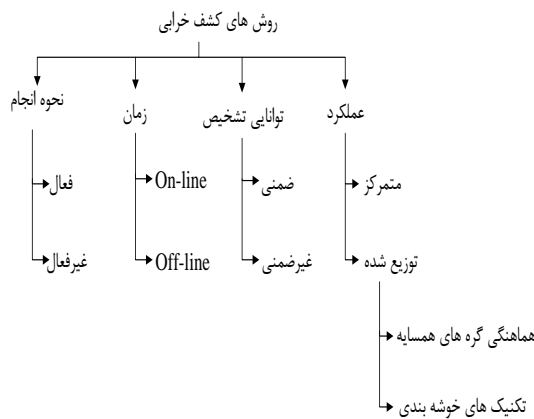
در ادامه این مقاله، در بخش دوم کارهای انجام شده در راستای مدیریت خرابی شبکه‌های حسگر بی‌سیم مورد بررسی قرار می‌گیرد. در بخش سوم تعاریف و فرضیاتی که برای تشریح الگوریتم پیشنهادی مورد استفاده قرار گرفته‌اند را معرفی می‌کنیم. جزئیات الگوریتم پیشنهادی و مراحل لازم برای تعیین وضعیت گره‌های حسگر در بخش چهارم این مقاله آورده شده است. در بخش پنجم نتایج حاصل از شبیه‌سازی‌ها مورد بررسی قرار گرفته و پارامترهای ارزیابی این الگوریتم را نسبت به الگوریتم‌های دیگر بررسی خواهیم کرد. نهایتاً در بخش ششم، نتیجه‌گیری و پیشنهادات برای کارهای آتی آورده شده است.

۲. کارهای انجام شده

تحمل‌پذیری خرابی به قابلیت از سیستم اشاره دارد که سیستم را قادر می‌سازد تا در صورت بروز خرابی بتواند آن را پوشش داده و فعالیت خود را مشابه وضعیت مطلوب ادامه دهد. هدف نهایی برای

برای شناسایی گره‌هایی که دچار خرابی نرم می‌باشند، رویکردهای پیچیده‌تری لازم است. در اغلب روش‌های کشف خرابی نرم برای تعیین وضعیت گره‌ها، داده دو حسگر همسایه S_i و S_j که به ترتیب با X_i و X_j نمایش داده می‌شوند با هم مقایسه می‌شوند [۲۳]. اگر اختلاف بین این داده‌ها از مقدار δ کمتر باشند $(|x_i^k - x_j^k| \leq \delta)$ وضعیت هر دو گره یکسان تعیین خواهد شد. داده‌های حس شده دو حسگر زمانی با هم مشابه خواهند بود که یا هر دو حسگر سالم و یا هر دو بنا به یک دلیل مشترک معیوب باشند که به خرابی ایجاد شده خرابی حالت مشترک^{۱۳} گفته می‌شود [۱۹]. در خرابی حالت مشترک، به دلیل یکسان بودن علت خرابی، داده حسگرها مشابه بوده و با رویکردهای مبتنی بر مقایسه تشخیص داده نشده و باعث کاهش کارایی این الگوریتم‌های می‌شوند. عوامل ایجاد کننده خرابی حالت مشترک را می‌توان به پنج گروه مختلف نقص طراحی تجهیزات^{۱۴}، عملکرد نامناسب^{۱۵}، اشتباه در نگهداری و عملکرد^{۱۶}، عوامل زیست محیطی^{۱۷} و پدیده‌های طبیعی^{۱۸} تقسیم‌بندی کرد. از آنجاییکه اغلب الگوریتم‌های مدیریت خرابی قادر به کشف خرابی‌های حالت مشترک نیستند. لذا چنین بنظر می‌رسد که برای تصمیم‌گیری درست در مورد وضعیت حسگرها باید به مقایسه داده حسگرهای همسایه اکتفا نکرد و باید تمهیداتی اندیشیده شود که مقایسه داده حسگر با داده حسگرهای غیر همسایه نیز مد نظر قرار گیرد. چرا که حسگرهای غیر همسایه چه بسا دچار خرابی حالت مشترک نشده باشند.

لذا در این مقاله در جهت پوشش مشکل مذکور، با ارائه یک رویکرد مدیریت خرابی سلسله مراتبی مبتنی بر خوشه‌بندی سعی می‌شود تا برای نائل شدن به یک تصمیم قطعی در مورد وضعیت گره‌ها به مقایسه محلی اکتفا نکرده و با تأثیر داده حسگرهای غیر همسایه تصمیم درستی اتخاذ شود. رویکرد پیشنهادی در یک حرکت پایین-بالا وضعیت احتمالی گره‌ها را تعیین می‌کند. سپس در یک حرکت بالا-بالا-پایین درستی



شکل ۲: روش‌های کشف خرابی در شبکه‌های حسگر بی‌سیم

در مرحله اول، هر گره S_i داده حس شده خود را با داده گره‌های همسایه، در زمان t ، مقایسه می‌کند. اگر اختلاف مقایسه در لحظه t از حد آستانه θ_1 بیشتر باشد، عمل مقایسه در لحظه $t+1$ دوباره تکرار می‌شود. حال اگر اختلاف مقایسه این داده‌ها در دو زمان مختلف، از حد آستانه θ_2 بیشتر باشد $c_{ij}=1$ می‌شود. به این معنی که داده این دو گره باهم مشابه نیستند. در مرحله دوم، هر گره S_i وضعیت احتمالی خود را بر اساس داده گره‌های همسایه‌اش تعیین می‌کند. به این صورت که به ازای تمام همسایه‌های S_i اگر مجموع C_{ij} های گره S_i ، از نصف تعداد همسایه‌هایش کمتر باشد، گره S_i احتمالاً سالم^{۲۳} و در غیر این صورت احتمالاً معیوب^{۲۴} تشخیص داده خواهد شد. در مرحله سوم، گره‌ی که احتمالاً سالم شناسایی شده است، در صورتیکه حداقل نصف تعداد همسایه‌هایش احتمالاً سالم باشند، به عنوان گره قطعاً سالم^{۲۵} شناخته شده و وضعیت قطعی خود را به گره‌های همسایه، به صورت انتشار همگانی ارسال می‌کند تا دیگر گره‌ها نیز وضعیت قطعی خودشان را تعیین کنند [۲۲].

تحمل‌پذیری خرابی، ساخت سیستم‌های قابل اتکا با کیفیت مورد قبول کاربر است. قابلیت اتکا^۹، توانایی یک سیستم برای ارائه و تحویل سرویس مورد نظر به کاربران است، بگونه‌ای که بتوان از ارائه آن سرویس اطمینان داشت. قابلیت اطمینان R_k یا تحمل‌پذیری خطای یک گره حسگر با توزیع پواسن^{۲۰} مدل می‌شود. این توزیع، احتمال نداشتن خرابی در بازه زمانی $(0,t)$ را نشان می‌دهد و با رابطه (۱) محاسبه می‌شود.

$$R(t) = \int_{-\infty}^{+\infty} \lambda_k e^{-\lambda_k \tau} d\tau \quad (1)$$

که λ_k نرخ خرابی گره حسگر k ام و t مدت زمان سالم ماندن گره را نشان می‌دهد. با فرض $\lambda > 0$ ساده شده رابطه (۱) به صورت رابطه (۲) خواهد بود.

$$R(t) = \int_t^{+\infty} \lambda_k e^{-\lambda_k \tau} d\tau = \lambda_k \left. \frac{1}{-\lambda_k} e^{-\lambda_k \tau} \right|_t^{+\infty} \\ = (-1)(e^{-\lambda_k(\infty)} - e^{-\lambda_k t}) = e^{-\lambda_k t} \quad (2)$$

تحمل‌پذیری خرابی شبکه حسگر بی‌سیم در چهار لایه، سخت‌افزار، نرم‌افزار، ارتباطی شبکه و لایه کاربردی مطرح می‌شود [۲۰]. یکی از رویکردهای رایج برای افزایش تحمل‌پذیری خرابی، مدیریت خرابی می‌باشد. مدیریت خرابی در شبکه‌های حسگر بی‌سیم شامل سه مرحله کشف خرابی^{۲۱}، موقعیت‌یابی و تعیین گره‌های خراب^{۲۲} و حذف گره‌های معیوب و بازیابی شبکه می‌باشد. تقسیم‌بندی‌های متعددی برای تکنیک‌های کشف خرابی از نظر نحوه عملکرد، توانایی تشخیص، زمان کشف خرابی و نحوه تشخیص ارائه شده است که این تقسیم‌بندی‌ها در شکل نمایش داده شده است.

