*مقاله پژوهشی*

# تخصیص منابع امنیتی برای مقابله با حملات در اینترنت اشیا با استفاده از یادگیری ماشین

نسیم نوائی و وصال حکمی

**چکیده: امروزه شبکه‌های اینترنت اشیا (IoT) با توجه به محدودیت منابع پردازشی، ناهمگونی و محدودیت انرژی در اشیا و همچنین عدم وجود استانداردی واحد برای پیاده‌سازی سازوکارهای امنیتی به کانون و مرکز توجه حملات امنیتی تبدیل شده‌اند. در این مقاله، یک راهکار برای مسئله تخصیص منابع امنیتی به جهت مقابله با حملات در اینترنت اشیا ارائه خواهد شد. مسئله تخصیص منابع امنیتی در شبکه IoT (SRAIoT) به جای‌گذاری امن‌افزارها در زیرساخت IoT اشاره دارد. برای حل این مسئله نیاز است که شرایط پویای محیط ارتباطی و عدم قطعیت در مورد عملکرد مهاجمان لحاظ شود. در رویکردهای سنتی تخصیص منابع امنیتی در IoT، مهاجم بر اساس مفروضات خود از شرایط سیستم، دست به حمله زده و در مقابل، مدافع نیز در سیستم با شناخت قبلی از رفتار مهاجم و گره‌های مورد حمله به مقابله می‌پردازد. برخلاف رویکردهای پیشین در این پژوهش از رویکردی واقع‌بینانه برای تخصیص پویای منابع امنیتی در شبکه IoT جهت مقابله با مهاجمانی با رفتار ناشناخته استفاده شده است. در مسئله مطرح‌شده به این علت که در بازه‌های یادگیری در مورد استقرار چند منبع امنیتی نیاز به اتخاذ تصمیم وجود دارد، فضای حالت راهبردها به صورت ترکیبیاتی بیان می‌شود. همچنین مسئله SRAIoT در چارچوب یک مسئله قمار چندبازویی ترکیبیاتی- تخاصمی مطرح می‌شود. از آنجا که در شرایط واقعی، جابه‌جایی منابع امنیتی استقراریافته دارای هزینه بالایی است، هزینه مذکور در تابع سودمندی مسئله لحاظ شده و بنابراین چارچوب پیشنهادی به‌صورت توأمان هزینه جابه‌جایی و پاداش کسب‌شده را مد نظر قرار می‌دهد. نتایج شبیه‌سازی نشان‌دهنده همگرایی سریع‌تر معیار پشیمانی ضعیف الگوریتم‌های پیشنهادی نسبت به الگوریتم ترکیبیاتی پایه است. علاوه بر این به‌منظور شبیه‌سازی شبکه IoT در بستری واقع‌بینانه، شبیه‌سازی سناریوی حمله با استفاده از شبیه‌ساز Cooja نیز انجام شده است.**

***کلیدواژه:*** **اینترنت اشیا، تخصیص پویای منابع امنیتی، مسئله قمار چندبازویی، یادگیری ماشین.**

## ۱- مقدمه

امروزه، اینترنت اشیا [1](IoT) به‌صورتی فزاینده مورد توجه صنعت و پژوهشگران قرار گرفته است. پیش‌بینی می‌شود که تا سال ۲۰۳۰، تعداد وسایل متصل به بستر اینترنت اشیا به مرز ۳۰ میلیارد برسد. با توجه به محدودیت منابع پردازشی، ناهمگونی و محدودیت انرژی در اشیا و نیز عدم وجود استانداردی واحد برای پیاده‌سازی سازوکارهای امنیتی، این فناوری به کانون حملات امنیتی تبدیل شده است. همچنین دستگاه‌های IoT برای مهاجمانی که قصد واردکردن صدمات بزرگی را دارند، به دلایل وابستگی کاربر به دستگاه خودش و قدرت تصمیمی که به دستگاه داده می‌شود، گزینه مناسبی هستند. بنابراین نگرانی در مورد امنیت این دستگاه‌ها رو به افزایش است [۱] و [۲]. برای مقابله با حملات و موجودیت‌های خرابکار در اینترنت اشیا که از طریق دسترسی به کانال ارتباطی، کنترل تجهیزات و تزریق داده‌های نادرست، قصد تخریب کارایی سیستم را دارند، وجود زیرساخت امنیتی ضروری است. با توجه به محدودیت منابع دستگاه‌های هوشمند، استفاده از رویکردهای امنیتی قدرتمند سنتی که دارای سربار پردازشی قابل توجهی هستند، ناکارآمد است. از این رو تخصیص منابع امنیتی همچون سیستم‌های تشخیص نفوذ (IDS) و هانی پات‌ها[2] به‌منظور جمع‌آوری اطلاعات از مهاجمان و جلوگیری از حملات در بستر اینترنت اشیا مفید و اجتناب‌ناپذیر است.

### ۱-۱ انگیزه‌های توسعه پژوهش

مسئله تخصیص منابع امنیتی در شبکه اینترنت اشیا (SRAIoT) به جاگذاری و نصب امن‌افزارها در زیرساخت IoT (گره‌ها، سرخوشه‌ها یا دروازه) اشاره دارد. برای حل این مسئله نیاز است که شرایط پویای محیط ارتباطی و عدم قطعیت در مورد عملکرد مهاجمان لحاظ شود. این مسئله از پیچیدگی بالایی برخوردار بوده و به‌طور کلی با دو رویکرد می‌توان با آن مواجهه نمود. در رویکردهای سنتی تخصیص منابع امنیتی در IoT، مهاجم بر اساس مفروضات خود از شرایط سیستم، دست به حمله زده و در مقابل، مدافع نیز در سیستم با شناخت قبلی از رفتار مهاجم و گره‌های مورد حمله به جمع‌آوری اطلاعات می‌پردازد. در واقع برای محاسبه راهبرد تخصیص منابع امنیتی فرض می‌شود که مدافع از مدل ارزش‌گذاری حملات توسط مهاجم اطلاع دارد و در نتیجه می‌تواند بهترین واکنش خود را پیشاپیش محاسبه نماید [۲] تا [۵]. در حالی که در سناریوهای واقعی، طرفین اعم از مدافع و مهاجم بدون دانش و شناخت قبلی از یکدیگر در سیستم فعالیت می‌کنند. در این مقاله، برخلاف رویکردهای سنتی مذکور از رویکردی واقع‌بینانه برای تخصیص پویای منابع امنیتی در شبکه IoT جهت مقابله با مهاجمانی با رفتار ناشناخته استفاده شده است. بدین ترتیب به‌علت وجود شرایط اطلاعات نامعلوم، استفاده از رویکرد مبتنی بر یادگیری ماشین حائز اهمیت است. به‌طور کلی انگیزه‌های توسعه پژوهش به شرح زیر هستند:

این مقاله در تاریخ ۱۹ آبان ماه ۱۴۰۱ دریافت و در تاریخ ۹ خرداد ماه ۱۴۰۲ بازنگری شد.

نسیم نوائی، دانشکده مهندسی کامپیوتر، دانشگاه علم و صنعت ایران، تهران، ایران، (email: nasim_navaei@comp.iust.ac.ir).

وصال حکمی (نویسنده مسئول)، دانشکده مهندسی کامپیوتر، دانشگاه علم و صنعت ایران، تهران، ایران، (email: vhakami@iust.ac.ir).

