علوم و مهندسی کامپیوتر
نشریه علمی پژوهشی انجمن کامپیوتر ایران
مجلد ۱۱، شماره ۳ (الف)، ۱۳۹۲
صفحات ۳۱-۳۹
مقاله عادی

بکارگیری اتوماتاهای یادگیر در خنثی کردن حمله ارسال انتخابی در شبکه‌های حسگر بی‌سیم

مجتبی جمشیدی[۱] مهدی اثنی عشری[۲] محمدرضا میبدی[۳]

[۱]آموزشکده فنی و حرفه‌ای سما، دانشگاه آزاد اسلامی واحد کرمانشاه، کرمانشاه، ایران
[۲]پژوهشکده فناوری اطلاعات، پژوهشگاه فضای مجازی، تهران، ایران
[۳]دانشکده کامپیوتر و فناوری اطلاعات، دانشگاه صنعتی امیرکبیر، تهران، ایران

چکیده

حمله ارسال انتخابی می‌تواند بسیاری از کاربردهای بحرانی نظارتی و ماموریتی نظیر دیده‌بانی نظامی و نظارت بر آتش‌سوزی جنگل‌ها را مختل کند. در این حمله، گره‌های بدخواه در اکثر اوقات مشابه گره‌های قانونی رفتار می‌کنند اما گاهی اوقات به‌طور انتخابی بسته‌های گزارشی حساس (نظیر بسته گزارشی مربوط به حرکت وسایل جنگی دشمن) را ساقط می‌کنند و از این‌رو شناسایی این نوع حمله دشوار است. در این مقاله یک الگوریتم کاملاً توزیعی، پویا، سبک وزن و هوشمند مبتنی بر اتوماتاهای یادگیر جهت مقابله با حمله ارسال انتخابی در شبکه‌های حسگر بی‌سیم پیشنهاد می‌شود. در این الگوریتم از مکانیزم شنود به همراه مدل اتوماتای یادگیر جهت انتخاب مسیرهای ایمن ارسال بسته‌ها در پروتکل‌های مسیریابی چندگامه استفاده می‌شود. هر گره مجهز به یک اتوماتای یادگیر است که وظیفه آن انتخاب گام- بعدی (گره بالادستی) برای ارسال داده‌ها به سمت ایستگاه پایه و نظارت بر عملکرد آن است. شبیه‌سازی الگوریتم پیشنهادی توسط شبیه‌ساز J-SIM صورت گرفته و نتایج شبیه‌سازی‌ها، در قالب معیارهای نرخ تحویل بسته‌ها، نرخ ساقط شدن بسته‌ها توسط گره‌های بدخواه و متوسط انرژی باقی‌مانده گره‌ها، حاکی از برتر بودن روش پیشنهادی نسبت به الگوریتم پایه، الگوریتم مبتنی بر تصدیق چندگامه، الگوریتم مبتنی بر چند جریان داده‌ای، الگوریتم چند مسیره و الگوریتم مبتنی بر گره‌های نگهبانناست.

کلمات کلیدی: شبکه‌های حسگر بی‌سیم، امنیت، حمله ارسال انتخابی، اتوماتای یادگیر.

۱- مقدمه

شبکه‌های حسگر بی‌سیم نوع خاصی از شبکه‌های موردی هستند که شامل صدها تا هزاران گره کوچک و ارزان قیمت می‌باشند. گره‌های حسگر توانایی حس محیط اطراف با هدف معین، پردازش اطلاعات، ذخیره‌سازی، تبادل اطاعات با سایر گره‌ها و هم‌چنین قابلیت وفق‌پذیری در مقابل تغییرات (توپولوژی و ...) را دارند. با توجه به کاربرد وسیع شبکه‌های حسگر بی‌سیم در فیلدهای نظامی (نظیر نظارت مرزها، تشخیص حضور یا حرکت وسایل جنگی یا نیروهای دشمن و ...)، هم‌چنین با توجه به ماهیت انتقال بی‌سیم و چندگامه داده‌ها، محدودیت‌ها (انرژی، ارتباطات، حافظه، قدرت محاسباتی)، عدم مراقبت از گره‌ها در محیط و ... برقراری امنیت در این شبکه‌ها امری بسیار مهم و حیاتی است [۱].

یکی از حمله‌های خطرناک لایه شبکه در شبکه‌های حسگر بی‌سیم، حمله ارسال انتخابی[۱] است که اولین بار در [۲] مطرح شد. در این حمله، گره بدخواه اقدام به ساقط[۲] کردن برخی از بسته‌های دریافتی می‌کند. اگر گره بدخواه تمام بسته‌های دریافتی را ساقط کند، تشخیص آن راحت است، ولی اگر گره بدخواه اقدام به ارسال انتخابی کند، یعنی فقط برخی از بسته‌های دریافتی را ساقط کند، تشخیص آن مشکل و چالش‌زا خواهد بود [۳، ۴].

یک رویکرد ممکن جهت کاهش اثرات حمله ارسال انتخابی، استفاده از پروتکل‌های مبتنی بر تصدیق چندگامه[۳] [۵، ۶] است. در این روش، اگر یک گره میانی از گره‌های بالا-دست یا پایین-دست خود بدرفتاری ببیند، یک پیغام هشدار تولید و آن را به گره منبع[۴] یا ایستگاه پایه[۵] تحویل می‌دهد. سپس گره منبع و ایستگاه پایه به کمک یک سیستم تشخیص نفوذ[۶] پیچیده تصمیم‌گیری کرده و عکس‌العمل مناسب را از خود بروز می‌دهند. در این روش از پروتکل‌های مسیریابی و انتقال خاص، نظیر Directed Diffusion [۷] و PSFQ [۸] استفاده می‌شود. به‌طور کلی، این نوع الگوریتم‌ها معایبی نظیر تأخیر و سربار زیاد محاسباتی، عدم کارایی (گره‌های حسگر باید تلاش زیادی جهت شناسایی حمله ارسال انتخابی انجام دهند)، مشکل امنیتی، عدم مقیاس‌پذیری، عکس‌العمل کند، مصرف انرژی بالا و متکی بودن به پروتکل‌های مسیریابی و انتقال خاص دارند [۹، ۱۰].

در این مقاله، یک الگوریتم مقاوم، پویا، کاملاً توزیعی، سبک وزن و هوشمند ارائه می‌شود که می‌تواند گره‌های بدخواه در حمله ارسال انتخابی را از مسیرهای داده‌ای به‌طور موثری کنار زند. روش پیشنهادی از مکانیزم شنود به همراه مدل اتوماتای یادگیر جهت انتخاب مسیرهای ایمن ارسال بسته‌ها در پروتکل‌های مسیریابی چندگامه استفاده می‌کند. از مزایای روش پیشنهادی می‌توان به عکس‌العمل سریع، نرخ بالای تحویل بسته‌ها به ایستگاه پایه، پایین بودن سربار ارتباطات و انرژی مصرفی، تطبیق پذیری با تغییر توپولوژی و ... اشاره کرد.

ادامه این مقاله به صورت زیر سازماندهی شده است. در بخش ۲ کارهای گذشته آمده است. اتوماتاهای یادگیر و پروتکل برپایی کلید LEAP به ترتیب در بخش‌های ۳ و ۴ شرح داده می‌شوند. مدل سیستم و حمله در بخش ۵ آمده است. در بخش ۶ الگوریتم پیشنهادی شرح داده می‌شود. ارزیابی کارایی و نتایج شبیه‌سازی در بخش ۷ و نتیجه‌گیری در بخش ۸ آمده است.