الگوریتم دیگری بنام الگوریتم Chen نیز برای کشف خرابی ارائه شده است. روند الگوریتم Chen مشابه الگوریتم Lee است با این تفاوت که در این الگوریتم برای رسیدن به یک تصمیم قطعی در مورد وضعیت گره‌ها، دو مرحله عمل مقایسه بین گره‌ها صورت می‌گیرد.

۳. تعاریف و فرضیات

قبل از تشریح الگوریتم پیشنهادی، ابتدا در این قسمت تعاریف و فرضیاتی که برای تشریح الگوریتم پیشنهادی ضروری هستند را معرفی می‌کنیم.

\mathbf{n} : تعداد کل گره‌های حسگر پخش شده در محیط
 \mathbf{r} : تعداد حسگرهای همسایه هر گره

S : مجموعه تمام حسگرهای شبکه، به طوری که
 $S = \{S_1, S_2, S_3, \dots, S_n\}$

w : متوسط تعداد حسگرهای هر خوشه

P_n : احتمال خرابی هر گره حسگر

N_{ch} : تعداد سرخوشه‌های همسایه هر سرخوشه

$N(S_i)$: مجموعه همسایه‌های گره S_i

$N(S_{ch})$: مجموعه سرخوشه‌های همسایه سرخوشه S_{ch}

$O(S_{chi})$: مجموعه گره‌های عضو سرخوشه S_{chi}

\mathcal{X}_i^t : داده‌های حس شده گره S_i در زمان t و داده تولید شده در تجمیع داده گره S_{chi}

c_{ij} : نتیجه مقایسه داده‌های حس شده گره‌های S_i و

S_j یا S_{chi} و S_{chj} به طوری که $c_{ij} \in \{0, 1\}$, $c_{ij} = c_{ji}$

$|C_i|$: تعداد صفرهای C_{ij}

θ_3, θ_4 : مقادیر آستانه از پیش تعیین شده

T_i : وضعیت گره، به طوری که

$T_i \in \{LF, LG, GD, FT\}$

NT_i : جدول مجاورت گره که شامل گره‌های همسایه

گره i و C_{ij} مربوط به آنها است.

q : عدد کوچک و مثبت برای افزونگی زمانی

v : شمارنده

M : ماتریس افزونگی به ابعاد $r \times q$ برای ذخیره‌سازی

نتایج q بار مقایسه داده‌های گره i با r گره همسایه، به

طوری که $M = [m_{ij}]$, $m_{ij} \in \{0, 1\}$

d_{ij}^t : تفاضل داده دو حسگر S_i و S_j در زمان t ، به

طوری که $d_{ij}^t = X_i^t - X_j^t$

Δt_l : اختلاف زمانی، به طوری که $\Delta t_l = t_{l+1} - t_l$

$\Delta d_{ij}^{\Delta t_l}$: تفاضل داده دو گره S_i و S_j از زمان t_l و t_{l+1} به طوری که

$$\Delta d_{ij}^{\Delta t_l} = d_{ij}^{t_{l+1}} - d_{ij}^{t_l} = (X_{chi}^{t_{l+1}} - X_{chj}^{t_{l+1}}) - (X_{chi}^{t_l} - X_{chj}^{t_l})$$

مقادیر حد آستانه θ_1 و θ_2 را می‌توان با استفاده از روابط (۳) و (۴) محاسبه نمود.

$$\theta_1 = \lceil w/2 \rceil \quad (۳)$$

(۴)

$$\theta_2 = \lceil q/2 \rceil$$

۴. ارائه الگوریتم پیشنهادی

الگوریتم‌های کشف خرابی از نظر نحوه عملکرد به دو گروه الگوریتم‌های کشف خرابی متمرکز و توزیع شده تقسیم‌بندی می‌شوند. رویکردهای متمرکز به دلیل عدم تعادل بار و انرژی، سربار بالای بسته‌های کنترلی رد و بدل شده و عدم وجود یک گره با توانایی‌های بالا به عنوان مدیر شبکه برای شبکه‌های حسگر بی‌سیم توصیه نمی‌شوند. رویکردهای توزیع

شده به دو صورت هماهنگی با گره‌های همسایه و روش‌های مبتنی بر خوشه‌بندی انجام می‌شوند. در شبکه‌های خوشه‌بندی شده مدیریت خرابی به صورت محلی و مجزا در هر خوشه صورت می‌گیرد. در چنین رویکردهایی سرخوشه‌ها می‌توانند گره‌های عضو را تحت نظر داشته باشند تا در صورت تشخیص گره‌های معیوب خوشه، آن را به گره‌های دیگر خوشه اعلام کرده و داده‌های رسیده از آنها را نادیده بگیرند.

همچنین در رویکردهای خوشه‌بندی شده، گره‌های غیر سرخوشه می‌توانند سرخوشه مربوطه را تحت نظر گرفته و در صورت معیوب تشخیص دادن سرخوشه، به سرخوشه‌های همسایه بیوندند. در حالت سوم می‌توان از گره‌های ویژه‌ای که دروازه نامیده می‌شود، برای

مدیریت خرابی خوشه استفاده کرد. در صورت معیوب تشخیص داده شدن تعداد معینی از گره‌های خوشه و ارسال پیغام هشدار از طرف گره دروازه، خوشه‌بندی مجدد در این شبکه‌ها صورت می‌گیرد.

در تکنیک‌های مدیریت خرابی مبتنی بر خوشه-بندی تصمیم‌گیری در مورد وضعیت گره‌ها به صورت محلی و براساس داده‌های دریافتی از گره‌های داخل هر خوشه صورت می‌گیرد. اگرچه ساده بودن و پایین بودن ترافیک ناشی از تبادل پیغام‌های وضعیتی از مزایای این روش‌ها محسوب می‌شود؛ ولی بروز یک خرابی از نوع خرابی حالت مشترک در هر خوشه، تمامی تصمیمات آنها را با اشتباه همراه خواهد کرد. لذا ایجاد ارتباط مابین خوشه‌های مختلف برای تصمیم‌گیری درست در مورد وضعیت گره‌های حسگر کاملاً ضروری بنظر می‌رسد. در رویکرد پیشنهادی به منظور ارتباط هر خوشه با خوشه‌های دیگر استفاده از رویکرد سلسله مراتبی را پیشنهاد می‌کنیم. رویکرد سلسله مراتبی پیشنهادی از سه لایه مجزا تشکیل می‌شود. در معماری پیشنهادی همه گره‌های شبکه در لایه اول خوشه‌بندی شده و برای هر خوشه نیز سرخوشه تعیین می‌شود. به این ترتیب امکان مدیریت خرابی به صورت محلی و داخل خوشه‌ای قابل انجام می‌باشد. اما برای ارتباط بین خوشه‌ای، سرخوشه‌های هر خوشه، دوباره در لایه دوم نیز خوشه‌بندی می‌شوند. به این ترتیب مقایسه بین داده گره‌های خوشه‌های مجاور فراهم خواهد شد. برای افزایش پوشش الگوریتم پیشنهادی و کاهش بیشتر اثر خرابی‌های حالت مشترک خوشه‌بندی را در لایه سوم نیز ادامه می‌دهیم. به همین ترتیب مقایسه داده هر حسگر تقریباً با اکثر گره‌های شبکه فراهم خواهد شد. نمونه‌ای از خوشه-بندی سلسله مراتبی سه لایه‌ای^{۲۶} در شکل ۳ **Error!** **Reference source not found.** نشان داده شده است.

در الگوریتم پیشنهادی برای پوشش خرابی‌های گذرا، داده دو حسگر همسایه، q بار مقایسه می‌شود.