1. Internet of Things

2. Honey Pot

۱) افزایش حملات به زیرساخت اینترنت اشیا
۲) گسترش کاربرد اینترنت اشیا در زندگی
۳) نیاز به رهیافت هوشمند برای تخصیص منابع امنیتی
۴) نیاز به حفاظت شبکه با حداقل هزینه منابع امنیتی
۵) نیاز به شبیه‌سازی و ارزیابی شبکه در بستری واقع‌بینانه

### ۱-۲ نوآوری روش پیشنهادی

ابتدا مسئله تخصیص منابع امنیتی در چارچوب یک مسئله قمار چندبازویی ترکیبیاتی عنوان می‌شود. از آنجا که جابه‌جایی منابع امنیتی استقراریافته در برخی شرایط دارای هزینه بالایی است، در مسئله مطرح‌شده سعی گردیده که این هزینه جابه‌جایی در تابع سودمندی مسئله لحاظ شود. بدین صورت که هزینه مهاجرت و جابه‌جایی منابع از یک سرخوشه به سرخوشه دیگر به‌عنوان معیار جریمه در کارایی فرایند یادگیری تأثیر داده می‌شود. در این پژوهش قصد داریم مسئله تخصیص منابع امنیتی در شبکه اینترنت اشیا را در نبود دانش آماری مهاجم به شکلی کارآمد حل کنیم. در این مسئله فضای حالت اتخاذ راهبردها ترکیبیاتی است و به همین دلیل به‌جای مسئله MAB کلاسیک با CMAB روبه‌رو هستیم. از آنجا که مهاجم سعی در کاهش کارایی شبکه دارد، مسئله از جنس تصمیم‌گیری در محیط تخاصمی می‌باشد. همچنین به علت لحاظ هزینه مهاجرت منبع از یک حوزه به حوزه دیگر شبکه، CMAB تخاصمی با هزینه جابه‌جایی (CMAB-SC) بهترین چارچوب برای مسئله تخصیص منابع امنیتی در شبکه‌های اینترنت اشیا خواهد بود.

در این مقاله برخلاف رویکردهای سنتی مذکور از رویکردی واقع‌بینانه برای تخصیص پویای منابع امنیتی در شبکه IoT جهت مقابله با مهاجمانی با رفتار ناشناخته استفاده شده است. در مسئله مطرح‌شده به این علت که در بازه‌های یادگیری در مورد استقرار چند منبع امنیتی نیاز به اتخاذ تصمیم وجود دارد، فضای حالت راهبردها به‌صورت ترکیبیاتی بیان می‌شود. همچنین مسئله SRAIoT در چارچوب یک مسئله قمار چندبازویی ترکیبیاتی- تخاصمی مطرح می‌شود. از آنجا که در شرایط واقعی، جابه‌جایی منابع امنیتی استقراریافته دارای هزینه بالایی است، هزینه مذکور در تابع سودمندی مسئله لحاظ شده است. بنابراین چارچوب پیشنهادی به‌صورت توأمان هزینه جابه‌جایی و پاداش کسب‌شده را مد نظر قرار می‌دهد.

الگوریتم پیشنهادی برای حل مسئله تخصیص منابع امنیتی نسبت به کارهای پیشین از چند جهت دارای نوآوری است:

۱) فراهم‌آوری چارچوب تخصیص منابع امنیتی برای IoT به‌صورت برخط
۲) لحاظ مهاجرت منابع امنیتی به‌عنوان معیار جریمه در کارایی فرایند یادگیری
۳) شبیه‌سازی و ارزیابی شبکه در بستری واقع‌بینانه با استفاده از Cooja

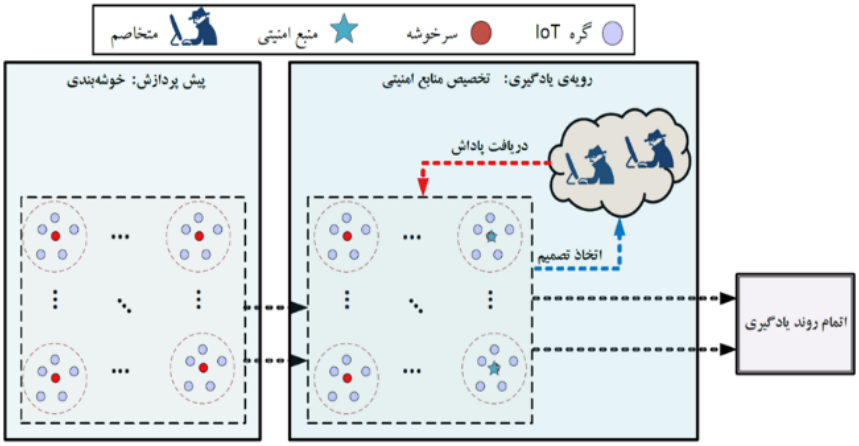
### ۱-۳ ساختار مقاله

ادامه این مقاله به‌صورت زیر ساختاربندی شده است. در بخش دوم به بررسی کارهای انجام‌شده در زمینه تخصیص منابع امنیتی در IoT پرداخته می‌شود. سپس در بخش سوم، مدل سیستم و گام‌های الگوریتم پیشنهادی تخصیص منابع امنیتی ارائه خواهد شد. در بخش چهارم، معیارهای ارزیابی و نتایج به‌دست‌آمده از ارزیابی روش پیشنهادی نمایش داده خواهد شد. در بخش پنجم، شبیه‌سازی یک سناریوی واقعی در بستر Cooja انجام گردیده و نتایج آزمایش‌های سناریوهای مختلف بررسی خواهد شد. نهایتاً در بخش آخر، نتیجه‌گیری و پیشنهادها برای کارها و پژوهش‌های آتی آمده است.

## ۲- پژوهش‌های پیشین

رفتار مدافع و مهاجم در کارهای پیشین به‌صورت یک بازی فرموله می‌شود. هر دوی بازیکن‌ها اعم از مدافع و مهاجم، تصمیم و عملی را اتخاذ کرده و بر اساس آن تصمیم، پاداش یا سود دریافت می‌کنند و طی یک روند تکراری، تصمیم خود را به‌روز می‌کنند تا زمانی که نتوانند سودمندی خود را بهبود دهند و به تعادل نش برسد. فریب دفاعی، یک رویکرد امیدوارکننده برای دفاع سایبری است. از طریق فریب دفاعی، یک مدافع می‌تواند حملات را با گمراه‌کردن یا فریب مهاجم یا مخفی‌کردن برخی از منابع خود پیش‌بینی کرده و از آن جلوگیری کند. کارهای مرتبط با حوزه فریب تدافعی متمرکز بر نظریه بازی و یادگیری ماشین است؛ زیرا این‌ها خانواده‌های برجسته‌ای از رویکردهای هوش مصنوعی هستند که به‌طور گسترده در فریب تدافعی به کار می‌روند [۶]. به طور کلی، کارهای مرتبط در سه دسته بازی فریب امنیتی و بازی استکلبرگ و روش‌های مبتنی بر یادگیری ماشین طبقه‌بندی می‌شوند.