۲- کارهای گذشته

همان‌طور که گفته شد حمله ارسال انتخابی برای اولین بار در [۲] مطرح شد و اولین راه‌کار جهت مقابله با این حمله، استفاده از پروتکل‌های مسیریابی چندمسیره بیان شد. در این روش، بسته‌ها از طریق n مسیر کاملاً مجزا از مبدا به سمت مقصد مسیردهی می‌شوند و از این‌رو تا زمانی که حداکثر n گره ضبط نشود، کاملاً در مقابل حمله ارسال انتخابی مقاوم می‌باشد. معایب این راه‌کار: پایین بودن امنیت، عدم تشخیص گره بدخواه و افزایش انرژی مصرفی و سربار ارتباطات است. در [۵] یک پروتکل دیگر ارائه شده است که در آن با توجه به پاسخ‌های دریافتی از گره‌های میانی، از یک الگوریتم تصدیق چندگامه استفاده می‌کند تا پیغام‌های هشدار را در سطح شبکه منتشر کند. در [۶] یک تکنیک دیگر برای شناسایی گره‌های بدخواه در حمله ارسال انتخابی ارائه شده که در حقیقت بهبود یافته تکنیک [۵] است. در [۱۱] یک الگوریتم تشخیص نفوذ متمرکز، مبتنی بر ماشین‌های برداری تقویتی[۷] و تکنیک پنجره لغزان[۸] برای مقابله با حمله چاله سیاه[۹] و ارسال انتخابی مطرح شده است. معایب این راه‌کار: عدم تشخیص گره بدخواه و انتخاب مسیر جایگزین، مشکل مقیاس‌پذیری و امنیتی (به دلیل متمرکز بودن). در [۱۲] یک پروتکل دیگر جهت مقابله با حمله ارسال انتخابی ارائه شده که مبتنی بر توپولوژی‌های چند جریان داده‌ای (MDT[۱۰]) می‌باشد. ایده اصلی MDT، تقسیم گره‌های حسگر به چند گروه یا جریان داده‌ای کاملاً مجزا می‌باشد. به طوری‌که، داده‌های حس شده توسط گره‌های منبع از طریق این جریان‌های داده‌ای مجزا به سمت ایستگاه پایه فرستاده شوند. معایب این راه‌کار: پایین بودن امنیت، عدم شناسایی گره بدخواه، بالا بودن هزینه شبکه و ارتباطات و پایین بودن طول عمر شبکه. در [۱۳] یک الگوریتم تحویل داده قابل اطمینان مبتنی بر منطق فازی ارائه شده است که در واقع شکل بهبود یافته تکنیک مسیریابی چند مسیره [۲] است. در این رویکرد فرض شده است که ایستگاه پایه از سطح انرژی شبکه آگاه است یا می‌تواند آن را تخمین بزند، ایستگاه پایه تعداد گره‌های بدخواه در آینده را می‌داند و هر گره از موقعیت مکانی خود در محیط آگاه است، هم‌چنین از الگوریتم تصدیق چندگامه [۵] برای تشخیص حمله ارسال انتخابی استفاده می‌کند. معایب این راه‌کار: دارا بودن همان محدودیت‌های روش مسیریابی چند مسیره، نیاز به تعیین تعداد گره‌های بدخواه در آینده و نیاز به سخت افزار اضافی مثل GPS برای تعیین مکان گره‌ها. در [۱۴] یک الگوریتم سبک وزن دیگر ارائه شده که در آن فقط از اطلاعات همسایه‌ها جهت شناسایی حمله ارسال انتخابی استفاده می‌شود. در [۱۵] یک الگوریتم دیگر برای مقابله با حمله ارسال انتخابی ارائه شده که از مدل شبکه حسگر ناهمگن جهت شناسایی حمله ارسال انتخابی استفاده می‌شود. این الگوریتم فقط در شبکه‌های حسگر مبتنی بر کلاستر عمل می‌کند. در [۱۶] یک روش دیگر برای مقابله با حمله ارسالی انتخابی مطرح شده است که در آن از یک رویکرد مبتنی بر چندجمله‌ای استفاده می‌شود. ایده اصلی این است که داده‌های حس شده به چند قطعه تقسیم می‌شوند و این قطعات از طریق مسیرهای مجزا به سمت ایستگاه پایه فرستاده می‌شوند. اگر ایستگاه پایه به تعداد کافی از قطعات بسته را دریافت کند، آن را به عنوان داده اصلی تجزیه می‌کند، در غیر این‌صورت ایستگاه پایه تشخیص می‌دهد که داده توسط گره بدخواه تحریف[۱۱] شده است. معایب این الگوریتم: بالا بودن سربار ذخیره‌سازی و محاسبات (به دلیل تقسیم کردن و پردازش داده اصلی به چند قطعه کوچکتر) و سربار ارتباطات زیاد به خاطر ارسال مقادیر چند جمله‌ای به ایستگاه پایه. در [۱۷] نیز یک الگوریتم مبتنی بر نظارت ترافیک جهت شناسایی حمله ارسال انتخابی مطرح شده است.

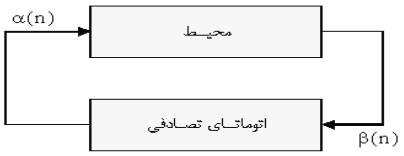
در این رویکرد از گره‌های EM[۱۲] جهت شنود و نظارت همه ترافیک شبکه استفاده می‌شود. معایب این راه‌کار: تأثیر تغییر توپولوژی بر کارایی الگوریتم و رنج بردن از مسئله شکست تک گره (با شکست یا ضبط ایستگاه پایه یا گره‌های EM). در [۱۸] نیز یک الگوریتم مبتنی بر آزمون مش ترتیبی[۱۳] جهت تشخیص حمله ارسال انتخابی در شبکه‌های حسگر مطرح شده است. ماهیت این الگوریتم متمرکز بوده و برای شبکه‌های مبتنی بر کلاستر کار می‌کند. در [۱۹] یکالگوریتم تدافعی سبک وزن ارائه شده است که از گره‌های همسایه به عنوان گره‌های مانیتور کننده (نظارتی) استفاده می‌کند. گره‌های نظارتی همسایه در واقع انتقال بسته‌های ساقط شده را مانیتور می‌کنند و بسته‌های ساقط شده را دوباره ارسال می‌کنند. در این الگوریتم از یکی توپولوژی توری شش ضلعی استفاده می‌شود. معایب این راه‌کار: تأثیر تغییر توپولوژی بر کارایی الگوریتم، اگر گره نظارتی توسط دشمن ضبط شود هیچ اقدام متقابلی ارائه نشده است، پر هزینه بودن الگوریتم به دلیل نیاز گره‌ها به GPS برای تعیین مکان خود، استفاده از یک مسیریابی مبتنی بر احتمال که بهینه نیست. در [۲۰] نیز یک الگوریتم دیگر جهت مقابله با حمله ارسال انتخابی ارائه شده که مبتنی بر مدل تئوری بازی‌ها[۱۴] می‌باشد. در [۲۱] نیز یک الگوریتم مقاوم با استفاده از سیستم دیدبان همسایه[۱۵] (NWS) در مقابل ساقط شدن بسته‌ها توسط گره‌های بدخواه ارائه شده است.

۳- اتوماتای یادگیر

یک اتوماتای یادگیر [۲۲، ۲۳] یک ماشین با حالات محدود است که می‌تواند تعداد محدودی عمل را انجام دهد. هر عمل انتخاب شده، توسط یک محیط تصادفی ارزیابی شده و پاسخی به اتوماتای یادگیر داده می‌شود. اتوماتای یادگیر از این پاسخ استفاده نموده و عمل خود را برای مرحله بعد انتخاب می‌کند. در طی این فرآیند، اتوماتای یادگیر یاد می‌گیرد که چگونه بهترین عمل را از بین اعمال

مجاز خود انتخاب کند. شکل ۱ ارتباط بین اتوماتای یادگیر و محیط را نشان می‌دهد.

محیط را می‌توان توسط سه‌تایی $E \equiv \{\alpha, \beta, c\}$ نشان داد که در آن $\alpha \equiv \{\alpha_1, \alpha_2, ..., \alpha_r\}$ مجموعه ورودی‌های محیط، $\beta \equiv \{\beta_1, \beta_2, ..., \beta_m\}$ مجموعه خروجی‌های محیط و $c \equiv \{c_1, c_2, ..., c_r\}$ مجموعه احتمال‌های جریمه می‌باشند. ورودی محیط یکی از r عمل انتخاب شده اتوماتا است. خروجی (پاسخ) محیط به هر عمل i توسط β_i مشخص می‌شود. اگر β_i یک پاسخ دودویی باشد، محیط مدل P نامیده می‌شود. در چنین محیطی $\beta_i(n) = 1$ به‌عنوان پاسخ نامطلوب یا شکست و $\beta_i(n) = 0$ به‌عنوان پاسخ مطلوب یا موفقیت در نظر گرفته می‌شوند. به این ترتیب اتوماتای یادگیر تصادفی را می‌توان با چهارتایی $LA \equiv \{\alpha, \beta, p, T\}$ نشان داد که $\alpha \equiv \{\alpha_1, \alpha_2, ..., \alpha_r\}$ مجموعه عمل‌های اتوماتا (r تعداد عمل‌های اتوماتا)، $\beta \equiv \{\beta_1, \beta_2, ..., \beta_r\}$ مجموعه ورودی‌های اتوماتا، $p \equiv \{p_1, p_2, ..., p_r\}$ بردار احتمال عمل‌های اتوماتا و $T \equiv p(n+1) = T[\alpha(n), \beta(n), p(n)]$ الگوریتم یادگیری می‌باشد.