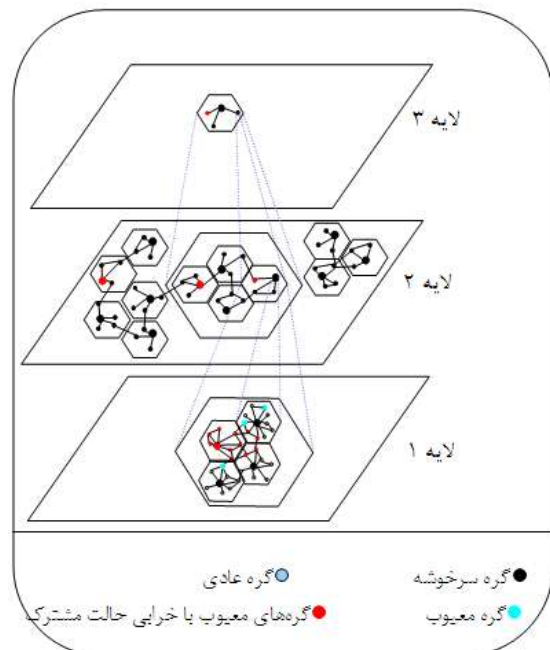
تا بروز یک خرابی گذرا نتیجه مقایسه را تحت تأثیر قرار ندهد. بنابراین در الگوریتم پیشنهادی از افزونگی زمانی برای مقابله با خرابی‌های گذرا استفاده می‌شود. خرابی‌ها از نظر کمیت به سه گروه جزئی، کلی و بزرگ تقسیم‌بندی می‌شوند. از آنجاییکه در الگوریتم پیشنهادی مقایسه بین داده خوشه‌های غیر مجاور هم فراهم شده است لذا انتظار می‌رود قادر باشد علاوه بر خرابی‌های جزئی، خرابی‌های بزرگ و تا حدودی خرابی‌های کلی را نیز پوشش دهد. در الگوریتم پیشنهادی ابتدا با یک حرکت پایین-بالا وضعیت احتمالی گره‌های حسگر تعیین می‌شود. سپس با مقایسه دوباره در لایه بالاتر و اطمینان از تصمیم‌های قبلی، در یک حرکت بالا به پایین وضعیت قطعی حسگرها معلوم خواهد شد. در مجموع روند اجرای الگوریتم پیشنهادی در شش مرحله خلاصه می‌شود که در ادامه با جزئیات بیشتر این مراحل را توضیح می‌دهیم

مرحله اول: در این مرحله و در لایه یک، تمامی حسگرها پس از شناسایی همه حسگرهای همسایه، جدول همسایگی NT_i را که شامل مشخصه‌های r حسگر همسایه می‌باشد را تولید می‌کنند. در این مرحله هر حسگر ماتریس افزونگی مربوط به خود را ایجاد کرده و تمامی فیلدهای آن را برابر یک قرار می‌دهد. قبل از شروع روند کشف خرابی هر حسگر وضعیت اولیه خود را احتمالاً معیوب فرض کرده و $T_i = LF$ می‌شود. پس از شروع الگوریتم کشف خرابی، گره حسگر S_i ، داده حس شده‌اش را با r گره همسایه به تعداد q بار مقایسه کرده و نتایج حاصل در ماتریس افزونگی M به ابعاد $r \times q$ ذخیره می‌شود. با فرض اینکه x_i^k و x_j^k به ترتیب مقادیر حس شده گره‌های S_i و S_j در زمان t^k باشند؛ اگر در هر بار مقایسه، داده این دو گره باهم برابر باشند^{۲۷} یا به عبارتی اختلاف مقادیر حس شده آنها از مقدار δ کمتر باشد ($|x_i^k - x_j^k| \leq \delta$)، در این صورت $m_{jk} = 0$ و در صورت

دوباره داده گره‌های این لایه با داده گره‌های همسایه‌اش مقایسه می‌شوند. برای اطمینان از بی‌اثر بودن خرابی‌های گذرا در عملکرد الگوریتم پیشنهادی در لایه دوم مقایسه داده حسگرهای این لایه دوبار انجام می‌شود. اگر $d_{ij}^t = |X_{chi}^t - X_{chj}^t| \leq \theta_3$ باشد، $C_{ij}=0$ در نظر گرفته می‌شود. اما اگر شرط فوق برقرار نباشد داده دو حسگر S_{chi} و S_{chj} در لحظه $t+1$ محاسبه شده و در d_{ij}^{t+1} قرار داده می‌شود. با محاسبه اختلاف d_{ij}^t و d_{ij}^{t+1} وضعیت و رابطه داده این دو حسگر نسبت به هم معلوم می‌شود. اگر اختلاف این دو متغیر از مقدار θ_4 بیشتر باشد به منزله عدم مشابهت داده این دو حسگر و $C_{ij}=1$ خواهد شد. در گام بعدی، هر گره S_{chi} وضعیت احتمالی خود را براساس نتایج حاصل از گام اول این مرحله انجام می‌دهند. برای این منظور اگر داده هر حسگر با بیش از نصف گره‌های همسایه مشابه باشد وضعیت گره، احتمالاً سالم و در غیر این صورت احتمالاً معیوب تشخیص داده می‌شود. برای تعیین وضعیت سرخوشه لایه دوم نیز مشابه گام دوم مرحله اول عمل می‌شود. به صورتی که اگر داده حس شده سرخوشه این لایه حداقل با θ_1 تا از داده گره‌های خوشه شباهت داشته باشد، احتمالاً سالم و در غیر این صورت احتمالاً معیوب تعیین خواهد شد. همان طور که دیده می‌شود هیچ تصمیمی قطعی در مورد وضعیت گره‌ها اتخاذ نشده و فقط رابطه بین داده حسگرها تعیین می‌شود.

مرحله سوم: در آخرین لایه از مدل سلسله مراتبی، گره‌های سرخوشه لایه دوم تشکیل یک خوشه در لایه سوم را می‌دهند. در این لایه مجدداً پس از دوبار مقایسه و محاسبه $\Delta d_{ij}^{At_1}$ ، وضعیت داده هر حسگر نسبت به داده حسگرهای همسایه‌اش تعیین می‌شود. این روند مشابه گام اول و دوم لایه دوم انجام می‌شود. در این مرحله گره‌هایی که احتمالاً سالم تشخیص داده شده‌اند و حداقل نصف حسگرهای همسایه آنها نیز احتمالاً سالم شناسایی شده باشد به عنوان گره‌های قطعاً سالم شناخته می‌شوند. گره‌هایی

عدم مشابهت $m_{jk}=1$ خواهد شد. تعداد صفرهای موجود در هر سطر ماتریس M تعیین‌کننده وضعیت C_{ij} یعنی ارتباط بین گره S_i و S_j می‌باشد. اگر تعداد صفرها بیشتر از θ_2 باشد C_{ij} برابر صفر می‌شود در غیر این صورت C_{ij} برابر یک می‌شود. سپس هر گره سرخوشه S_{chi} وضعیت احتمالی خود را به این صورت تعیین می‌کنند که اگر داده S_{chi} حداقل با θ_1 تا از گره‌های عضو خوشه یکسان باشد؛ وضعیت گره S_{chi} احتمالاً سالم ($T_{chi}=LG$) و در غیر این صورت احتمالاً معیوب ($T_{chi}=LF$) تعیین خواهد شد. همان طور که واضح است در این مرحله فعلاً وضعیت احتمالی گره‌های حسگر و سرخوشه‌ها تعیین می‌شود.