در بخش بازی فریب امنیتی، طبق فرض کار [۱] مهاجمان معمولاً می‌توانند از طریق پویش شبکه به برخی اطلاعات داخلی مربوط به ساختار شبکه دست یابند. بدین ترتیب مهاجم قصد دارد با انتخاب آگاهانه گره قربانی از میان مجموعه تمام گره‌های قابل دسترس، پاداش مورد نظر خود را حداکثر کند. همچنین مدافع از محل دقیق حضور مهاجم در شبکه مطلع نبوده و برای قراردادن یک هانی پات جدید در لبه شبکه، متحمل هزینه ثابتی می‌شود. در این بازی، هر کدام از بازیکن‌ها در صدد افزایش تابع پاداش خود هستند؛ اما از آنجا که یک بازی مجموع صفر مدل‌سازی می‌شود، افزایش پاداش در یکی به منزله کاهش پاداش در دیگری است. در [۲] بازی تصادفی تا حدی قابل مشاهده (POSG) به جهت مدل‌سازی پویایی بازی فریب بین مهاجم و مدافع بررسی گردیده است. در چنین سناریویی، مجموعه آسیب‌پذیری‌ها و گراف حمله متغیر با زمان بوده و به این علت، مهاجم اطمینانی در مورد وضعیت واقعی شبکه ندارد. این مقاله برای درنظرگرفتن یک مدل تهدید عملی، بازی‌ای را در نظر می‌گیرد که هر دو بازیکن تا حدی راهبرد یکدیگر را مشاهده می‌کنند. در [۷] راهبرد تخصیص منابع در دو مرحله انجام می‌شود: در ابتدا برای رویکردهای تخصیص منابع، یک مسئله بهینه‌سازی سه‌هدفه محاسبه می‌گردد. در ادامه جهت کمینه‌کردن ریسک، مسئله بهینه‌سازی یک‌هدفه محاسبه می‌شود. از آنجا که این راهبرد شامل کمترین مصرف انرژی و ارزان‌ترین زیرساخت می‌باشد، جواب چنین مسئله‌ای بهینه است. مدل‌سازی تعامل در [۵] به‌صورت بازی استکلبرگ بین مدافع و مهاجم می‌باشد. همچنین مهاجم برای انجام حمله باید حداقل یک منبع امنیتی را به خطر بیندازد. بنابراین مهاجم باید حداقل به یک گره دسترسی داشته و از منابع امنیتی که مدافع در سطح شبکه گذاشته باخبر می‌باشد. هدف مدافع، تأمین امنیت کل شبکه با انتخاب و جایگذاری درست منابع است؛ به قسمی که منبع امنیتی بتواند به بهترین شکل به حملات رسیدگی کند. در [۸] یک رویکرد فریب آنلاین پیشنهاد شده است. مدافع یک باور متشکل از یک حالت امنیتی را حفظ می‌کند؛ در حالی که اقدامات حاصل به عنوان فرایند تصمیم‌گیری مارکوف جزئی قابل مشاهده (POMDP) مدل می‌شود. این مدل مبتنی بر یادگیری تقویتی فرض می‌کند که باور مدافع در مورد پیشرفت مهاجم از طریق یک سیستم تشخیص نفوذ مبتنی بر شبکه (NIDS) مشاهده می‌شود. در [۹] یک راهبرد استقرار بهینه برای منابع




شکل ۱: مدل سیستم و نمایی کلی از روند یادگیری مسئله SRAIoT.

فریب مانند هانی پات‌ها شناسایی شد. یک الگوریتم یادگیری $Q$ را برای یک سیاست استقرار هوشمند ایجاد کردند تا منابع فریب را با تغییر وضعیت امنیت شبکه به‌صورت پویا قرار دهند. با تجزیه و تحلیل راهبرد مهاجم در شرایط عدم قطعیت و راهبردهای یک مدافع با چندین خط مشی مکان استقرار، یک بازی مهاجم- مدافع در نظر گرفته شده است.

به‌عنوان جمع‌بندی در عمده کارهای موجود فرض بر این است که مدافع از مدل ارزش‌گذاری حملات توسط مهاجم باخبر است؛ بنابراین پیشاپیش بهترین واکنش خود را با توجه به شرایط موجود محاسبه می‌کند. در مقابل، مهاجم نیز حمله را بر اساس مفروضات خود از شرایط سیستم شروع می‌کند. همچنین به علت پیچیدگی بالای فضاهای عملیاتی، امکان مدل‌سازی دقیق یک حمله هنگام انجام حملات متعدد وجود ندارد و در شرایطی که شبکه تحت تأثیر چندین حمله قرار می‌گیرد، درنظرگرفتن تعاملات فقط میان یک مهاجم و یک مدافع کافی نیست. یک سناریوی واقع‌بینانه این است که در مدل‌سازی، طرفین (اعم از مدافع و مهاجم) شناخت کاملی از پارامترهای تابع هدف رقیب ندارند؛ بنابراین نیاز به روشی تطبیقی مبتنی بر یادگیری برخط جهت تخصیص منابع امنیتی به شبکه IoT است.

## ۳- مدل سیستم

شکل ۱، مدل سیستم و نمایی کلی از روند یادگیری مسئله تخصیص منابع امنیتی در اینترنت اشیا (IoT) را نشان می‌دهد. گره‌های اینترنت اشیا با توجه به محدودیت منابع پردازشی، ناهمگونی و محدودیت انرژی در اشیا و نیز عدم وجود استانداردی واحد برای پیاده‌سازی سازوکارهای امنیتی به کانون و مرکز توجه حملات امنیتی تبدیل شده است. پیاده‌سازی رویکردهای امنیتی سنتی به علت محدودیت انرژی و هزینه بالا، مناسب این شبکه نیست؛ بنابراین نیازمند استفاده از رویکردهای جدید و متناسب با محدودیت‌ها و چالش‌های این بستر هستیم. این شبکه به علت کمبود منابع امنیتی در معرض حملات بوده و این حملات می‌توانند انواع مختلفی مانند حمله Rank و حمله Sinkhole داشته باشند.

در این مقاله، شبکه اینترنت اشیا به‌صورت گراف وزن‌دار غیرجهت‌دار با $N$ گره در نظر گرفته شده است. این شبکه گرافی با کمک الگوریتم مجموعه ناوابسته حریصانه خوشه‌بندی گردیده و تعداد $K$ سرخوشه به جهت کارگذاری منابع امنیتی مشخص می‌شود. معماری روش پیشنهادی مبتنی بر شیار زمانی است و بنابراین کل دوره زمانی اجرای الگوریتم به مجموعه $T$ از شیارهای زمانی گسسته می‌شود. همچنین مجموعه مهاجم‌ها توسط مجموعه $M$ نمایش داده می‌شود. در این کار فرض بر این است که مهاجم از نوع متخاصم بوده و تمام اقدامات مدافع را از پیش می‌داند و دقیقاً همانند وی از الگوریتم یادگیری هوشمندانه استفاده می‌کند تا بتواند حمله را انجام دهد. ایده کلی این کار از [۱۰] که مختص تخصیص رادیو به کانال‌های شبکه‌های رادیویی شناختی است، گرفته شده و گام‌های الگوریتم مختص شبکه اینترنت اشیا، شخصی‌سازی و تغییر داده شده است.