شکل ۱- اتوماتای یادگیر تصادفی [۲۲]

اتوماتای یادگیر با عمل‌های متغیر: در بعضی از کاربردها نیاز به اتوماتایی با تعداد عمل متغیر می‌باشد [۲۴]. این اتوماتا در لحظه n عمل خود را فقط از یک زیر مجموعه غیر تهی ($V(n)$) از عمل‌ها که عمل‌های فعال نامیده می‌شوند، انتخاب می‌کند. انتخاب مجموعه $V(n)$ توسط یک عامل خارجی و بصورت تصادفی انجام می‌شود. نحوه فعالیت این اتوماتا بصورت زیر است. برای انتخاب یک عمل در زمان n، ابتدا مجموع احتمال عمل‌های فعال خود ($K(n)$) را محاسبه می‌کند و سپس بردار $p(n)$ را مطابق رابطه ۱ محاسبه می‌کند. آن‌گاه اتوماتا یک عمل از مجموعه عمل‌های فعال خود را بصورت تصادفی و مطابق بردار احتمال $p(n)$ انتخاب کرده و بر محیط اعمال می‌کند. اگر عمل انتخاب شده α_i باشد، پس از دریافت پاسخ محیط، اتوماتا بردار احتمال $p(n)$ عمل‌های خود در صورت دریافت پاداش (a میزان پاداش) بر اساس رابطه ۲ و در صورت دریافت جریمه (b میزان جریمه) طبق رابطه ۳ به‌روز می‌کند (در محیط مدلP) [۲۴].

$$p_i(n) = prob[\alpha(n) = \alpha_i \mid V(n) \text{ is set of active actions}, \alpha_i \in V(n)] = \frac{p_i(n)}{K(n)} \quad (۱)$$

الف- پاسخ مطلوب از محیط

$$\begin{aligned} p_i(n+1) &= p_i(n) + a.(1 - p_i(n)) \quad \alpha(n) = \alpha_i \\ p_i(n+1) &= p_j(n) + a.p_i(n) \quad \alpha(n) = \alpha_i,\ \forall j\ \ j \neq i \end{aligned} \quad (۲)$$

ب- پاسخ نامطلوب از محیط

$$\begin{aligned} p_i(n+1) &= (1-b).p_i(n) \quad \alpha(n) = \alpha \\ p_i(n+1) &= \frac{b}{r-1} + (1-b)p_j(n) \quad \alpha(n) = \alpha_i,\ \forall j\ \ j \neq i \end{aligned} \quad (۳)$$

سپس اتوماتا بردار احتمال عمل‌ها $p(n)$ را با استفاده از بردار $p(n+1)$ و بصورت زیر به‌روز می‌کند:

$$\begin{aligned} p_j(n+1) &= p_j(n+1).K(n) \quad \text{for all j}, \alpha_j \in V(n) \\ p_j(n+1) &= p_j(n) \quad \text{for all j}, \alpha_j \notin V(n) \end{aligned} \quad (۴)$$

۴- پروتکل برپایی کلید در LEAP

پروتکل LEAP [۲۵]، ازبرپایی چهار نوع کلید زیر برای هر گره حسگر پشتیبانی می‌کند:

کلید منحصر به فرد: این کلید بین گره حسگر و ایستگاه پایه به اشتراک گذاشته می‌شود و زمانی‌که ایستگاه پایه نیاز به تصدیق اطلاعات دریافتی دارد، گره حسگر از این کلید جهت محاسبه کدهای تصدیق پیغام (MACs) اطلاعات حس شده استفاده می‌کند.

کلید جفتی: این کلید بین یک گره و هر یک از همسایه‌هایش به اشتراک گذاشته می‌شود و جهت مخابره‌های امنیتی که نیاز به محرمانگی و تصدیق هویت منبع دارند استفاده می‌شود. برای مثال، یگ گره می‌تواند از کلید جفتی خود جهت توزیع کلید کلاستر به همسایه‌هایش، و یا جهت انتقال ایمن داده‌های حس شده به گره تجمیع‌گر استفاده کند.

کلید کلاستر: یک کلید کلاستر توسط یک گره و همه همسایه‌هایش به اشتراک گذاشته می‌شود و در اصل برای ایمنی پیغام‌های انتشاری محلی، نظیر اطلاعات کنترلی مسیریابی استفاده می‌شود. از آن‌جا که، تکنیک‌های پردازشی درون شبکه‌ای، از جمله تجمع داده‌ها و مشارکت غیرفعال برای ذخیره انرژی در شبکه‌های حسگر امری مهم است، لذا یک گره می‌تواند انتقالات گره حسگر همسایه خود را شنود کند و از انتقالات مشابه جلوگیری کند. در پروتکل LEAP، هر گره u یک کلید کلاستر یکتا با همه همسایه‌هایش به اشتراک می‌گذارد. همسایه‌هایش نیز از همان کلید جهت رمزگشایی و تصدیق پیغام‌های گره u استفاده می‌کنند.

کلید گروهی: این کلید بین همه گره‌های شبکه به اشتراک گذاشته می‌شود و توسط ایستگاه پایه جهت رمزگذاری پیغام‌هایی که نیاز است به تمام شبکه منتشر شود استفاده می‌شود.

۵- مدل شبکه و حمله

در این بخش، فرضیات، مدل شبکه و مدل حمله و برای الگوریتم پیشنهادی ارائه می‌شود.

۵-۱- مدل شبکه و فرضیات

ما به شبکه حسگر به عنوان یک گراف بدون وزن غیرجهت‌دار $G = (V, E)$ نگاه می‌کنیم که در آن، $V = \{v_1, v_2, \dots, v_\psi\}$ مجموعه گره‌های حسگر و $E = \{e_{ij}, \dots\}$ مجموعه لینک‌های ارتباطی بین گره‌ها در شبکه است. به طوری که، $|V| = \psi$ تعداد عضوهای V و $|E| = \xi$ تعداد عضوهای E می‌باشند. $e_{ij} \in E = v_i \rightarrow v_j$ اگر $v_i \in V$ در مجاورت $v_j \in V$ قرار گرفته باشد. گره‌های حسگر به دو دسته‌ی گره‌های منبع (SN) و گره‌های میانی (FN) تقسیم می‌شوند. هر گره یک شناسه یکتا دارد. محیط عملیاتی مورد نظر، یک محیط دوبعدی

می‌باشد که گره‌های منبع به‌طور مشخص و گره‌های میانی به‌طور تصادفی در این محیط پراکنده می‌شوند. تمام گره‌ها، همگن و دارای برد رادیویی ثابت (برابر r) هستند. هم‌چنین، فرض می‌شود که گره‌ها با یکدیگر از طریق کانال رادیویی بی‌سیم مخابره و از انتشار به شیوه همه‌جهته استفاده می‌کنند. لینک‌های ارتباطی دوطرفه هستند، یعنی اگر گره u بتواند یک پیغام از گره v دریافت کند، هم‌چنین می‌تواند یک پیغام به v بفرستد. گره‌های منبع داده‌های حس شده خود را با کمک گره‌های میانی و از طریق یک مسیر چندگامه به گره ایستگاه پایه در شبکه تحویل می‌دهند. گره‌های حسگر و لینک‌های ارتباطی ایمن نیستند، یعنی دشمن می‌تواند گره‌ها را ضبط و تحت عنوان گره‌های بدخواه، برنامه‌ریزی مجدد[۱۶] کند. هم‌چنین، فرض می‌شود مدل شبکه ما از پروتکل برپایی کلید LEAP [۲۵] جهت برپایی کلیدها بین گره‌ها در شبکه بهره می‌گیرد.

۵-۲- مدل حمله

مدل حمله در نظر گرفته شده در این مقاله، برگرفته از مدل حمله در [۲، ۱۲] می‌باشد. هنگامی‌که شبکه‌های حسگر در کاربردهای نظامی بکار گرفته شوند، تشخیص به موقع روخدادها در محیط هدف و ارسال سریع گزارش‌ها به ایستگاه پایه امری مهم می‌باشد. ولی این عمل می‌تواند به آسانی توسط حمله‌های ارسال انتخابی مخدوش شود. در این حمله، گره بدخواه می‌تواند از ارسال بسته‌های دریافتی به سمت ایستگاه پایه امتناع ورزد و به آسانی بسته دریافتی را ساقط کند. در این‌جا فرض می‌شود که دشمن می‌تواند گره‌های نرمال در شبکه را ضبط و به عنوان گره‌های بدخواه در شبکه وارد کند و یا گره‌های بدخواه خارجی را به داخل شبکه تزریق کند. هم‌چنین، فرض می‌شود که گره‌های بدخواه با همدیگر همکاری ندارند و هیچ محدودیتی در نرخ ساقط کردن بسته‌ها ندارند.