شکل ۳: شبکه حسگر بی‌سیم با خوشه‌بندی سلسله مراتبی سه لایه‌ای

مرحله دوم: گره‌های موجود در لایه دوم همان گره‌های سرخوشه لایه اول می‌باشند. در این مرحله سعی می‌شود با مقایسه داده‌های سرخوشه‌ها بتوان تصمیمات دقیق‌تری در مورد وضعیت آنها اتخاذ کرد. لذا در این مرحله و در لایه دوم گره‌هایی که در مرحله اول احتمالاً سالم تشخیص داده شده‌اند احتمال سرخوشه شدن آنها افزایش می‌یابد. در این مرحله

که وضعیت قطعی آنها تعیین شده باشد وضعیت خودشان را به کلیه گره‌های لایه سوم انتشار همگانی می‌کنند. به این ترتیب وضعیت همه گره‌های موجود در لایه سوم تعیین خواهد شد. برای تعیین وضعیت گره‌های این لایه به این صورت عمل می‌شود که اگر گره S_{chj} که وضعیت فعلی آن احتمالاً سالم یا احتمالاً معیوب باشد و با گره قطعاً سالم S_{chi} همسایه باشد و داده تولید شده آنها یکسان و $C_{ij}=0$ باشد گره S_{chj} به عنوان گره قطعاً سالم شناسایی خواهد شد. ولی در صورت مغایرت داده آنها و اگر $C_{ij}=1$ باشد، وضعیت گره S_{chj} قطعاً معیوب خواهد بود. این مرحله L بار و به صورت موازی تکرار می‌شود تا وضعیت همه حسگرها تعیین شود. در پایان مرحله سوم وضعیت قطعی کلیه گره‌های لایه سوم تعیین می‌شود. به این ترتیب مرحله پایین-بالا به پایان می‌رسد.

مرحله چهارم: در این مرحله گره‌های لایه سوم وضعیت قطعی‌شان را به لایه دوم ارسال می‌کنند تا وضعیت گره‌های این لایه تعیین شود. روند تعیین وضعیت قطعی گره‌ها مشابه گام دوم لایه سوم می‌باشد.

مرحله پنجم: در این مرحله، این روند برای تعیین وضعیت نهایی گره‌های لایه یک نیز تکرار شده و در نهایت وضعیت قطعی تمامی گره‌ها با ضریب اطمینان بسیار بالایی تعیین می‌شود.

مرحله ششم: این مرحله یک مرحله تکمیلی می‌باشد که اگرچنانچه وضعیت نهایی گره‌ها در پنج مرحله مذکور تعیین نشود، در این مرحله وضعیت قطعی آن تعیین گردد. اگر چنانچه دو گره S_j و S_k همسایه‌های گره فرضی S_i باشند به طوری که $T_j = T_k=0$ و مقدار C_{ij} مخالف مقدار C_{ki} باشد؛ در این صورت اگر وضعیت گره S_i احتمالاً سالم باشد، وضعیت نهایی آن به قطعاً سالم و اگر وضعیت آن احتمالاً معیوب باشد، وضعیت نهایی آن به قطعاً معیوب تغییر پیدا می‌کند.

به این ترتیب الگوریتم پیشنهادی ابتدا در حرکت

پایین-به-بالا وضعیت احتمالی گره‌ها و در حرکت بالا-به-پایین وضعیت قطعی گره‌های شبکه را معلوم می‌کند. می‌توان پیش‌بینی کرد که به دلیل خوشه‌بندی سلسله مراتبی و مقایسه سه مرحله‌ای مصرف انرژی این الگوریتم نسبتاً بالا باشد. لذا چالش بعدی در راستای بهبود این الگوریتم، تلاش برای کاهش مصرف انرژی آن می‌باشد. از آنجاییکه مقدار قابل توجهی از انرژی گره‌ها برای عملیات خوشه‌بندی و تعیین سرخوشه‌ها مصرف می‌شود که با توجه به ساختار رویکرد پیشنهادی، امکان کاهش در این مرحله از الگوریتم وجود ندارد. لذا برای کاهش مصرف انرژی باید در تعداد دفعات انجام این الگوریتم و تعداد بروزرسانی‌های ماتریس‌ها، رویکردهای هوشمند به کار گرفته شود. در اغلب روش‌های مدیریت خرابی، روند کشف خرابی با فرکانس ثابت و در بازه زمان‌های مشخص صورت می‌گیرد. لذا برای کاهش مصرف انرژی، می‌توان یک رویکرد پویا به کار گرفت که الگوریتم پیشنهادی را در مواقعی که احتمال وقوع خرابی وجود دارد، اجرا کند. در ضمن از آنجایی که بروزرسانی جداول همسایگی و ماتریس‌های افزونگی در هر بار اجرای الگوریتم غیر ضروری و نیاز به انرژی مصرفی بالایی دارند؛ قصد داریم با تخمین زمان مناسب برای کشف خرابی از تعداد دفعات اجرای الگوریتم کاسته شود. ما این پارامتر را میانگین زمان مابین خرابی‌ها $^{28}(MTBF)$ می‌نامیم. حال سؤالی که پیش می‌آید اینست که چه زمانی برای شروع روند کشف خرابی مناسب است؟ برای تعیین زمان مناسب برای کشف خرابی، بیان یک سری فرضیات و تعدادی مقادیر آستانه از پیش تعیین شده لازم است که در ادامه معرفی می‌شوند.

داده‌های تولید شده گره حسگر i در طول زمان t به صورت تابع $\delta(i, t) = F(v_1, v_2, \dots, v_k)$ نشان داده می‌شود که v_i نشان دهنده یکی از داده‌های تولید شده این گره در زمان تعیین شده می‌باشد. همان طور که قبلاً نیز گفته شده است، یکسری از عوامل شناخته

و ناشناخته می‌توانند بر روی داده‌های تولید شده حسگرها تأثیر بگذارند. از آنجاییکه این عوامل در طول زمان متغیر بوده و دارای رفتاری پویا و غیرخطی می‌باشند؛ لذا مدل کردن این عوامل در اغلب کاربردها غیر ممکن خواهد بود. بنابراین بجای تلاش برای به دست آوردن داده واقعی گره حسگر می‌توان با تجزیه و تحلیل داده‌های تولید شده $(\delta(i,t))$ ، خواص پایه‌ای تابع F را فرموله کرد. ما قصد داریم با فرموله کردن و تعیین روابط مشخص بین داده‌های تولید شده یک گره حسگر، تقریباً میانگین زمان مابین خرابی‌ها را تشخیص دهیم. داده تولید شده هر گره حسگر دارای سه فرض می‌باشد. اولاً داده تولید شده هر حسگر $(\delta(i,t))$ ، کاملاً مستقل از بقیه حسگرها می‌باشد. دوماً دو عدد ثابت به عنوان کران بالا و کران پائین که به ترتیب با c_{\min} و c_{\max} نشان داده می‌شوند؛ وجود دارد که داده حسگر سالم در بین این دو کران قرار می‌گیرد $(c_{\min} \leq \delta(i,t) \leq c_{\max})$. ثالثاً داده تولید شده $(\delta(i,t))$ در بازه $[c_{\min}, c_{\max}]$ به صورت آماری پیوسته بوده و دارای یک توزیع احتمال مانند توزیع نرمال می‌باشد. بنابراین یک تابع چگالی احتمال پیوسته مانند $\varphi(i)$ می‌تواند برای نشان دادن این توزیع احتمال استفاده شود. لازم بذکر است برای کاربردهای مختلف بازه $[c_{\min}, c_{\max}]$ و $\varphi(i)$ متفاوت خواهد بود. با توجه به مطالب ذکر شده، داده مورد انتظار از یک گره حسگر یا میانگین داده تولید شده توسط حسگر که با $E(i)$ نشان داده می‌شود از رابطه (۵) به دست می‌آید.

اگر داده حسگر در بین c_{\min} و c_{\max} قرار بگیرد یعنی $c_{\min} \leq \delta(i,t) \leq c_{\max}$ و دارای توزیع نرمال $\varphi(i)$ باشد؛ این داده جزء داده عادی فرض می‌شود و نشان دهنده سالم بودن حسگر خواهد بود. اگر داده حسگر در رابطه $|\delta(i,t) - E(i)| \geq \tau_1$ به طوری که $\tau_1 > 0$ صدق کند؛ نشان دهنده اینست که داده حسگر در بازه رخداد عادی شبکه قرار دارد. در این حالت نیز هیچ حساسیتی برای رخداد خرابی وجود ندارد.

اگر داده حسگر در یکی از سه شرط زیر صدق کند به احتمال زیاد دچار خرابی شده است.