مدافع در ابتدای شیار زمانی $(t \in T)$ یک راهبرد از مجموعه راهبرد $S$ انتخاب می‌کند که راهبرد منتخب به‌صورت $X_t$ نشان داده می‌شود. بدین ترتیب، راهبرد منتخب در شیار زمانی بعدی $(t+1)$ به‌صورت $X_{t+1}$ معین می‌شود. در این راستا جابه‌جایی منبع از یک سرخوشه به سرخوشه دیگر، مقداری هزینه به همراه دارد؛ بنابراین هزینه جابه‌جایی از راهبرد $X_t$ به راهبرد $X_{t+1}$ به‌صورت

$$c(X_{t-1}, X_t) \in [0,1] \qquad (1)$$

فرموله می‌شود. واضح است اگر راهبرد در شیار زمانی بعدی عوض نشود، این هزینه برابر با صفر خواهد بود. برای سادگی کار، هزینه جابه‌جایی برای اولین شیار زمانی بدون توجه به اینکه $X_1$ چیست، برابر با مقدار زیر تنظیم می‌شود

$$c(X_0, X_1) = c_0 \qquad (2)$$

پاداش دریافتی از تشخیص موفقیت‌آمیز حمله توسط منبع امنیتی روی سرخوشه $k$ در راهبرد $s$ در شیار زمانی $t$ به‌صورت زیر فرموله می‌شود

$$f_{k,t} = \begin{cases} r_1, & \text{if Clusterhead } k \in X_t \text{ and} \\ & \text{attack is detected} \\ 0, & \text{otherwise} \end{cases} \qquad (3)$$

پاداش دریافتی از راهبرد منتخب برابر با مجموع پاداش‌های دریافتی از هر سرخوشه در شیار زمانی $t$ است که به‌صورت زیر فرموله می‌شود

$$g_{s,t} = \begin{cases} \sum_{k \in s} f_{k,t}, & \text{if } s = X_t \\ 0, & \text{otherwise} \end{cases} \qquad (4)$$

### ۳-۱ سنجش و ارزیابی روش پیاده‌سازی‌شده

به‌طور استاندارد، معیار کارایی برای ارزیابی عملکرد الگوریتم‌ها یا عامل‌های یادگیرنده در MAB به نام «معیار پشیمانی» است. برای یک عامل یادگیرنده، پشیمانی در هر لحظه از زمان به‌صورت اختلاف میانگین زمانی پاداش‌های به‌دست‌آمده از بازوهای منتخب توسط عامل یادگیرنده


شکل ۲: شیارهای زمانی به‌ازای دسته زمانی $\tau = ۴$.

با پاداش متوسط بهینه تعریف می‌شود. مفروض است که مدافع در طول افق زمانی $T$ از دنباله راهبرد تولیدشده یعنی $X_۱, X_۲, \ldots, X_T$ توسط الگوریتم $A$ پیروی می‌کند. در پایان شیار زمانی $T$، پاداش راهبرد تجمعی به‌صورت (۵) تعریف می‌شود

$$G_A \overset{def}{=} \sum_{t=۱}^{T} g_{X_t,t} \qquad (۵)$$

در این بین، مدافع متحمل هزینه جابه‌جایی تجمعی می‌شود

$$L_A \overset{def}{=} \sum_{t=۱}^{T} c(X_{t-۱}, X_t) \qquad (۶)$$

در نتیجه، میزان سودمندی الگوریتم $A$ از منابع امنیتی تخصیص‌داده‌شده برابر با میزان اختلاف پاداش تجمعی الگوریتم و هزینه جابه‌جایی تجمعی آن است

$$U_A = G_A - L_A \qquad (۷)$$

برای ارزیابی الگوریتم اجرایی، از حالت خاص پشیمانی در بدترین حالت، یعنی پشیمانی ضعیف به عنوان معیار استفاده می‌شود. برای محاسبه میزان این پشیمانی، نیاز داریم تا میزان اختلاف بین سودمندی از بهترین حالت الگوریتم و الگوریتم اجرایی $A$ به‌دست آید. میزان سودمندی الگوریتم بهترین حالت، زمانی است که بین بازه‌های زمانی $t$ هرگز راهبرد عوض نشود؛ بنابراین هزینه جابه‌جایی به‌جز در بازه زمانی اولیه برابر با صفر است و در نتیجه داریم

$$L_{best} = c. \qquad (۸)$$

همچنین این الگوریتم می‌بایست بیشترین میزان پاداش را داشته باشد؛ بنابراین از بین راهبردها، راهبردی انتخاب می‌شود که بیشترین میزان پاداش را دارد و در نتیجه، میزان سودمندی این الگوریتم بدین صورت نمایش داده خواهد شد

$$U_{best} = G_{best} - U_A \qquad (۹)$$

بنابراین می‌توانیم میزان پشیمانی را محاسبه کنیم

$$R_A = U_{best} - U_A \qquad (۱۰)$$

برای کنترل موازنه بین پاداش و هزینه جابه‌جایی، تمام شیارهای زمانی را به دسته‌های زمانی متوالی و جدا از هم گروه‌بندی می‌کنیم. ما در هر دسته زمانی به همان راهبرد پایبند هستیم تا از هزینه جابه‌جایی جلوگیری کنیم. بین دسته‌ها، یک راهبرد مجدداً به جهت دریافت پاداش‌های بالاتر انتخاب می‌شود. اندازه دسته زمانی کوچک‌تر ممکن است منجر به پاداش بیشتر اما هزینه جابه‌جایی بیشتر شود؛ در حالی که اندازه دسته زمانی بزرگ‌تر ممکن است منجر به هزینه جابه‌جایی کمتر اما پاداش کمتر شود. با توجه به پارامتر تعیین دسته زمانی $(\tau)$، شیارهای زمانی $\{۱, ۲, \ldots, t, \ldots, T\}$ به دسته‌های زمانی متوالی و جدا از هم تقسیم می‌شوند

$$J = \left\lceil \frac{T}{\tau} \right\rceil \qquad (۱۱)$$

به‌طوری که برای $۱ \le j \le J$ داریم

جدول ۱: احتمال راهبرد در الگوریتم‌ها [۱۰].

| نام الگوریتم | احتمال راهبرد در دسته زمانی $j$ |
|---|---|
| SRIoT-۱ | $p_{s,j} = (۱-\gamma) \times \frac{w_{s,j}}{W_j} + \frac{\lambda}{S}$ |
| SRIoT-۲ | $p_{s,j} = \frac{w_{s,j}}{W_j}$ |
| SRIoT-۳ | $p_{s,j} = (۱-\gamma) \times \frac{w_{s,j}}{W_j} + \frac{\gamma}{C} \mathrm{I}_{s \in C}$ |

جدول ۲: وزن راهبرد در الگوریتم‌ها [۱۰].

| نام الگوریتم | وزن راهبرد در دسته زمانی $j+۱$ |
|---|---|
| SRIoT-۱ | $w_{s,j+۱} = w_{s,j} \exp \frac{\gamma}{S} \frac{\frac{۱}{\tau} \times \sum_{i=۱}^{\tau} f_{k,[j]+i}}{\sum_{s:k \in s} P_{s,j}}$ |
| SRIoT-۲ | $w_{s,j+۱} = w_{s,j} \exp(-\eta \frac{\frac{۱}{\tau} \times \sum_{i=۱}^{\tau} f_{k,[j]+i}}{\sum_{s:k \in s} P_{s,j}})$ |
| SRIoT-۳ | $w_{s,j+۱} = w_{s,j} \exp(\eta \sum_{k \in s} \frac{\frac{۱}{\tau} \times \sum_{i=۱}^{\tau} f_{k,[j]+i} + \beta}{(۱-\gamma) \frac{\sum_{s:k \in s} w_{s,j}}{W_j} + \frac{\gamma}{C} \times C_k})$ |

$$[j] = (j-۱)\tau \qquad (۱۲)$$

بنابراین دسته زمانی $j$ - همان طور که در شکل ۲ آمده است- از شیار زمانی $[j]+۱$ شروع شده و در شیار زمانی $[j]+\tau$ به پایان می‌رسد.