۶- الگوریتم پیشنهادی

الگوریتم پیشنهادی می‌تواند بر روی هر یک از الگوریتم‌های مسیریابی چندگامه اجرا گردد. در این‌جا، صرفاً به عنوان نمونه، از یک درخت مسیریابی چندگامه مبتنی بر کوتاه‌ترین مسیر استفاده شده است که ایستگاه پایه به عنوان ریشه این درخت در نظر گرفته می‌شود. الگوریتم پیشنهادی از ۳ فاز تشکیل شده است. فاز اول، ساخت درخت مسیریابی و تعیین سطح گره‌ها در درخت مسیریابی، فاز دوم، پیکربندی اتوماتاهای یادگیر و فاز سوم، ارسال داده‌ها به همراه انتخاب مسیرهای ایمن است.

۶-۱- فاز اول: تعیین سطح گره‌ها

در الگوریتم پیشنهادی، از یک بسته تولید مسیر که قالب آن در شکل (۲-الف) آمده است، جهت ساخت درخت مسیریابی و تعیین مسیرهای ممکن از گره‌های حسگر به ایستگاه پایه استفاده می‌شود. بسته تولید مسیر دارای دو فیلد NodeID و HopCount می‌باشد که به ترتیب دربرگیرنده شناسه گره و حداقل فاصله گره تا ایستگاه پایه برحسب گام می‌باشند. هم‌چنین، هر گره حسگر در حافظه خود یک جدول مسیریابی دو ستونه مطابق شکل (۲-ب) دارد که در ستون NodeID شناسه هر گره همسایه خود و در ستون NodeLevel شماره سطح آن گره همسایه در درخت مسیریابی را ذخیره می‌کند. به‌علاوه، هر گره حسگر در حافظه خود نیز یک فیلد دیگر به نام MinHopCount دارد که بیان‌گر حداقل فاصله گره تا ایستگاه پایه برحسب گام، یا به عبارت دیگر مشخص کننده سطح گره در درخت مسیریابی است. هر گره حسگر پس از دریافت بسته‌های تولید مسیر از همسایه‌های خود، مقدار فیلد MinHopCount را از رابطه زیر (رابطه ۵) محاسبه می‌کند.

$$\text{MinHopCount}=\min(\text{NodeLevel})+1 \qquad (۵)$$

پس از گسترش گره‌ها در محیط، ایستگاه پایه به عنوان ریشه درخت مسیریابی (تنها گره سطح صفر که شناسه آن هم ۰ می‌باشد)، یک بسته تولید مسیر با مقدار <NodeID=0, HopCount=0> تولید و آن را در همسایگی خود منتشر می‌کند. در حالت ایده‌آل و بدون در نظر گرفتن نویزهای احتمالی، تمام گره‌های حسگر v_i که در برد رادیویی ایستگاه پایه قرار گرفته باشند، به عنوان گره‌های سطح ۱ درخت مسیریابی، این بسته تولید مسیر را دریافت می‌کنند. هر یک از گره‌های سطح ۱ درخت مسیریابی، با دریافت این بسته، در جدول مسیریابی خود یک سطر اضافه می‌کنند به طوری‌که در فیلد NodeID شناسه گره ایستگاه پایه (یعنی عدد ۰) و در فیلد NodeLevel عدد ۰ را قرار می‌دهند.

سپس هر یک از این گره‌های همسایه ایستگاه پایه، یعنی v_iها، یک بسته تولید مسیر جدید ایجاد می‌کنند، به طوری که فیلد NodeID آن با شناسه‌ی خود گره v_i و فیلد HopCount آن با مقدار MinHopCount (که در این‌جا، طبق فرمول (۵)، برای همه v_iها برابر ۱ است) مقداردهی می‌شود، و سپس این بسته تولید مسیر خود را منتشر می‌کنند تا به دست همسایه‌های سطح بعدی برسد. این روند ادامه می‌یابد تا بسته تولید مسیر به همه گره‌های دسترس‌پذیر در شبکه برسد.

(الف) بسته تولید مسیر	
NodeID	HopCount

(ب) جدول مسیریابی	
NodeID	NodeLevel

شکل ۲- (الف) قالب بسته‌های تولید مسیر و (ب) جدول مسیریابی گره‌ها

۶-۲- فاز دوم: پیکربندی اتوماتای یادگیر

در الگوریتم پیشنهادی، هر گره حسگر مجهز به یک اتوماتای یادگیر با عمل متغیر است. در این فاز، هر گره در جدول مسیریابی خود، گره‌های همسایه‌ای که مقدار NodeLevel آن‌ها از MinHopCount خودش کوچکتر باشد (یعنی همسایه‌های سطح بالاتر) را در لیستی به نام SelectionSet قرار می‌دهد. این لیست در واقع همان لیست مجموعه عمل‌های ممکن برای اتوماتای یادگیر است. در آغاز، همه عمل‌های اتوماتا فعال (یعنی، V(k)=SelectionSet) و دارای احتمال یکسانی می‌باشند که از رابطه زیر محاسبه می‌شود:

$$P_i=\frac{1}{r} \qquad \text{r تعداد اعضای SelectionSet می‌باشد،} \qquad (۶)$$

سپس، اتوماتای یادگیر هر گره حسگر u، به‌طور تصادفی یکی از عمل‌های خود را انتخاب می‌کند. عمل انتخاب شده در واقع همان گره همسایه‌ای است که جهت ارسال بسته‌ها به سمت ایستگاه پایه انتخاب می‌شود و در فاز بعدی گره u بسته‌های خود و یا بسته‌های دریافتی را از طریق این گره همسایه به سمت ایستگاه پایه جلو می‌برد. در ادامه به گره همسایه انتخاب شده توسط اتوماتای یادگیر، "گام- بعدی" می‌گوییم.

۶-۳- فاز سوم: ارسال داده‌ها و انتخاب مسیرهای ایمن

پس از پایان فاز دوم که خیلی کوتاه می‌باشد، الگوریتم وارد فاز سوم می‌شود و گره‌ها اقدام به ارسال داده‌های خود می‌کنند. هر گره منبع پس از تولید گزارش، آن را به گام- بعدی خود ارسال می‌کند تا بسته به صورت چندگامه به دست ایستگاه پایه برسد. گره‌های حسگر بسته‌های گزارشی را با کلید کلاستری خود رمزگزاری می‌کنند تا عمل شنود امکان‌پذیر شود.

در این فاز، علاوه بر ارسال داده‌ها، اتوماتاهای یادگیر باید به طور توزیعی و پویا مسیرهای ایمن به سمت ایستگاه پایه را انتخاب کنند. برای این‌کار، گره u هنگام ارسال بسته‌های خود به گام بعدی، مثلاً گره v، بسته‌های با دوره تناوب $T_{OverHear}$ (دوره تناوب شنود) را در بافر خود ذخیره می‌کند و کانال را شنود می‌کند تا مشخص گردد گره v آن بسته را ارسال می‌کند یا خیر. گره u پس از گذشت زمان T_{wait}، با توجه به نتیجه شنود (موفق یا ناموفق) به دو صورت عمل می‌کند.

در صورت شنود موفق، اتوماتای یادگیر به عمل متناظر با همین گره v طبق فرمول (۲) پاداش می‌دهد و گره v هم‌چنان به عنوان گام بعدی گره u در ارسال‌های بعدی مورد استفاده قرار خواهد گرفت. در غیر این‌صورت، اتوماتای یادگیر عمل متناظر با گره v را طبق فرمول (۳) جریمه نموده، گره v را موقتاً از لیست عمل‌های فعال اتوماتا خارج می‌کند (یعنی، $V(k) = SelectionSet - \{v\}$) و سپس از بین عمل‌های فعال خود گره دیگری، مثلاً w، را مطابق فرمول (۱) به عنوان گام-بعدی انتخاب نموده، بسته ساقط شده را مجدداً از طریق گام-بعدی جدید (یعنی گره w) ارسال می‌کند و سپس گره v را مجدداً به SelectionSet خود اضافه می‌کند.