اگر رابطه $\forall t, |\delta(i,t) - E(i)| > \tau_2$ به طوری که $0 < \tau_1 < \tau_2$ برقرار باشد؛ به این معنی است که داده تولید شده، خارج از محدوده داده عادی و داده رخداد شبکه بوده و به احتمال زیاد این حسگر دچار خرابی شده است.

اگر در چند بازه پی‌درپی، برای داده حسگر i ، رابطه $|\delta(i,t) - \delta(i,t+1)| \approx 0$ و برای داده حسگر j که همسایه حسگر i می‌باشد، رابطه $|\delta(j,t) - \delta(j,t+1)| > c$ برقرار باشد؛ به این معنی است که داده حسگر j در طول زمان متغیر، ولی داده حسگر i در طول زمان ثابت بوده است. رخداد چنین حالتی احتمال وقوع خرابی حسگر i را افزایش می‌دهد. اگر در داده تولید شده حسگر روابط $c_{\min} < \delta(i,t+2) < c_{\max}$ ، $\delta(i,t+3) > \tau_2$ و $\forall t, c_{\min} < \delta(i,t) < c_{\max}$ ، $\delta(i,t+1) > \tau_2$ صدق کند؛ بمنزله تولید داده تصادفی در این حسگر قلمداد شده و احتمال معیوب بودن آنرا افزایش می‌دهد.

بنابراین تا زمانی که اغلب داده حسگرها در حالت اول یا دوم قرار می‌گیرند نیازی به اجرای الگوریتم پیشنهادی وجود ندارد. لذا با این تصمیم ساده می‌توان به مقدار قابل توجهی از تعداد دفعات اجرای الگوریتم در شرایط غیر ضروری کاهش داد. ولی اگر چنانچه تعداد معینی از داده حسگرها در حالت سوم قرار

$$E(i) = \int_{c_{\min}}^{c_{\max}} \delta(i,t) \times \varphi(i) d(t) \quad (5)$$

بنابراین با مطالب ذکر شده، داده تولید شده یک گره حسگر در یکی از سه حالت زیر قرار می‌گیرد که بر اساس آن می‌توان وضعیت احتمالی حسگر را تخمین زد.

جدول (۱): پارامترهای استفاده شده در شبیه‌سازی

پارامترها	مقادیر
E_{elec}	50 nJ/bit
E_{DA}	5 nJ/bit/signal
\mathcal{E}_{fs}	10 pJ/bit/m ²
\mathcal{E}_{mp}	0.0013 pJ/bit/m ⁴
اندازه بسته داده	8192 bits
d_0	87 m
انرژی اولیه گره‌ها	۳ ژول

مدل مصرف انرژی در شبکه‌های حسگر ارتباط مستقیمی با نحوه طراحی زیرلایه کنترل دسترسی به رسانه^{۲۹} در این شبکه‌ها دارد. اما اگر بخواهیم از مدل مشترکی که مستقل از پارامترهای طراحی تعریف شده در زیرلایه کنترل دسترسی به رسانه باشد از روابط ذیل جهت مدل کردن میزان مصرف انرژی در این شبکه‌ها استفاده می‌کنیم. در الگوریتم پیشنهادی مدل مصرف انرژی همانند مدل مصرف انرژی Leach در نظر گرفته شده است که هر دو مدل کانال فضای باز (اتلاف انرژی d^2) و چند مسیره (اتلاف انرژی d^4) را با توجه به مسافت میان فرستنده و گیرنده، بکار می‌برد. در نتیجه انرژی مصرف شده برای ارسال یک بسته l بی‌تی در مسافت d با استفاده از رابطه (۶) به دست می‌آید.

$$E_{Tx}(l, d) = E_{Tx-elec}(l) + E_{Tx-amp}(l, d) = \begin{cases} lE_{elec} + l_{fs}d^2 & d < d_0 \\ lE_{elec} + l_{smp}d^4 & d \geq d_0 \end{cases} \quad (6)$$

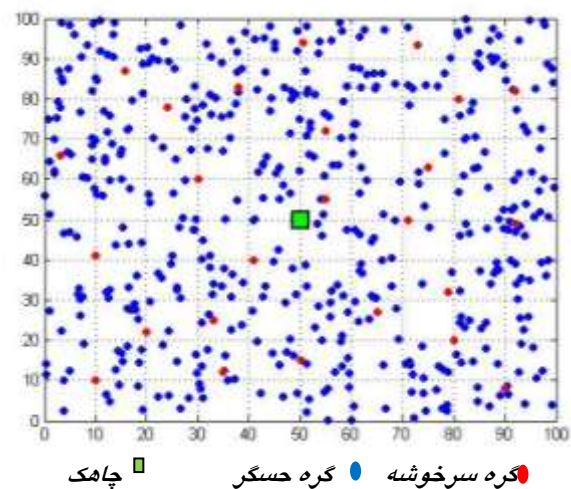
در این رابطه، E_{elec} انرژی لازم برای فعالسازی مدارات الکترونیکی است، \mathcal{E}_{fs} و \mathcal{E}_{amp} به ترتیب انرژی‌های لازم جهت تقویت سیگنال‌های ارسالی برای انتقال یک بیت در مدل فضای باز و مدل چند مسیره هستند. d فاصله بین گره‌ها و d_0 نیز مقدار آستانه مسافت است که از رابطه (۷) به دست می‌آید.

$$d_0 = \sqrt{\frac{\mathcal{E}_{fs}}{\mathcal{E}_{amp}}} \quad (7)$$

بگیرند روند کشف خرابی با رویکرد پیشنهادی انجام خواهد شد.

۵. شبیه‌سازی و ارزیابی کارایی الگوریتم پیشنهادی

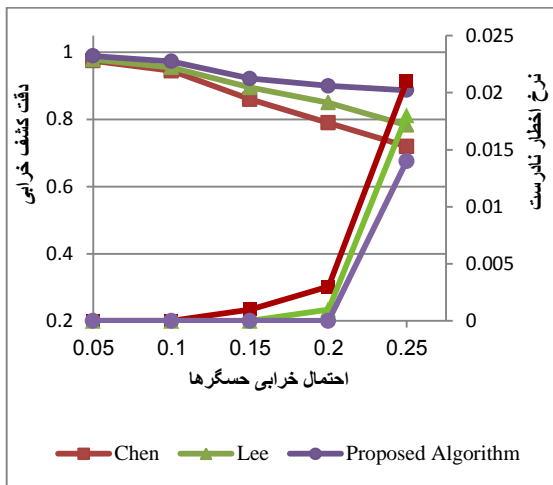
ما یک شبکه حسگر بی‌سیم را در محیطی به ابعاد 100×100 متر مربع، مطابق شکل ۴ و با توزیع یکنواخت و تصادفی ۵۱۲ گره حسگر، با نرم‌افزار متلب شبیه‌سازی کردیم. در شبیه‌سازی انجام شده P_n برای گره‌های حسگر به ترتیب با احتمال‌های ۰,۱، ۰,۰۵، ۰,۱۵، ۰,۲، ۰,۲۵ در نظر گرفته شده است. میانگین تعداد گره‌های هر خوشه نیز به ترتیب ۷ و ۱۰ در نظر گرفته شده است. بنابراین در نحوه خوشه‌بندی و تعیین تعداد سرخوشه‌ها دقت شده که از افزایش تعداد حسگرها در هر خوشه جلوگیری بعمل بیاید.



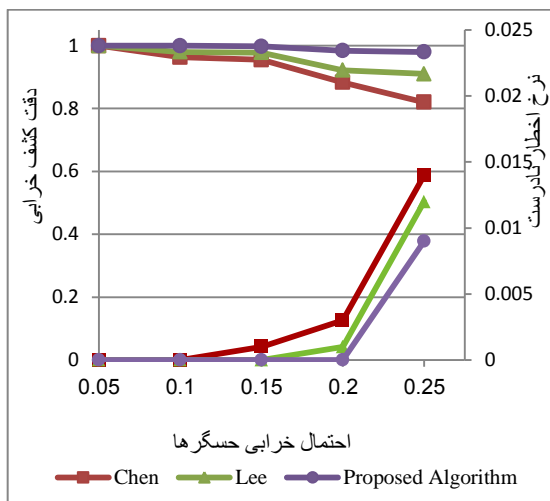
شکل ۴: توزیع یکنواخت و تصادفی ۵۱۲ گره حسگر در محیطی به ابعاد 100×100 مترمربع

در این مدل شبیه‌سازی شده گره چاهک ثابت و در موقعیت (۵۰ و ۵۰) در نظر گرفته شده است. شبیه‌سازی در طول ۱۰۰۰ دوره تکرار شده و انرژی مصرفی براساس مقادیر جدول (محاسبه شده است).