## ۳-۲ الگوریتم پیشنهادی

برای مسئله تخصیص منابع امنیتی در شبکه‌های اینترنت اشیا با توجه به شرایط مسئله در یک چارچوب یکسان، سه الگوریتم یادگیری تقویتی مورد بررسی و تجزیه و تحلیل قرار گرفت. برای بحث تئوری، الگوریتم به توضیح مراحل SRAIoT۳ اکتفا کرده و در مورد توضیح فرمول به‌روزرسانی وزن راهبرد و نهایتاً وزن سرخوشه در این الگوریتم خواهیم پرداخت. زیرا این الگوریتم، رفتار نسبتاً بهتری از الگوریتم‌های قبلی داشته و به‌علت معرفی مفهوم جدیدی از مجموعه پوششی راهبردها در مقیاس‌های بزرگ‌تر، سریع‌تر از بقیه به راهبرد بهینه نزدیک می‌شود. فرمول توابع اصلی هر سه الگوریتم در جداول ۱ و ۲ قابل مشاهده است. در این جداول، فرمول‌های احتمال راهبرد در دسته زمانی $j$ و وزن راهبرد در دسته زمانی $j+۱$ آورده شده است.

در ادامه این بخش، مراحل مدل سیستم به تفکیک مورد بررسی قرار خواهند گرفت.

### ۳-۲-۱ محاسبه راهبردهای مختلف

از آنجا که حالت‌های مختلفی برای تخصیص منابع امنیتی بر روی سرخوشه‌های شبکه اینترنت اشیا وجود دارد، مدافع با راهبردهای مختلفی برای کارگذاری منابع امنیتی روبه‌رو خواهد بود. این مسئله به دنبال تخصیص $l$ منبع محدود امنیتی به $k$ سرخوشه است؛ بنابراین تعداد کل راهبردها برابر با مجموعه $S$ بوده و به‌صورت زیر تعریف می‌شود

$$S = \binom{K}{l} \qquad (۱۳)$$

در الگوریتم SRAIoT-۳، معیار پشیمانی ضعیف واقعی با $O(T^{\frac{2}{3}})$ با هر گونه اطمینان تعریف‌شده توسط کاربر محدود می‌شود. علاوه بر این با معرفی یک مفهوم جدید به نام مجموعه هم‌پوشان راهبرد[1]، ضریب کران پشیمانی ضعیف از $O(\sqrt[3]{S}\ln S)$ به $O(\sqrt[3]{C}\ln S)$ کاهش پیدا می‌کند که در آن $C \le K$ است.

### ۳-۲-۲ محاسبه احتمال راهبردها

هر راهبرد مدافع با یک احتمال مشخص در هر دسته زمانی $j$ می‌تواند انتخاب گردد و همچنین در طول کل دسته زمانی $j$، راهبرد عوض نمی‌شود. احتمال راهبرد مدافع با $p_{s,j}$ مشخص گردیده است و بر اساس وزن راهبرد محاسبه می‌شود. برای محاسبه احتمالات راهبرد، مفهوم جدیدی به نام مجموعه هم‌پوشان راهبرد معرفی می‌شود. مجموعه هم‌پوشان راهبرد به مجموعه‌ای از راهبردها اطلاق می‌گردد که تمام سرخوشه‌های $K$ را پوشش می‌دهند؛ به قسمی که سرخوشه $k \in K$ توسط $C$ پوشش داده می‌شود، اگر راهبرد $s \in S$ وجود داشته و سرخوشه $k$ در این راهبرد حضور داشته باشد $(k \in s)$. این مجموعه، زیرمجموعه‌ای از مجموعه تمام راهبردها است $(C \subset K)$. وزن اولیه برای هر راهبرد بر اساس دانش گذشته حمله مهاجم و اهمیت سرخوشه تعیین می‌گردد. برای محاسبه احتمال انتخاب هر راهبرد در هر دسته زمانی از (۱۴) استفاده می‌شود

$$p_{s,j} = (1-\gamma)\times\frac{w_{s,j}}{W_j} + \frac{\gamma}{C}\mathrm{I}_{s\in C} \qquad (۱۴)$$

پارامتر $\gamma$ برای محاسبه احتمال راهبردها و ایجاد توازن میان اکتشاف و بهره‌برداری استفاده می‌شود. اولین عبارت (۱۴) بهره‌برداری از راهبردهایی با سابقه پاداش خوب است و دومی، اکتشاف تمام راهبردها را تضمین می‌کند. $\mathrm{I}_{s\in C}$ تابع نشان‌گر است؛ اگر $s \in c$ باشد، مقداری برابر با عدد یک خواهد داشت و در غیر این صورت برابر با صفر خواهد بود. به این ترتیب راهبردهای موجود در مجموعه هم‌پوشان، بیشتر از سایرین انتخاب می‌شوند. در نتیجه، SRIoT-۳ می‌تواند همه سرخوشه‌ها را سریع‌تر کشف کند و فرایند اکتشاف برای بهترین راهبرد تسریع می‌شود. همچنین در (۱۴)، $W_j$ مجموع وزن راهبردها است؛ به قسمی که داریم

$$W_j = \sum_{s\in S} w_{s,j} \qquad (۱۵)$$

### ۳-۲-۳ انتخاب راهبرد

در مرحله قبل، احتمال انتخاب هر راهبرد مدافع در دسته زمانی $j$ محاسبه شد. احتمال راهبردها در قالب یک آرایه تعریف گردیده و از این توزیع احتمال، نمونه‌گیری اولیه می‌شود. خروجی این پیاده‌سازی، شاخص راهبرد $(j)$ در مجموعه راهبردها $(S)$ است و بدین ترتیب، راهبرد منتخب $Z_j \in S$ در تکرار کنونی به‌دست می‌آید. راهبرد منتخب در دسته زمانی $j$، یعنی $Z_j$ برای تمام شیارهای زمانی $\tau$ در دسته زمانی $j$، یکسان و بدون تغییر باقی می‌ماند. به بیان دیگر اگر راهبرد انتخابی جهت تخصیص منابع امنیتی در دسته زمانی $j$ام برابر با $Z_j$ باشد، به ازای $1 \le i \le \tau$ داریم

$$X_{([j])+i} = Z_j \qquad (۱۶)$$

از این رو هزینه جابه‌جایی $c(Z_{j-1}, X_j)$ برای دسته زمانی $j$، تنها یک بار رخ می‌دهد و مدافع بر اساس حمله‌ای که رخ می‌دهد، پاداشی دریافت می‌کند. مدافع، سوابق $f_{k,[j]+i}$ را برای همه $k \in Z_j$ و $1 \le i \le \tau$ نگه می‌دارد. پاداش راهبردی که توسط مدافع به‌دست می‌آید، مجموع تمام پاداش‌های دریافتی از سرخوشه‌های نظارت‌شده است. ماتریس $P_d(M_k)$ بیانگر احتمال تشخیص موفقیت‌آمیز حضور تعداد $M_k$ مهاجم است. درایه‌های این ماتریس با منطق افزایش مقدار احتمال تشخیص با دو عامل تخصیص منبع امنیتی به سرخوشه $k$ و تعداد حمله مهاجمین به آن سرخوشه محاسبه می‌شوند. بنابراین با الهام از [۱۰] به ازای هر حمله به سرخوشه داریم

$$P_d(M_k) = 0.9\times(\mathrm{AttackToClusterHead}\times 0.01) \qquad (۱۷)$$

مدل حمله مهاجم بر آن اساس است که راهبردش با استفاده از نمونه‌گیری از توزیع احتمال راهبردهای به‌دست‌آمده تعیین می‌شود. برخلاف راهبرد تخصیص منابع امنیتی که در هر تکرار و دسته زمانی به طول $\tau$، راهبرد تغییر نمی‌کند، راهبرد مهاجمین در هر طول تکرار عوض می‌شود.