به این ترتیب اتوماتاهای یادگیر با کمترین سربار ممکن، مانع از انتخاب گره‌های بدخواه به عنوان گام بعدی گره‌های نرمال می‌شوند. نکته خیلی مهم در این الگوریتم این است که هیچ گره‌ای به عنوان گره بدخواه علامت زده نمی‌شود و هیچ پیغامی مبنی بر شناسایی گره بدخواه در شبکه منتشر نمی‌شود، بلکه فقط اتوماتاهای یادگیر احتمال آن‌ها را در بردار احتمالات عمل‌های اتوماتا کاهش می‌دهند تا به ندرت یا هیچ وقت به عنوان گام- بعدی در درخت مسیریابی انتخاب نشوند. این ویژگی، خصوصاً در حالتی‌که دشمن با یک گره خارجی موبایل و با تولید پارازیت، حمله ارسال انتخابی را شبیه‌سازی می‌کند بسیار مناسب است.

به عنوان مثال، این سناریو را در نظر بگیرید که مهاجم خارجی موبایل تا زمان t_1 در مجاورت گره قانونی u قرار گرفته باشد و با تولید پارازیت بسته‌های ارسالی آن را ساقط کند، در این زمان گره‌های همسایه‌ای که گره u را به عنوان گام بعدی خود انتخاب کرده باشند، گره u را بدخواه تلقی می‌کنند و به آن بسته ارسال نمی‌کنند یا به ندرت ارسال می‌کنند. سپس، اگر مهاجم خارجی موبایل بعد از زمان t_1 به مکان دیگری از شبکه حرکت کند، با توجه به ماهیت انتخاب گام- بعدی توسط اتوماتاهای یادگیر، گره u این شانس را خواهد داشت که مجدداً توسط همسایه‌هایش به عنوان گام- بعدی قابل اعتماد انتخاب شود. درحالی که الگوریتم‌های دیگر، نظیر الگوریتم ارائه شده در [۵]، ممکن است گره u را برای همیشه به عنوان گره بدخواه علامت زنند و حتی بعد از زمان t_1 نیز هیچ‌گاه به عنوان گام- بعدی توسط دیگر گره‌ها انتخاب نشود.

به این ترتیب اتوماتاهای یادگیر با گذشت زمان و به‌صورت کاملاً توزیعی و پویا مسیرهای ایمن جهت ارسال بسته‌ها به سمت ایستگاه پایه را یاد می‌گیرند و گره‌های بدخواه را از مسیرهای جریان داده‌ای کنار می‌زنند.

۷- ارزیابی کارایی و نتایج شبیه‌سازی‌ها

در این بخش، ابتدا سربار ارتباطات و ذخیره‌سازی الگوریتم پیشنهادی را ارزیابی می‌کنیم. سپس به ارائه نتایج شبیه‌سازی الگوریتم خود و مقایسه آن با دیگر الگوریتم‌ها می‌پردازیم.

۷-۱- سربار ارتباطات

برخلاف الگوریتم‌های [۲، ۵، ۱۲] که از تعداد زیادی گره جهت تحویل یک بسته منفرد استفاده می‌کنند، الگوریتم [۲۱] از تعداد کمتری گره نگهبان جهت هدایت یک بسته استفاده می‌کند، ولی در نقاطی که رفتار بدخواهانه مشاهده شود از انتقال چند مسیره استفاده می‌کند که این باعث بالا رفتن سربار ارتباط و در نتیجه مصرف بیشتر انرژی می‌شود. اما الگوریتم ما نیاز دارد هر گره فقط گام بعدی خود را جهت تحویل بسته‌ها نظارت کند و با شنود کردن، مطمئن شود که گام بعدی بسته‌های آن را ارسال می‌کند. این الگوریتم نیاز به گره‌های نگهبان با قابلیت‌های خاص (که باید ترافیک همه همسایه‌های خود را نظارت کنند) ندارد بلکه هر گره به طور مستقل مسئول نظارت بر گام بعدی خود در درخت مسیریابی است. از این‌رو، میانگین هزینه ارتباطات الگوریتم پیشنهادی ما برای تحویل بسته‌ها کمتر از الگوریتم‌های [۲، ۵، ۶، ۱۲، ۲۱] می‌باشد.

۷-۲- سربار ذخیره‌سازی

در شبکه‌ای که هر گره آن به طور میانگین d گره همسایه داشته باشد، سربار ذخیره‌سازی برای الگوریتم [۲۱] برابر $O(d^2)$ می‌باشد. در حالی که در الگوریتم پیشنهادی ما، هرگره نیاز به $2d$ واحد حافظه جهت ذخیره لیست همسایه‌های خود در جدول مسیریابی (مطابق شکل ۲-ب)، d واحد جهت نگه‌داری عمل‌های اتوماتا و d واحد جهت نگه‌داری بردار احتمالات عمل‌های اتوماتای یادگیر دارد. لذا کل سربار ذخیره‌سازی برابر $4d$ و در نتیجه از مرتبه $O(d)$ می‌باشد.

۷-۳- نتایج شبیه‌سازی‌ها

به منظورارزیابی عملکرد الگوریتم پیشنهادی چند آزمایش انجام گرفته و نتایج حاصل از این الگوریتم با نتایج به دست آمده از الگوریتم‌های ذکر شده در جدول ۱ مقایسه شده است.

برای انجام شبیه‌سازی‌ها از شبیه‌ساز J-SIM [۲۶] استفاده شده است. در این شبیه‌سازی‌ها، ۳۰۰ گره حسگر به طور تصادفی در یک محیط ۱۰۰ × ۱۰۰ متر مربع پراکنده می‌شوند. از این تعداد، ۲۰ گره به عنوان گره‌های منبع در نظر گرفته شده‌اند که در شبیه‌سازی‌ها از نظر مکانی ثابت (در لبه‌های مرزی محیط عملیاتی) می‌باشند. البته، در آزمایش ۲ به طور استثناء، فقط یک گره منبع وجود دارد. گره ایستگاه پایه به طور ثابت در مختصات (۳۰، ۵۰) قرار می‌گیرد. همه گره‌ها برد رادیویی ثابت و برابر با ۱۰ متر دارند. گره‌های بدخواه نیز به طور تصادفی از بین دیگر گره‌ها، یعنی گره‌های غیر از گره ایستگاه پایه و گره‌های منبع انتخاب می‌شوند. در تمام آزمایش‌ها (بجز آزمایش ۳) احتمال ساقط شدن بسته‌ها توسط گره بدخواه برابر با ۰/۵ می‌باشد، از پروتکل CSMA در لایه MAC برای پیاده سازی الگوریتم‌ها استفاده شده است. در همه شبیه‌سازی‌ها، گره‌های منبع در هر ۵ واحد زمانی یک بسته تولید و ارسال می‌کنند. هر یک از شبیه‌سازی‌ها به مدت ۱۰۰۰۰ واحد زمانی اجرا شده و نتیجه هر آزمایش، از میانگین ۱۰ اجرا بر روی

۱۰۰ توپولوژی مختلف شبکه به دست آمده است.

جدول ۱- پروتکل‌های مورد مقابسه با الگوریتم پیشنهادی

نام الگوریتم	پارامترهای امنیتی
Single Path Forwarding	-
Multi-Path[2]	n=5 تعداد مسیرهای مجزا
MDT[12]	n=2 تعداد جریان‌های داده‌ای
ACK-based[5]	$ACK_{span} = 2, TTL = 4, t = 1$
NWS[21]	-

معیارهای زیرجهت ارزیابی و مقایسه الگوریتم‌ها مورد استفاده قرار گرفته‌اند:

نرخ تحویل بسته: درصدی از بسته‌های تولید شده توسط گره‌های منبع که به دست ایستگاه پایه می‌رسند.

نرخ ساقط شدن بسته‌ها توسط گره‌های بدخواه: درصدی از بسته‌ها که توسط گره‌های بدخواه ساقط می‌شوند و به دست ایستگاه پایه نمی‌رسند. توجه شد که در محاسبه این معیار، صرفاً بسته‌هایی که توسط گره‌های بدخواه ساقط می‌شوند در نظر گرفته می‌شوند. یعنی بسته‌هایی که به دلیل خطای کانال و یا تصادم بسته‌ها ساقط می‌شوند لحاظ نخواهد شد.

متوسط انرژی باقی‌مانده گره‌ها: این معیار متوسط انرژی باقی‌مانده کل گره‌های سطح شبکه در طول زمان را بیان می‌کند.

آزمایش ۱: در این آزمایش، نرخ تحویل بسته‌ها در حالتی‌که تعداد گره‌های بدخواه در شبکه ۰، ۱۰، ۲۰، ... ۱۰۰ باشد را در الگوریتم پیشنهادی و چهار الگوریتم دیگر مورد ارزیابی قرار داده و نتایج بدست آمده را در شکل ۳ نشان داده‌ایم. همان‌طور که از نتایج شبیه‌سازی آشکار است، نرخ تحویل بسته‌ها در الگوریتم پیشنهادی بالاتر از الگوریتم‌های دیگر می‌باشد. چرا که، الگوریتم پیشنهادی به‌طور توزیعی و هوشمند عمل کرده و به محض مشاهده عمل بدخواهانه توسط هر گره، سریعاً آن را از مسیرهای جریان داده‌ای کنار می‌زند.