مقایسه قرار گرفته‌اند.



شکل ۵: DA و FAR الگوریتم پیشنهادی با $w=7$ و $D=1$



شکل ۶: DA و FAR الگوریتم پیشنهادی با $w=10$ و $D=1$

جدول (۲) مقادیر عددی حاصل از مقایسه این الگوریتم‌ها از نظر دقت کشف خرابی و نرخ اخطار نادرست را زمانی که تنوع خرابی برابر یک باشد را نشان می‌دهد. نتایج حاصل نشان می‌دهند زمانی که احتمال خرابی حسگرها برابر $P_n=0,1$ و $W=7$ می‌باشد؛ دقت کشف خرابی در الگوریتم‌های Chen و Lee به ترتیب برابر $0,865$ و $0,925$ ولی در الگوریتم

انرژی مصرف شده برای دریافت یک بسته l بیتی نیز از رابطه (۸) به دست می‌آید.

$$E_{Rx}(l) = E_{Rx-elec}(l) = lE_{elec} \quad (8)$$

برای بررسی کارایی الگوریتم پیشنهادی سعی شده است نتایج حاصل از شبیه‌سازی‌ها نسبت به دو الگوریتم Lee و Chen مورد ارزیابی قرار بگیرد. از پارامترهای دقت کشف خرابی^{۳۰} و نرخ اخطار نادرست^{۳۱} برای مقایسه این الگوریتم‌ها استفاده شده است. دقت کشف خرابی نشان دهنده تعداد گره‌های معیوب تشخیص داده شده به کل گره‌های معیوب موجود در شبکه می‌باشد. بنابراین اگر α نشان دهنده تعداد گره‌های معیوب کشف شده باشد؛ مقدار دقت کشف خرابی از رابطه (۹) به دست می‌آید.

$$\text{دقت کشف خرابی} = \frac{\alpha}{np} \quad (9)$$

نرخ اخطار نادرست نشان دهنده نسبت تعداد گره‌های سالمی که معیوب تشخیص داده شده‌اند به کل تعداد گره‌های سالم در شبکه می‌باشد. بنابراین اگر β نشان دهنده تعداد گره‌های سالمی که معیوب تشخیص داده شده باشند باشد؛ نرخ اخطار نادرست از رابطه (۱۰) به دست می‌آید.

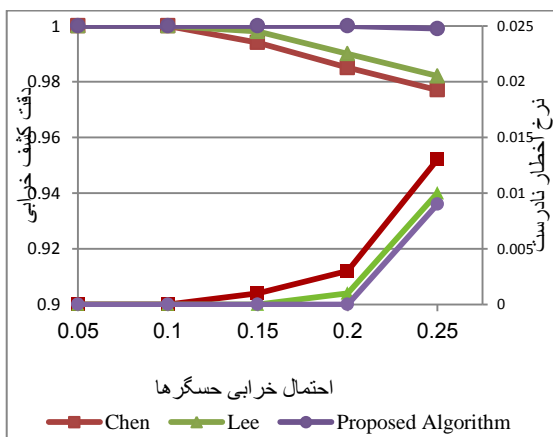
$$\text{نرخ اخطار نادرست} = \frac{\beta}{n(1-P)} \quad (10)$$

برای ارزیابی بهتر الگوریتم پیشنهادی تعداد گره‌های هر خوشه را متغیر فرض کرده‌ایم. همچنین اجرای الگوریتم در حالتی که تنوع^{۳۲} خرابی برابر یک ($D=1$) و یا برابر دو ($D=2$) باشد تکرار شده تا عملکرد این الگوریتم در حالت‌های مختلف نسبت به دو الگوریتم Lee و Chen ارزیابی شود. شکل ۵ و ۶ مقایسه الگوریتم پیشنهادی را نسبت به الگوریتم‌های Lee و Chen نشان می‌دهند. این الگوریتم‌ها از نظر دقت کشف خرابی و نرخ اخطار نادرست در حالتی که فقط یک نوع خرابی رخ داده باشد و تعداد گره‌های هر خوشه به ترتیب برابر $W=7$ و $W=10$ باشد مورد

نادرست الگوریتم پیشنهادی در حالتی که دو نوع خرابی رخ داده باشد و تعداد گره‌های هر خوشه به ترتیب برابر $W=7$ و $W=10$ باشد را نشان می‌دهند.

جدول (۲): DA و FAR الگوریتم پیشنهادی با P_n های مختلف و $D=1$

	$P_n=0.05$	0.1	0.15	0.2	0.25
(W=7)					
Chen	0.975/ 0.0	0.9 45/ 0.0	0.8 6/ 0.001	0.79/ 0.00	0.72/ 0.021
Lee	0.978/ 0.0	0.9 55/ 0.0	0.8 95/ 0.0	0.85/ 0.00	0.783/ 0.018
پیشنهادی	0.989/ 0.0	0.9 73/ 0.0	0.9 22/ 0.0	0.90/ 0.0	0.887/ 0.014
(W=10)					
Chen	1.0/ 0.0	0.9 63/ 0.0	0.9 55/ 0.0	0.88 3/ 0.03	0.82/ 0.014
Lee	1.0/ 0.0	0.9 8/ 0.0	0.9 78/ 0.0	0.92 2/ 0.01	0.91/ 0.012
پیشنهادی	1.0/ 0.0	1.0 /0.0	0.9 98/ 0.0	0.98 4/ 0.0	0.98/ 0.009



شکل ۷: DA و FAR الگوریتم پیشنهادی با $w=7$ و $D=2$

پیشنهادی برابر 0.979 می‌باشد. با افزایش تعداد گره‌های هر خوشه به ده عدد نتایج متفاوت تری در هر سه الگوریتم مشاهده می‌شود. به طور مشابه زمانی که احتمال خرابی حسگرها برابر $P_n=0.05$ و $W=10$ می‌باشد؛ دقت کشف خرابی در هر سه الگوریتم Chen، Lee و الگوریتم پیشنهادی برابر 1.0 خواهد بود. در حالی که با افزایش احتمال خرابی حسگرها به 0.2 ، دقت کشف خرابی در الگوریتم‌های Chen و Lee به ترتیب برابر 0.883 و 0.922 ولی در الگوریتم پیشنهادی برابر 0.984 می‌باشد. همچنین نتایج شبیه‌سازی‌ها نشان می‌دهند که اگر احتمال خرابی حسگرها برابر $P_n=0.25$ و $W=7$ باشد؛ نرخ اخطار نادرست در الگوریتم‌های Chen و Lee به ترتیب برابر 0.021 و 0.018 ولی در الگوریتم پیشنهادی برابر 0.014 می‌باشد. با افزایش تعداد گره‌های هر خوشه به ده عدد، نتایج متفاوت تری در هر سه الگوریتم مشاهده می‌شود. به طور مشابه زمانی که احتمال خرابی حسگرها برابر $P_n=0.25$ و $W=10$ می‌باشد؛ نرخ اخطار نادرست الگوریتم‌های Chen و Lee به ترتیب برابر 0.012 و 0.014 ولی در الگوریتم پیشنهادی برابر 0.009 می‌باشد.

از نتایج حاصل از شبیه‌سازی می‌توان چنین استنباط کرد که با افزایش P_n دقت کشف خرابی الگوریتم‌ها کاهش و نرخ اخطار نادرست هر سه الگوریتم افزایش می‌یابد. در هر سه الگوریتم با افزایش تعداد گره‌های همسایه هر گره و تعداد گره‌های هر خوشه دقت کشف خرابی و نرخ اخطار نادرست بهبود خواهد داشت. با این تفاوت که این بهبودها در الگوریتم پیشنهادی مشهودتر از دو الگوریتم دیگر می‌باشد.