### ۳-۲-۴ به‌روزرسانی وزن راهبردها

وزن‌های راهبردها در انتهای هر دسته زمانی بر طبق مراحل بعدی به‌روزرسانی می‌شود. در قدم اول لازم است هر زمان که منبع امنیتی نصب‌شده بر روی سرخوشه $k$ موفق به تشخیص حمله در هر زمان $i$ در دسته زمانی $\tau$ شد، پاداش دریافتی از آن در $f_k$ به‌عنوان پاداش سرخوشه برای سرخوشه $k$ نگهداری شود. در انتهای دسته زمانی، متوسط پاداش دریافتی سرخوشه برای سرخوشه $k$، دارای منبع امنیتی در دسته زمانی $j$ به‌صورت زیر محاسبه می‌شود

$$\overline{f}_{k,j} = \frac{1}{\tau}\sum_{i=1}^{\tau} f_{k,[j]+i} \qquad (۱۸)$$

هر واحد پاداش دریافتی برابر با مقدار مشخص $r$ بوده که نهایتاً برابر با معکوس تعداد منابع امنیتی است. با درنظرگرفتن (۳) داریم

$$\overline{f}_{k,j} \in [0, r] \qquad (۱۹)$$

در قدم بعد، احتمال انتخاب سرخوشه با جمع‌کردن احتمالات راهبردهای شامل آن سرخوشه به‌صورت زیر محاسبه می‌شود

$$q_{k,j} = \sum_{s:k\in s} p_{s,j} = (1-\gamma)\frac{\sum_{s:k\in s} w_{s,j}}{W_j} + \frac{\gamma}{C}\times C_k \qquad (۲۰)$$

و در آن $C_k$ تعداد راهبردهایی را نشان می‌دهد که در مجموعه هم‌پوشان راهبرد حضور داشته و شامل سرخوشه $k$ هستند

$$C_k = \left|\{S \mid s \in C \wedge k \in s\}\right| \qquad (۲۱)$$

برای محاسبه میانگین امتیاز سرخوشه نیاز به پارامتر $\beta$ است؛ بنابراین بر اساس (۱۸) و (۲۰)، متوسط امتیاز سرخوشه برای سرخوشه $k$ در دسته زمانی $j$ به‌صورت زیر محاسبه می‌شود

$$\overline{f}'_{k,j} = \frac{\overline{f}_{k,j} + \beta}{q_{k,j}} \qquad (۲۲)$$

از پارامتر $\beta$ برای کاهش تبعیض[2] مابین سرخوشه‌های دارای منبع امنیتی و سرخوشه بدون منبع امنیتی استفاده می‌گردد. سپس وزن هر سرخوشه توسط فرمول زیر به‌روزرسانی می‌شود

$$h_{k,j+1} = h_{k,j}\exp(-\eta\overline{f}'_{k,j}) \qquad (۲۳)$$

1. Covering Strategy Set

2. Bias

**SRAIoT Algorithm**

**Parameter**: $\tau \in [1, T]$
**Initialization**: $w_{s,1} \leftarrow W$ for all $s \in S$, $J \leftarrow \lceil T/\tau \rceil$.
**For** $j \leftarrow 1,\ldots,J$ **do**
  Calculate the strategy probability $p_{s,j}$ for all $s \in S$ using Eq.(14)
  Choose strategy $Z_j \in S$ randomly according to the probability distribution $p_{1,j}, \ldots, p_{s,j}$ and incur switching cost $c(Z_{j-1}, Z_j)$
  Defender using $Z_j$ for $\tau$ timeslots, i.e, $X_{[j]+i} \leftarrow Z_j$ for $1 \le i \le \tau$ and record the reward of each monitored ClusterHead, $f_{k,[j]+i}$ for all $k \in Z_j$, $1 \le i \le \tau$
  Update strategy weight $w_{s,j+1}$ for all $s \in S$ using Eq.(25)
**end**

شکل ۳: شبه‌کد الگوریتم تخصیص منابع امنیتی.

نهایتاً تعریف رسمی وزن راهبرد به‌صورت زیر است

$$W_{s,j} = \prod_{k \in s} h_{k,j} \tag{۲۴}$$

با ترکیب (۲۲) و (۲۳) می‌توان مستقیماً وزن راهبرد برای هر راهبرد $s \in S$ را به‌روزرسانی کرد

$$w_{s,j+1} = w_{s,j} \exp(-\eta \bar{g}'_{s,j}) \tag{۲۵}$$

در حالی که $\bar{g}'_{s,j}$ متوسط امتیاز راهبرد برای هر راهبرد $s \in S$ بوده و به‌صورت زیر محاسبه می‌شود

$$\bar{g}'_{s,j} = \sum_{k \in s} \bar{f}'_{k,j} \tag{۲۶}$$

شایان ذکر است با ترکیب (۱۸)، (۲۰) و (۲۲) می‌توان به‌صورت مستقیم $\bar{g}'_{s,j}$ را محاسبه کرد

$$\bar{g}'_{s,j} = \sum_{k \in s} \frac{\frac{1}{\tau} \times \sum_{i=1}^{\tau} f_{k,[j]+i} + \beta}{(1-\gamma)\frac{\sum_{s:k \in s} w_{s,j}}{W_j} + \frac{\gamma}{C} \times C_k} \tag{۲۷}$$

## ۳-۳ شبه‌کد الگوریتم پیشنهادی

تمامی قدم‌های الگوریتم که در بخش‌های پیشین به تفصیل توضیح داده شد در شبه‌کد شکل ۳ آمده است. خط دوم آن، نمایانگر بحث نمونه‌گیری مذکور در بخش پیشین است و برخلاف [۱۰] در ابتدای کار، وزن راهبردها برابر یک نیست.

با اجرای این الگوریتم، معیار پشیمانی ضعیف به‌طور حدی به صفر همگرا می‌شود. با استناد به قضیه دوم از [۱۰] به ازای هر نوعی از مهاجم و با احتمال حداقل $1-\gamma$، معیار پشیمانی ضعیف الگوریتم SRIoT-۳ توسط $O(T^{\frac{2}{3}})$ محدود می‌شود. بنابراین مقادیر ابرپارامترهای الگوریتم لازم است به‌طور مشخص به شکل زیر تعریف شوند تا برای معیار پشیمانی ضعیف، همگرایی به سمت صفر اتفاق بیفتد. همچنین ضرایب $B_\tau$، $B_\gamma$ و $B_\beta$ ثابت هستند