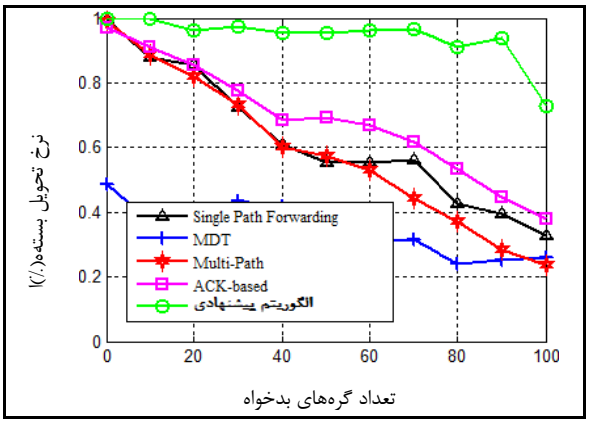
آزمایش ۲: در این آزمایش، معیار نرخ تحویل بسته‌ها در حالتی‌که فقط یک گره منبع در شبکه وجود داشته باشد و گره‌های بدخواه به طور استراتژیک در محیط شبکه پراکنده شده باشند را در الگوریتم پیشنهادی، الگوریتم NWS و الگوریتم پایه مورد ارزیابی قرار می‌دهیم. در این آزمایش، فقط یک گره منبع در شبکه وجود دارد و تعداد گره‌های بدخواه را از ۰، ۵، ۱۰، ۱۵، ... ۵۰ تغییر داده و تأثیر آن را بر نرخ تحویل بسته‌ها اندازه‌گیری می‌کنیم. البته، در این آزمایش، گره‌های بدخواه را به‌طور استراتژیک در یک ناحیه مربع، بین ایستگاه پایه و گره منبع در محیط شبکه توزیع می‌کنیم و هم‌چنین احتمال ساقط شدن بسته‌ها (DropProbability) توسط گره‌های بدخواه را ۱۰۰٪ در نظر می‌گیریم (یعنی گره بدخواه همه بسته‌های دریافتی را ساقط می‌کند). شکل ۴ نتیجه این آزمایش را نشان می‌دهد. همان‌طور که از نتایج آزمایش معلوم است، نرخ تحویل بسته‌ها در الگوریتم پیشنهادی بیش از ۹۹٪ خواهد بود درحالی که نرخ این معیار در الگوریتم NWS و الگوریتم پایه، زمانی‌که تعداد گره‌های بدخواه ۵۰ باشد، به ترتیب برابر ۶۰٪ و تقریباً ۰٪ خواهد شد. نتایج شبیه سازی حاکی از موثر بودن الگوریتم پیشنهادی در چنین شرایط خاصی می‌باشد.

آزمایش ۳: در این آزمایش، تأثیر پارامتر $T_{OverHear}$ بر نرخ تحویل بسته‌ها در الگوریتم پیشنهادی ارزیابی می‌شود. همان‌طور که پیش‌تر هم گفتیم، در پروتکل پیشنهادی، هر گره حسگر فقط بسته‌های ارسالی توسط گام- بعدی خود را شنود می‌کند تا به ماهیت (بدخواه یا نرمال) آن پی ببرد. گره u جهت شنود می‌تواند به این صورت عمل کند که به طور مرتب به ازای هر $T_{OverHear}$ (>= $T_{OverHear}$ بسته‌ای که به گام بعدی ارسال می‌کند، یک بسته را داخل بافر خود نگه دارد و به مدت زمان T_{wait} به کانال گوش دهد تا مشخص گردد گام-بعدی بسته مورد نظر را ارسال می‌کند یا خیر. $T_{OverHear}$ یک پارامتر امنیتی است و نتایج شبیه‌سازی‌ها در شکل ۵ نشان می‌دهد که $T_{OverHear}$ کوچکتر سبب تشخیص سریع‌تر ماهیت گام-بعدی می‌شود و در نتیجه بسته‌های کمتری توسط گره‌های بدخواه ساقط می‌شود و در نتیجه نرخ تحویل بسته‌ها بالا می‌رود. ولی اگر مقدار $T_{OverHear}$ بزرگ انتخاب شود سرعت تشخیص پایین می‌آید و در نتیجه تعداد بسته‌های بیشتری توسط گره بدخواه ساقط می‌شود و در نتیجه نرخ تحویل بسته‌ها پایین می‌آید. می‌توان مقدار $T_{OverHear}$ را برای هر گره ثابت و یا متغیر در نظر گرفت. اگر مقدار $T_{OverHear}$ برای هر گره، متغیر و به طور تصادفی minThershold ≤ $T_{OverHear}$ ≤ maxThershold () انتخاب شود امنیت بالاتر می‌رود. هم‌چنین، جهت مقاومت بیشتر پروتکل در برابر حمله ارسال انتخابی، هر گره حسگر می‌تواند به طور متناوب بعد از یک فاصله زمانی مشخص و یا هر بار که اتوماتای یادگیر عمل مختلفی را انتخاب کند (یا به عبارت دیگر، گره جدیدی برای گام- بعدی انتخاب کند)، می‌تواند مقدار $T_{OverHear}$ را تغییر دهد چرا که گره بدخواه ممکن است به طور تصادفی مقدار $T_{OverHear}$ را حدس بزند و بسته‌های با دوره تناوب $T_{OverHear}$ را ساقط نکند ولی بقیه بسته‌ها را ساقط کند.

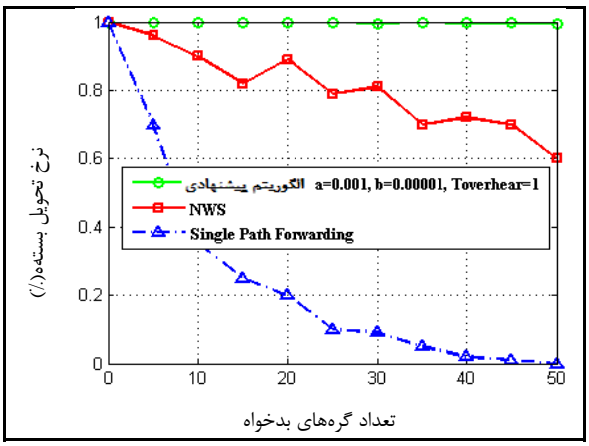
آزمایش ۴: در این آزمایش، که شکل ۶ نتایج آن را نشان می‌دهد، نرخ ساقط شدن بسته‌ها توسط گره‌های بدخواه، در حالتی‌که تعداد گره‌های بدخواه در شبکه ۰، ۱۰، ۲۰، ... ۱۰۰ باشد، در الگوریتم پیشنهادی و چهار الگوریتم دیگر مورد ارزیابی قرار گرفته است. همان‌طور که در این شکل دیده می‌شود، نرخ ساقط شدن بسته‌ها در الگوریتم پیشنهادی بسیار کمتر از دیگر الگوریتم‌ها می‌باشد. چراکه در الگوریتم پیشنهادی، پس از مشاهده عمل بدخواهانه (ساقط شدن بسته‌ها) از سوی گره‌های بدخواه، این عمل سریعاً تشخیص داده شده و این گره‌های بدخواه از مسیرهای جریان داده‌ای کنار زده می‌شوند و بسته ساقط شده مجدداً از مسیر دیگری به سمت ایستگاه پایه هدایت می‌شود. ولی در الگوریتم‌های MDT و Multi-Path رفتار بدخواهانه (ساقط شدن بسته‌ها) گره‌های بدخواه تشخیص داده نمی‌شود. هم‌چنین، در الگوریتم ACK-based عمل تشخیص گره‌های بدخواه فقط توسط ایستگاه پایه و گره‌های منبع صورت می‌گیرد، از این‌رو، الگوریتم فاقد عکس‌العمل سریع بوده و بسته‌های زیادی ساقط می‌شوند.

آزمایش ۵: در این آزمایش میزان تأثیر احتمال ساقط شدن بسته‌ها توسط گره‌های بدخواه (DropProbability) بر کارایی الگوریتم پیشنهادی مورد ارزیابی قرار گرفته است. بدین منظور، تعداد گره‌های بدخواه برابر با ۵۰ عدد در نظر گرفته شده است. DropProbability از ۰ تا ۱ تغییر داده شده و نتیجه آزمایش در قالب نرخ ساقط شدن بسته‌ها توسط گره‌های بدخواه در شکل ۷ به تصویر کشیده شده است. همان‌طور که از نتیجه آزمایش معلوم است، نرخ DropProbability تأثیر چندانی بر کارایی الگوریتم پیشنهادی ندارد، چرا که در الگوریتم پیشنهادی با تنظیم پارامتر $T_{overHear}$=1 می‌توان ساقط شدن حتی یک بسته را نیز تشخیص داد.