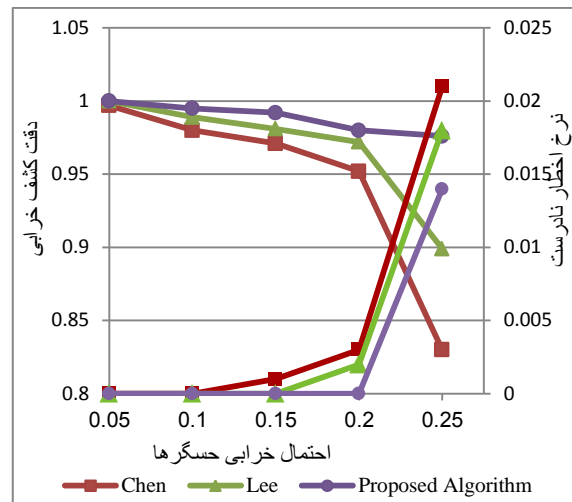
برای بررسی عملکرد الگوریتم پیشنهادی در صورت رخداد دو نوع خرابی در شبکه، شبیه‌سازی دوباره با این شرایط اجرا شده و در شرایط $W=7$ و $W=10$ نتایج حاصل مورد تجزیه و تحلیل قرار گرفته است. شکل ۷ و ۸ میزان دقت کشف خرابی و نرخ اخطار

زمان مابین خرابی‌ها را در نظر نگرفته و الگوریتم کشف خرابی را در هر دور اجرا کرده و میانگین انرژی باقیمانده گره‌های شبکه را مورد ارزیابی قرار می‌دهیم. در حالت دوم با تأثیر دادن پارامتر میانگین مابین خرابی‌ها از تعداد دفعات اجرای الگوریتم پیشنهادی کاسته شده و عملکرد متفاوتی در الگوریتم دیده می‌شود. میانگین انرژی باقیمانده گره‌ها در الگوریتم پیشنهادی پی از سپری شدن ۱۰۰۰ دوره از عملکرد شبکه در دو حالت مذکور در شکل ۱۰ آورده شده است.

جدول (۳): DA و FAR الگوریتم پیشنهادی با P_n های مختلف و $D=2$

	$P_n=0.0$ 5	0.1	0.15	0.2	0.25
(W=7)					
Chen	0.997/ 0.0	0.9 8/0. 0	0.971 /0.00 1	0.9 52/ 0.00 3	0.83/ 0.021
Lee	1.0/0.0	0.9 89/ 0.0	0.981 /0.0	0.9 72/ 0.00 1	0.899 /0.018
پیشنهادی	1.0/0.0	0.9 95/ 0.0	0.992 /0.0	0.9 8/0. 0	0.976 /0.014
(W=1)					
Chen	1.0/0.0	1.0/ 0.0	0.994 /0.00 1	0.9 85/ 0.00 3	0.977 /0.013
Lee	1.0/0.0	1.0/ 0.0	0.998 /0.0	0.9 9/0. 001	0.982 /0.01
پیشنهادی	1.0/0.0	1.0/ 0.0	1.0/0. 0	1.0/ 0.0	0.999 /0.009

همان طور که در شکل ۱۰ دیده می‌شود نمودار میانگین انرژی باقیمانده گره‌ها در الگوریتم‌های Lee و Chen در طول دوره‌های شبیه‌سازی تقریباً به طور یکنواخت کاهش پیدا می‌کنند. همان طور که انتظار داشتیم انرژی مصرفی گره‌ها در الگوریتم پیشنهادی در حالت بدون استفاده از پارامتر MTBF بیشتر از الگوریتم‌های مذکور می‌باشد. با به کارگیری پارامتر



شکل ۸: DA و FAR الگوریتم پیشنهادی با $w=10$ و $D=2$

در جدول (۳) نیز نتایج حاصل از شبیه‌سازی‌ها با تنوع خرابی برابر دو آورده شده است. با مقایسه این جدول با جدول (۲) یک افزایش چشم‌گیر در میزان دقت کشف خرابی الگوریتم پیشنهادی دیده می‌شود. به عنوان مثال با فرض $P_n=0.1$ و $W=7$ اگر تنوع خرابی برابر یک باشد؛ دقت کشف خرابی الگوریتم پیشنهادی برابر ۰٫۹۷۹ خواهد بود. در حالی که در این شرایط اگر تنوع خرابی برابر دو باشد؛ دقت کشف خرابی الگوریتم پیشنهادی برابر ۰٫۹۹۵ خواهد بود. نکته قابل توجه اینست که در صورت رخداد دو نوع خرابی، تعداد گره‌های خوشه تأثیر محسوسی در میزان نرخ اخطار نادرست الگوریتم نخواهند داشت. نکته بسیار مهمی که از مقایسه جدول (۳) با جدول (۲) حاصل می‌شود اینست که تنوع خرابی تأثیر آنچنانی بر روی نرخ اخطار نادرست این سه الگوریتم نگذاشته است.

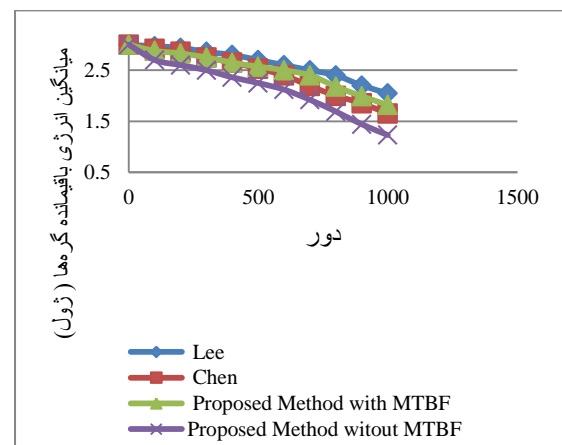
تمامی نتایج حاصل، اثبات‌کننده دقت کشف خرابی بالا و نرخ اخطار نادرست پایین الگوریتم پیشنهادی می‌باشند. مهم‌ترین عیب الگوریتم پیشنهادی مصرف انرژی نسبتاً بالای آن در مقایسه با دو الگوریتم دیگر می‌باشد. برای ارزیابی مصرف انرژی، عملکرد الگوریتم پیشنهادی را در دو حالت مختلف مورد بررسی قرار می‌دهیم. حالت اول اینکه پارامتر میانگین

پیشنهادی کمتر از دو الگوریتم مورد بررسی می‌باشد. در ضمن با افزایش تراکم گره‌ها بهبود دقت کشف خرابی مشهودتر از دو الگوریتم دیگر خواهد بود.

با بررسی نتایج شبیه‌سازی‌ها نتیجه می‌گیریم با افزایش احتمال خرابی حسگرها، نرخ اخطار نادرست افزایش می‌یابد. همچنین افزایش تعداد حسگرهای هر خوشه موجب کاهش نرخ اخطار نادرست می‌شود. با یک بررسی دقیق نتایج شبیه‌سازی‌ها دیده می‌شود؛ افزایش تعداد حسگرهای هر خوشه تاثیر مثبت بیشتری بر روی دقت کشف خرابی در مقایسه با نرخ اخطار نادرست خواهد داشت. همچنین نتیجه می‌گیریم که در صورت رخداد دو نوع خرابی، تعداد گره‌های خوشه تأثیر محسوسی در میزان نرخ اخطار نادرست الگوریتم پیشنهادی نخواهند داشت. همچنین با ارائه یک رویکرد هوشمند و تخمین زمان مابین خرابی‌ها توانستیم تعداد دفعات اجرای این الگوریتم را کاهش و طول عمر شبکه را افزایش دهیم. بنابراین نتیجه می‌گیریم که اجرای الگوریتم‌های کشف خرابی با فرکانس ثابت و در بازه زمان‌های مشخص، برای شبکه‌های حسگر بی‌سیم مناسب نخواهند بود.