$$\tau = B_\tau T^{\frac{1}{3}} \in [1, T] \tag{۲۸}$$

$$\gamma = B_\gamma T^{-\frac{1}{3}} \in (0, \frac{1}{2}) \tag{۲۹}$$

$$\beta = B_\beta T^{-\frac{1}{3}} \in (0, 1) \tag{۳۰}$$

$$\eta = \frac{B_\gamma}{2lC} \times T^{-\frac{1}{3}} \tag{۳۱}$$

## ۴- شبیه‌سازی و ارزیابی روش

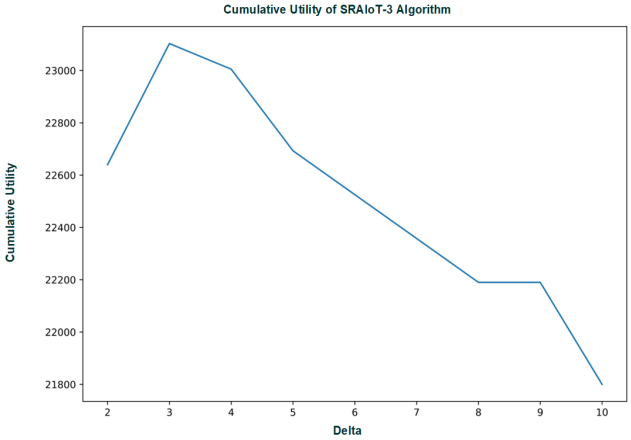
به‌منظور بررسی و نمایش عملکرد الگوریتم‌های پیشنهادی برای استقرار منابع امنیتی در شبکه‌های اینترنت اشیا، آزمایش‌ها و شبیه‌سازی‌های گسترده‌ای انجام شد. کدهای الگوریتم‌ها با استفاده از زبان برنامه‌نویسی پایتون نوشته و روی سیستم Core i۷ هشت‌هسته‌ای با RAM ۶۴ GB و Cache ۱۲ MB اجرا شده است. لازم به ذکر است نتایج شبیه‌سازی در ادامه آمده و هر یک از آنها به‌طور متوسط بیش از ۱۰۰ آزمایش، تکرار و محاسبه گردیده و نتایج آن به شیوه‌ای خودکار توسط اسکریپت‌ها مجزا و به نمودار تبدیل شده است. ما ابتدا همگرایی معیار پشیمانی‌های ضعیف نرمال‌شده هر سه الگوریتم را به همراه الگوریتم پایه مورد مقایسه نشان داده‌ایم و سپس عملکرد آنها را به ازای مهاجم هوشمند مورد مطالعه و مقایسه قرار می‌دهیم. همچنین درباره آنکه چگونه ابرپارامترهای[1] الگوریتم بر عملکرد الگوریتم‌های پیشنهادی تأثیر می‌گذارند، بحث می‌کنیم. برای این کار علاوه بر ارجاع به قضایای [۱۰] از برخی ابرپارامترهای مهم به ازای مقادیر مختلف اجرا گرفته شده است.

### ۴-۱ مفروضات و پارامترهای ارزیابی

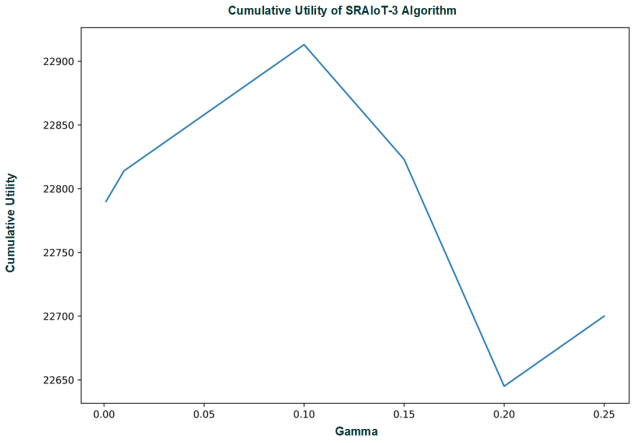
در این شبیه‌سازی، یک گراف وزن‌دار غیرجهت‌دار به‌منزله شبکه اینترنت اشیا حضور پیدا می‌کند و نیز برای سادگی، تنها یک نوع منبع امنیتی در تنظیمات شبیه‌سازی وجود دارد. همچنین فرض بر آن است که دو مهاجم از نوع سازگار[2] (هوشمند)، قصد حمله به شبکه دارند. در تنظیم مهاجم سازگار (هوشمند)، هر مهاجم از حالت SRIoT-۱ اصلاح‌شده[3] استفاده می‌کند؛ به عبارتی، نسخه غیرترکیبیاتی استفاده شده و تعداد بازوهای انتخابی هر مهاجم به تعداد سرخوشه می‌باشد. سایر پارامترهای شبیه‌سازی در جدول ۳ آمده‌اند.

روش پایه مورد مقایسه در این پژوهش، الگوریتم CUCB[4] بوده که توسعه‌ای بر الگوریتم UCB۱ است. به عبارت دیگر، الگوریتم CUCB، توسعه ترکیبیاتی الگوریتم UCB می‌باشد که این روش در [۱۱] شرح داده شده است. به‌روزرسانی وزن بازوی ترکیبیاتی در این الگوریتم از طریق رابطه زیر انجام می‌شود

1. Hyperparameter
2. Adaptive Adversary
3. Modified SRAIoT1
4. Combinatorial Upper Confidence Bound




شکل ۴: تأثیر پارامتر دسته زمانی بر سودمندی تجمعی.




شکل ۵: تأثیر پارامتر $\gamma$ بر روی سودمندی تجمعی.

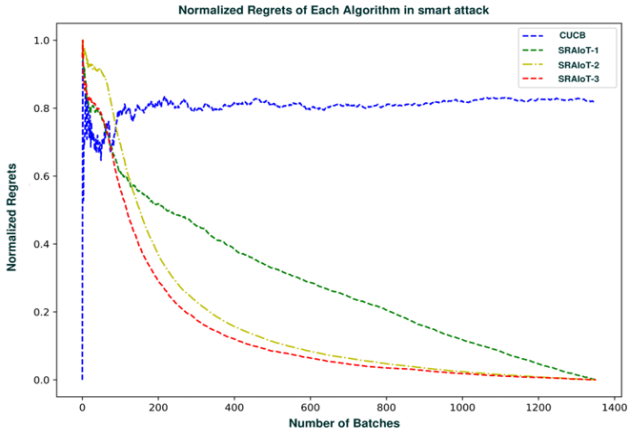
$$\mu_i = \mu_i' + \sqrt{\frac{۳\ln t}{۲T_i}} \qquad (۳۲)$$

## ۴-۲ نتایج ارزیابی

در این بخش با تکرار آزمایش‌ها جهت ارزیابی روش پیشنهادی با معیارهای مورد بحث، نتایج به‌دست‌آمده را به تفکیک در هر نمودار مشخص کرده و به تحلیل و بررسی کارایی روش مطرح‌شده می‌پردازیم. همچنین نمودارهایی برای مقایسه روش پیشنهادی با کار مقایسه‌ای، تحلیل خواهد شد.

### ۴-۲-۱ تأثیر پارامترهای الگوریتم

$\tau$ در میان تمام پارامترهای هر سه الگوریتم، مهم‌ترین آنها و اندازه دسته زمانی است که موازنه بین پاداش تجمعی و هزینه جابه‌جایی تجمعی را کنترل می‌کند. شبیه‌سازی در شرایطی انجام شده که بزرگی پارامتر دسته زمانی $\tau$ از رابطه $T^{\frac{۱}{۵}}$ به‌دست می‌آید. ما به ازای $\tau$ مختلف از الگوریتم‌های پیشنهادی اجرا گرفته و پاداش تجمعی محاسبه می‌شود. نمودار برای مهاجم سازگار (هوشمند) رسم گردیده و نتایج برای هر سه الگوریتم SRAIoT کاملاً یکسان است. در اینجا تنها به تحلیل نمودار الگوریتم SRIoT-۳ می‌پردازیم. همان گونه که در شکل ۴ مشاهده می‌شود، هنگامی که $\Delta$ برابر با سه است، الگوریتم دارای بیشترین سودمندی تجمعی است؛ بنابراین در تمام تنظیمات شبیه‌سازی اندازه $\Delta$ را برابر عدد سه در نظر گرفته و اندازه پارامتر دسته زمانی، مستقیماً از رابطه $T^{\frac{۱}{۳}}$ محاسبه می‌شود.