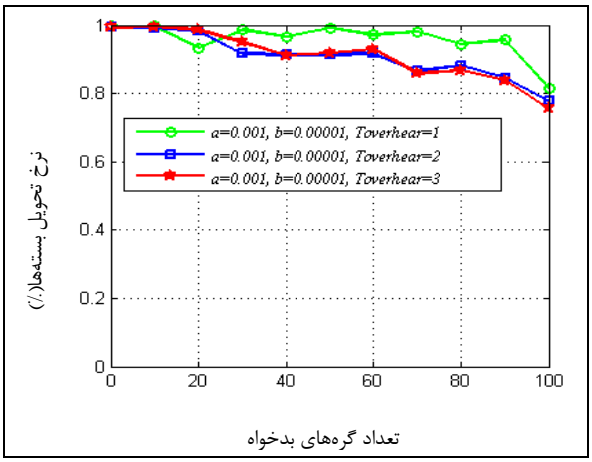
آزمایش ۶: در این آزمایش، متوسط انرژی باقی‌مانده گره‌ها در طول حیاط شبکه، در صورت استفاده از الگوریتم پیشنهادی و چهار الگوریتم دیگر مورد مقایسه قرار گرفته است. در این آزمایش، ۵۰ گره بدخواه به‌طور تصادفی انتخاب شده‌اند. انرژی اولیه تمام گره‌های حسگر ۵ ژول در نظر گرفته شده است. شکل ۸ نتیجه این آزمایش را برای زمان‌های ۸۸۰۰ تا ۱۰۰۰۰ نشان می‌دهد. نتیجه آزمایش نشان می‌دهد متوسط انرژی باقی‌مانده گره‌ها در صورت استفاده از الگوریتم پیشنهادی کمتر از الگوریتم پایه و بیشتر از دیگر الگوریتم‌ها می‌باشد.



شکل ۳- نرخ تحویل بسته‌ها در الگوریتم پیشنهادی ($T_{overHear}$=1, a=b=0.0001) و الگوریتم‌های دیگر

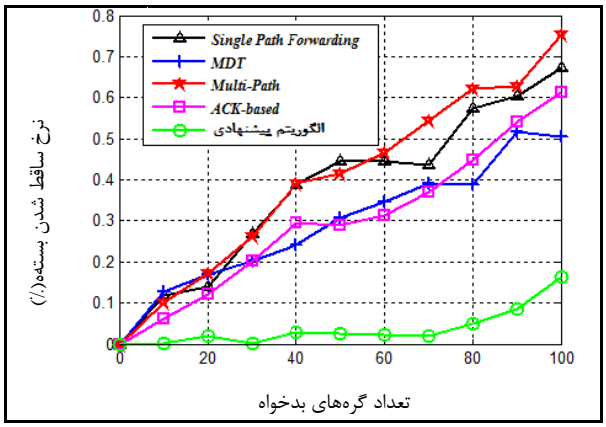


شکل ۴- مقایسه نرخ تحویل بسته‌های الگوریتم پیشنهادی با الگوریتم تک مسیره و الگوریتم NWS

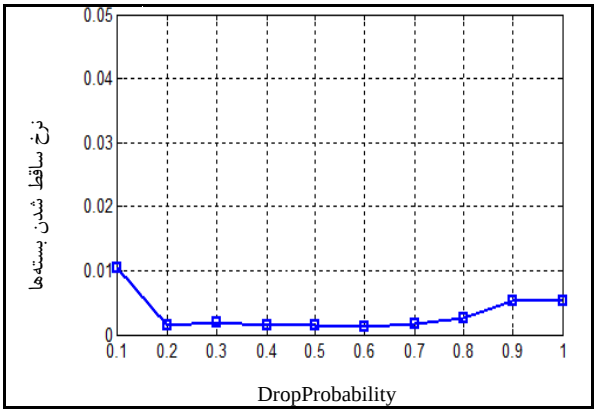


شکل ۵- تأثیر پارامتر $T_{OverHear}$ بر نرخ تحویل بسته‌ها در الگوریتم پیشنهادی

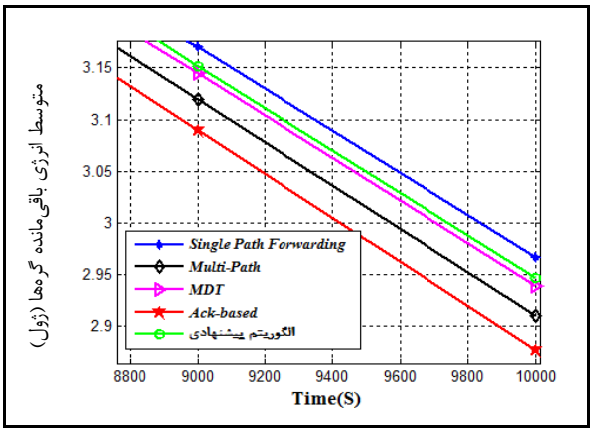
دلیل این نتیجه واضح است، الگوریتم‌های MDT و Multi-Path یک بسته داده‌ای را از طریق چندین مسیر به سمت ایستگاه پایه هدایت می‌کنند که این سبب مصرف زیاد انرژی می‌شود. همچنین، در الگوریتم ACK-based گره‌های حسگر انرژی زیادی صرف ارسال پیغام‌های ACK می‌کنند. این در حالی است که الگوریتم پیشنهادی از ارسال چند مسیره و یا ارسال پیغام‌های ACK استفاده نمی‌کند بلکه گره‌ها فقط نیاز به عمل شنود دارند که عمل شنود بسته‌ها (دریافت بسته‌ها) از نظر مصرف انرژی بسیار کم هزینه‌تر از عمل ارسال بسته‌هاست. در الگوریتم پایه، چون هیچ مکانیزمی جهت مقابله با حمله ارسال انتخابی اتخاذ نشده است، لذا بسته‌های زیادی در طول مسیرهای داده‌ای توسط گره‌های بدخواه ساقط می‌شوند (که مجدداً ارسال نمی‌شوند) و در نتیجه تعداد عمل‌های ارسال بسته‌ها توسط گره‌ها در سطح شبکه کاهش می‌یابد و این سبب حفظ انرژی گره‌های حسگر می‌شود.



شکل ۶- نرخ ساقط شدن بسته‌ها در الگوریتم پیشنهادی (a=0.001, $T_{overHear}$=1،b=0.00001) و الگوریتم‌های دیگر



شکل ۷- تأثیر پارامتر DropProbability بر نرخ ساقط شدن بسته‌ها توسط گره‌های بدخواه در الگوریتم پیشنهادی



شکل ۸- انرژی مصرفی الگوریتم پیشنهادی در مقایسه با دیگر الگوریتم‌ها

۸- نتیجه‌گیری

در این مقاله یک الگوریتم کاملاً توزیعی، پویا، سبک وزن و هوشمند مبتنی بر اتوماتاهای یادگیر جهت مقابله با حمله ارسال انتخابی در شبکه‌های حسگر بی‌سیم ارائه گردید. الگوریتم پیشنهادی، از مکانیزم شنود به همراه مدل اتوماتای یادگیر جهت انتخاب مسیر ایمن ارسال بسته‌ها در پروتکل‌های مسیریابی چندگامه استفاده می‌کند تا گره‌های بدخواه در حمله ارسال انتخابی را از مسیرهای داده‌ای کنار زند. با شبیه‌سازی پروتکل پیشنهادی و انجام آزمایش‌های مختلف، مشخص شد که الگوریتم پیشنهادی در مقایسه با الگوریتم‌های مشابه از نقطه نظر نرخ تحویل بسته‌ها، نرخ ساقط شدن بسته‌ها توسط گره‌های بدخواه و انرژی مصرفی کارآمدتر می‌باشد.

مراجع

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayircl, "A survey on sensor networks," *Proc, IEEE Communication Magazine*, vol. 40, no. 5, pp. 102-114, 2002.

[2] C. Karlof, and D. Wagner, "Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures," *Proc, Int'l Conf. Ad hoc Networks*, pp. 299-302, 2003.

[3] K. Sharma, and et al., "A Comparative Study of Various Security Approaches Used in Wireless Sensor Networks," *Proc, Int'l Journal of Advanced Science and Technology*, vol. 17, no. 4, pp. 31-44, 2010.