به عنوان کارهای آتی پیشنهاد می‌شود که با تغییر در روند خوشه‌بندی عملکرد این الگوریتم‌ها بررسی شود. همچنین اضافه کردن تکنیک‌های هوشمندانه و رویکردهای خودتشخیصی به گره‌های حسگر پیشنهادی دیگر برای کارهای آتی می‌باشد.

MTBF و تخمین زمان مناسب برای اجرای رویکرد پیشنهادی، میزان انرژی مصرفی به مقدار قابل ملاحظه‌ای کاهش می‌یابد. در دو نمودار مربوط به الگوریتم پیشنهادی مشاهده می‌شود که در دوره‌های پایین شبکه، یک کاهش شدید در انرژی گره‌های شبکه صورت گرفته است که ناشی از خوشه‌بندی سلسله مراتبی الگوریتم پیشنهادی می‌باشد. با در نظر گرفتن رویکرد هوشمند می‌بینیم که انرژی مصرفی در دوره‌های ۴۰۰ به بعد بهتر از الگوریتم Chen خواهد بود. متأسفانه در دوره ۶۵۰ به بعد، به دلیل افزایش حسگرهای معیوب شبکه تعداد دفعات اجرای الگوریتم بیشتر شده و کاهش شدید انرژی باقیمانده گره‌ها را شاهد هستیم.



شکل ۱۰: انرژی مصرفی الگوریتم پیشنهادی در مقایسه با دو الگوریتم Lee و Chen

۶. نتیجه‌گیری

در این مقاله یک رویکرد جدید کشف خرابی سلسله مراتبی مبتنی بر خوشه‌بندی برای مدیریت خرابی شبکه‌های حسگر بی‌سیم ارائه شده است. نتایج شبیه‌سازی‌ها نشان می‌دهند که دقت کشف خرابی و نرخ اخطار نادرست الگوریتم پیشنهادی بهتر از الگوریتم‌های مورد بررسی می‌باشد. از نتایج حاصل چنین نتیجه می‌گیریم که با افزایش احتمال خرابی حسگرها، کاهش دقت کشف خرابی الگوریتم

۷. مراجع

- [1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey" *Comput. Netw.*, Vol. 38, PP. 393-422, 2002.
- [2] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless Sensor Network Survey" *Comput. Netw.*, Vol.52, PP. 2292-2330, 2008.
- [3] J. Zheng, and A. Jamalipour, *Wireless Sensor Networks: A Networking Perspective*. John Wiley & Sons, Inc., Hoboken, 2009.

- Scheme for Wireless Sensor Networks” *Wirel. Sensor Netw.* , Vol 4, PP.84-90, 2012.
- [18] A. Mahapatro, and P.M. Khilar, “Scalable Distributed Diagnosis Algorithm for Wireless Sensor Networks,” *Proc. of the Advances in Computing, Communication and Control*, Springer link, PP. 400-405, 2011.
- [19] W.C. Gangloff, “Common Mode Failure Analysis” *IEEE Trans. Power Appl. Syst.*, Vol. 94, PP. 27-30, 1974.
- [20] F. Koushanfar, M. Potkonjak, and A. Sangiovanni-Vincentelli, *Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems*. CRC Press, chapter 36, PP. 1-36, 2004.
- [21] M.-H. Lee and Y.-H. Choi, “Fault detection of wireless sensor networks” *Comput. Commun.* , Vol. 31, PP. 3469-3475, 2008.
- [22] J. Chen, S. Kher, and A. Somani, “Distributed Fault Detection of Wireless Sensor Networks,” *Proc. of the 2006 workshop on Dependability issues in wireless ad hoc networks and sensor networks (DIWANS '06)*. USA, PP. 65-72, 2006.
- [23] Zh. Chenglin, S. Xuebin, S. Songlin, and J. Ting, “Fault diagnosis of sensor by chaos particle swarm optimization algorithm and support vector machine” *Expert Syst. Appl.* , Vol. 38, PP. 9908-9912, 2011.
- [4] H. Alemdar, and C. Ersoy, “Wireless sensor networks for healthcare: A survey” *Comput. Netw.* , Vol. 54, PP. 2688-2710, 2010.
- [5] O. Boyinbode, H. Le, A. Mbogho, M. Takizawa, and R. Poliah, “A Survey on Clustering Algorithms for Wireless Sensor Networks,” *Proc. of the 13th Int. Conf. on Network-Based Information Systems (NBIS '10)*. Japan, PP. 358-364, 2010.
- [6] A.A. Abbasi, and M. Younis, “A survey on clustering algorithms for wireless sensor networks” *Comput. Commun.* , Vol. 30, PP. 2826–2841, 2007.
- [7] S. Fedor, and M. Collier, “On the problem of energy efficiency of multi-hop vs one-hop routing in wireless sensor networks,” *Proc. of the 21st Int. Conf. on Advanced Information Networking and Applications Workshops (AINAW '2007)*. Canada. PP. 380-385, 2007.
- [8] W.B. Heinzelman, A.P. Chandrakasan, and H. Balakrishnan, “Application specific protocol architecture for wireless microsensor networks” *IEEE Trans. Wirel. Commun.* , Vol. 1, PP. 660-670, 2002.
- [9] C. Li, H. Zhang, B. Hao, and J. Li, “A Survey on Routing Protocols for Large-Scale Wireless Sensor Networks” *Sensors*, Vol. 11, PP. 3498-3526, 2011.
- [10] A.D.G. Biroli, M. Martina, and G. Masera, “An LDPC Decoder Architecture for Wireless Sensor Network Applications” *Sensors*, Vol. 12, PP. 1529-1543, 2012.
- [11] J. Shell, S. Coupland, and E. Goodyer, “Fuzzy Data Fusion for Fault Detection in Wireless Sensor Networks” *Proc. of the 10th Annual Workshop on Computational Intelligence (UKCI '10)*. London, UK, PP. 1-6, 2010.
- [12] I. Koren, and C.M. Krishna, *Fault-Tolerant systems*. Morgan Kaufmann Publishers Inc. 2007.
- [13] D.-Y Kim, and J. Cho, “Active caching: a transmission method to guarantee desired communication reliability in wireless sensor networks” *IEEE Communications Letters*, Vol. 13, PP. 378-385, 2009.
- [14] M.C Vuran, and I.F. Akyildiz, “Error control in wireless sensor networks: a cross layer analysis” *Journal of IEEE/ACM Trans. on Netw.* , Vol.17, PP. 1186-1199, 2009.
- [15] M. Radi, B. Dezfouli, K.A. Bakar, and M. Lee, “Multipath Routing in Wireless Sensor Networks: Survey and Research Challenges” *Sensors*, Vol. 12, PP. 650-685, 2012.
- [16] L. Paradis, and Q. Han, “Fault Management in Wireless Sensor Networks: a survey” *Journal of Netw. and syst. Manag.* , Vol. 15, PP. 171-190, 2007.
- [17] S. Hyun Oh, Ch.O Hong, and Y.-H Choi, “A Malicious and Malfunctioning Node Detection

¹ Data Fault

² Cyclic Redundancy Check

³ Low-Density Parity-check Code

⁴ Functional Fault

⁵ Fault Avoidance Techniques

⁶ Active Caching

⁷ Multi-Path

⁸ Fault Prevention

⁹ Fault Detection

¹⁰ Fault Isolation

¹¹ Fault Diagnostic

¹² Fault Recovery

¹³ Common Mode Failure (CMF)

¹⁴ Equipment Design Deficiency

¹⁵ Functional Deficiency

¹⁶ Operation and Maintenance Errors

¹⁷ External Normal Environment

¹⁸ External Phenomena

¹⁹ Dependability

²⁰ Poisson Distribution

²¹ Fault Detection

²² Fault Diagnosis

-
- ²³ Likely Good (LG)
 - ²⁴ Likely Faulty (LF)
 - ²⁵ Good (GD)
 - ²⁶ Three layer- Hierarchical Clustering
 - ²⁷ Match
 - ²⁸ Mean Time Between Failure (MTBF)
 - ²⁹ Media Access Control (MAC)
 - ³⁰ Detection Accuracy (DA)
 - ³¹ False Alarm Rate (FAR)
 - ³² Diversity