[4] S. Mohammadi, R. E. Atani, and H. Jadidoleslamy, "A Comparison of Link Layer Attacks on Wireless Sensor Networks," *Proc, Journal of Information Security*, pp. 69-84, 2011.

[5] B. Yu, and B. Xiao, "Detecting selective forwarding attacks in wireless sensor networks," *Proc, Int'l Workshop on Security in Systems and Networks*, pp. 1-8, 2006.

[6] B. Xiao, B. Yu, and C. Gao, "CHEMAS: identify suspect nodes in selective forwarding attacks," *Proc, Journal of Parallel and Distributed Computing*, vol. 67, no. 11, pp. 1218-1230, 2007.

[7] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: A scalable and robust communication paradigm for sensor networks," *Proc, ACM MobiCom*, pp. 56-67, 2000.

[8] C. Y. Wan, A. T. Campbell, and L. Krishnamurthy, "PSFQ: A Reliable Transport Protocol for Wireless Sensor Networks," *Proc, ACM Int'l Workshop on Wireless Sensor Networks and Applications*, pp. 1-11, 2002.

[9] L. K. Bysani, and A. K. Turuk, "A Survey on Selective Forwarding Attack in Wireless Sensor Networks," *Proc, Int'l Conf. Device and Communications*, pp. 1-5, 2011.

[10] W. Z. Kkan, Y. Xiang, and M. Y. Aalsalem, "Comprehensive Study of Selective Forwarding Attack in Wireless Sensor Networks," *Proc, IEEE Int'l Conf. Computer Network and Information Security*, pp. 1-10, 2011.

[11] S. Kaplantzis, A. Shilton, N. Mani, and Y. Sekercioglu, "Detecting selective forwarding attacks in wireless sensor networks using support vector machines," *Proc, IEEE Int'l Conf. Intelligent Sensors, Sensor Networks and Information*, pp. 335-340, 2007.

[12] H. M. Sun, C. M. Chen, and Y. C. Hsiao, "An efficient countermeasure to the selective forwarding attack in wireless sensor networks," *Proc, IEEE Int'l Conf. TENCON*, pp. 1-4, 2007.

[13] H. Y. Lee, and T. H. Cho, "Fuzzy-based reliable data delivery for countering selective forwarding in sensor networks," *Proc, Ubiquitous Intelligence and Computing*, pp. *535-544, 2007.*

[14] T. H. Hai, and E.-N. Huh, "Detecting Selective Forwarding Attacks in Wireless Sensor Networks Using Two-hops Neighbor Knowledge," *Proc, IEEE Int'l Symp. Network Computing and Applications*, pp. 325-331, 2008.

[15] J. Brown, and X. Du, "Detection of selective forwarding attacks in heterogeneous sensor networks," *Proc, IEEE Int'l Conf. Communications*, pp. 1583-1587, 2008.

[16] X. Lei, H. Yong-jun, P. Yong, and Z. Yue-Fei, "A Polynomialbased Countermeasure to Selective Forwarding Attacks in Sensor Networks," *Proc, Int'l Conf. Communications and Mobile Computing*, pp. 455- 459, 2009.

[17] C. Tumrongwittayapak, and R. Varakulsiripunth, "Detecting Sinkhole Attack And Selective Forwarding Attack In Wireless Sensor Networks," *Proc. Int'l Conf. Information, Communications and Signal Processing*, pp. 1-5, 2009.

[18] G. Li, X. Liu, and C. Wang, "A Sequential Mesh Test based Selective Forwarding Attack Detection Scheme in Wireless Sensor Networks," *Proc, Int'l Conf. Networking, Sensing and Control*, pp. 554-558, 2010

[19] W. Xin-sheng, Z. Yong-zhao, X. Shu-ming, and W. Liangmin, "Lightweight defense scheme against Selective forwarding attacks in wireless sensor networks," *Proc, IEEE Int'l Conf. Cyber-Enabled Distributed Computing and Knowledge Discovery*, pp. 226-232, 2009.

[20] Y. B. Reddy, and S. Srivathsan, "Game Theory Model for Selective Forward Attacks in Wireless Sensor Networks," *Proc, Mediterranean Conf. Control and Automation Makedonia Palace*, pp. 458-463, 2009.

[21] S. B. Lee, and Y. H. Choi, "A Resilient Packet-Forwarding Scheme against Maliciously Packet-Dropping Nodes in Sensor Networks," *Proc, ACM Workshop on Security of Ad hoc and Sensor Networks*, pp. 59-70, 2006.

[22] K. S. Narendra, and M. A. L. Thathachar, *Learning automata: An introduction*, Dover, 1989.

[23] K. S. Narendra, and M. A. L. Thathachar, "Learning automata a survey," *IEEE Trans. Systems, Man and Cybernetics*, vol. 4, no. 4, pp. 323-334, 1974.

[24] M. A. L. Thathachar, and R. H. Bhaskar, "Learning automata with changing number of actions," *IEEE Trans. Systems, Man and Cybernetics*, vol. 17, no. 6, pp. 1095-1100, 1987.

[25] S. Zhu, S. Setia, and S. Jajodia, "LEAP, Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks," *Proc, ACM Int'l Conf. Computer and Communications Security*, pp. 752-759, 2003.

[26] J-Sim Simulator, http://www.j-sim.org, April 2005.

مجتبی جمشیدی فارغ‌التحصیل کارشناسی مهندسی نرم‌افزار از دانشگاه علمی کاربردی کرمانشاه در سال ۱۳۸۸ و کارشناسی‌ارشد مهندسی نرم‌افزار از دانشگاه آزاد اسلامی قزوین در سال ۱۳۹۱. علاقمند به موضوعات شبکه‌های کامپیوتری، سیستم‌های یادگیر و امنیت.
آدرس پست‌الکترونیکی ایشان عبارت است از:
jamshidi.mojtaba@gmail.com

مهدی اثنی عشری فارغ‌التحصیل کارشناسی مهندسی نرم‌افزار، کارشناسی ارشد و دکترای هوش مصنوعی از دانشگاه صنعتی امیرکبیر تهران به ترتیب در سال‌های ۱۳۸۱، ۱۳۸۴ و ۱۳۹۰. هم‌اکنون استادیار پژوهشگاه فضای مجازی تهران بوده و علاقه‌مند به موضوعات شبکه‌های کامپیوتری، سیستم‌های یادگیر و محاسبات نرم.
آدرس پست‌الکترونیکی ایشان عبارت است از:
esnaashari@csri.ac.ir

محمدرضا میبدی فارغ‌التحصیل کارشناسی و کارشناسی ارشد اقتصاد از دانشگاه شهید بهشتی تهران به ترتیب در سال‌های ۱۳۵۲ و ۱۳۵۶. هم‌چنین فارغ‌التحصیل کارشناسی ارشد و دکترای علوم کامپیوتر از دانشگاه Oklahoma آمریکا به ترتیب در سال‌های ۱۳۵۹ و ۱۳۶۲ می‌باشد. هم‌اکنون استاد تمام دپارتمان مهندسی کامپیوتر دانشگاه صنعتی امیرکبیر تهران می‌باشد و قبل از آن، استادیار دانشگاه Western Michigan در طی سال‌های ۱۳۶۲ تا ۱۳۶۴ و استادیار دانشگاه Ohio در طی سال‌های ۱۳۶۴ تا ۱۳۷۰ بوده است. علاقمند به موضوعات مدیریت کانال در شبکه‌های سلولی، سیستم‌های یادگیر، الگوریتم‌های موازی، محاسبات نرم و توسعه نرم‌افزار.
آدرس پست‌الکترونیکی ایشان عبارت است از:
mmeybodi@aut.ac.ir

اطلاعات بررسی مقاله:

تاریخ ارسال: ۹۲/۱/۳۰
تاریخ اصلاح: ۹۲/۵/۱۲
تاریخ قبول شدن: ۹۲/۵/۱۶
نویسنده مرتبط: مجتبی جمشیدی، آموزشکده فنی و حرفه‌ای سما، دانشگاه آزاد اسلامی واحد کرمانشاه، کرمانشاه، ایران.

[1]Selective Forwarding Attack
[2]Drop
[3]Multi-Hop A know Lodgment Schema
[4]Source Node
[5]Base Station
[6]Intrusion Detection System
[7]Support Vector Machines
[8]Sliding Window
[9]Black Hole
[10]Multi-Data Flow Topology
[11]Tamper
[12]Eavesdropper and Monitor
[13]Sequential Mesh Test Based
[14]Game Theory Model
[15]Neighbor Watch System
[16]Reprogramming