شکل ۶: معیار پشیمانی نرمال‌شده تمام الگوریتم‌ها در حالت مهاجم هوشمند.

جدول ۳: پارامترهای شبیه‌سازی.

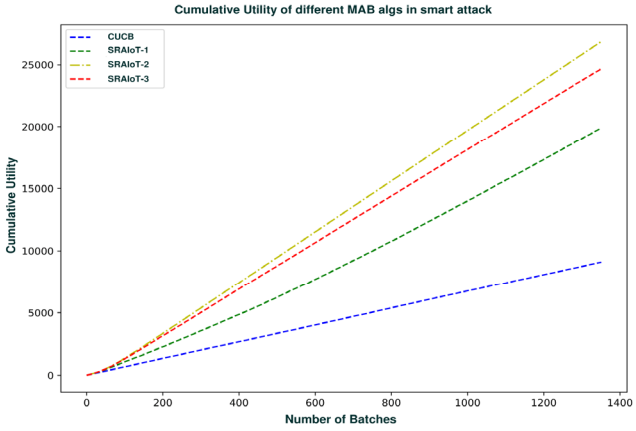
| پارامتر | مقدار | شرح |
|---|---|---|
| $N$ | ۱۵۰ | تعداد گره‌های شبکه اینترنت اشیا |
| $T$ | ۵۰۰۰۰ | زمان کل (آخرین برهه زمانی) |
| $\tau$ | ۳۷ | پارامتر تعیین اندازه دسته زمانی |
| $J$ | ۱۳۵۲ | تعداد دسته زمانی |
| $\ell$ | ۲ | تعداد منابع امنیتی |
| $M$ | ۲ | تعداد مهاجم |
| $r$ | ۰/۳ | پاداش |
| $c$ | ۰/۰۳ | هزینه |
| $\beta$ | ۰/۱ | پارامتر محاسبه امتیاز سرخوشه |
| $\gamma$ | ۰/۱ | پارامتر محاسبه احتمال راهبرد |
| $p_d$ | ۰/۹ | احتمال اولیه تشخیص حمله |

شکل ۵، تأثیر پارامتر $\gamma$ را بر روی سودمندی تجمعی در الگوریتم SRAIoT-۳ نمایش می‌دهد. این پارامتر میان اکتشاف و بهره‌برداری توازن برقرار می‌کند. همان گونه که در نمودار مشاهده می‌شود به ازای ۰/۱، بیشترین سودمندی تجمعی حاصل می‌شود. این حالت برای الگوریتم SRAIoT-۱ نیز به‌طور مشابه اتفاق می‌افتد. در الگوریتم SRAIoT-۲ این پارامتر حضور ندارد؛ اما همچنان موازنه میان اکتشاف و بهره‌برداری وجود دارد. به دلیل فرمول به‌روزرسانی وزن راهبرد در این الگوریتم، راهبردی که قبلاً انتخاب نشده، دارای بالاترین وزن بوده و این مهم توسط فرمول به‌روزرسانی وزن راهبرد الگوریتم نوع دوم- همان طور که در جدول ۲ آمده است- تضمین می‌شود.

پارامتر $\beta$ منحصراً در الگوریتم SRAIoT-۳ حضور داشته و بر روی سودمندی‌های تجمعی و معیار پشیمانی ضعیف تأثیر می‌گذارد. همچنین برای محاسبه امتیاز سرخوشه استفاده گردیده و به جبران امتیاز سرخوشه‌هایی می‌پردازد که بدون منبع امنیتی هستند. یعنی برای تمام سرخوشه‌های بدون منبع امنیتی، امتیاز سرخوشه به جای صفر، $\beta$ در نظر گرفته می‌شود. این مقدار با استناد به [۱۰] برابر با ۰/۱ در نظر گرفته شده است.

### ۴-۲-۲ معیار پشیمانی ضعیف

همان طور که قبل‌تر گفته شد، یکی از روش‌های سنجش و ارزیابی روش پیشنهادی، استفاده از معیار پشیمانی ضعیف است. هرچه این معیار به صفر همگرا شود، الگوریتم پیشنهادی کاراتر است. معیار پشیمانی نرمال‌شده به ازای تمام الگوریتم‌ها در حالت مهاجم هوشمند در شکل ۶ آمده است. همان طور که انتظار می‌رفت با افزایش افق زمانی $T$، معیار




شکل ۷: سودمندی تجمعی مهاجم سازگار برای الگوریتم‌های مختلف.

پشیمانی ضعیف برای الگوریتم پیشنهادی SRAIoT کاهش یافته و به صفر همگرا می‌شود. این حالت از تحلیل نظری اشاره‌شده در [۱۰]، پشتیبانی می‌کند؛ به‌طوری که معیار پشیمانی ضعیف نرمال‌شده به ازای $T \to \infty$ به عدد صفر همگرا می‌شود؛ اما معیار پشیمانی ضعیف برای کار مقایسه‌ای در طول زمان کاهش پیدا نمی‌کند.

### ۴-۲-۳ سودمندی تجمعی

شکل۷، برای الگوریتم‌های مختلف، سودمندی تجمعی مهاجم سازگار(هوشمند) را نمایش می‌دهد. در این حالت، الگوریتم SRAIoT-۲ از تمامی الگوریتم‌ها بهتر عمل کرده و روند افزایشی دارد. اگرچه نمودار الگوریتم CUCB، با توجه به ماهیت افزایشی سودمندی تجمعی، روندی رو به رشد دارد اما در مقایسه با دیگر الگوریتم‌ها درست عمل نکرده و مقدار سودمندی آن به مراتب کمتر است. در این حالت، تنظیمات مهاجم از نوع تخاصمی بوده و الگوریتم مقایسه‌ای از تمامی الگوریتم‌های پیشنهادی SRAIoT بدتر عمل می‌کند.

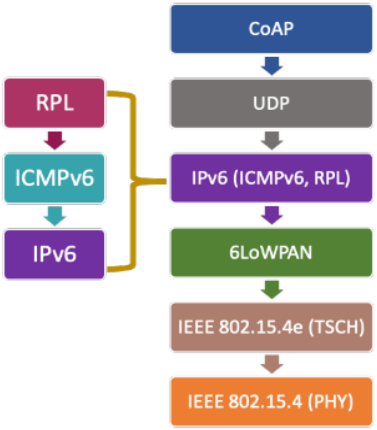
## ۵- شبیه‌سازی در بستر Cooja

نمونه واقعی یک حمله معمول برای ارزیابی روش پیشنهادی با استفاده از شبیه‌ساز Cooja پیاده‌سازی گردیده و سپس کارایی شبکه در حضور الگوریتم پیشنهادی و الگوریتم پایه مقایسه می‌شود. Contiki یک سیستم عامل متن‌باز برای شبکه‌های اینترنت اشیا بوده و روی میکروکنترلرهای کوچک کم‌مصرف اجرا می‌شود. این سیستم عامل شامل یک شبیه‌ساز حسگر به نام Cooja است و آخرین نسخه آن با نام Contiki-NG شناخته می‌شود. شبیه‌ساز Cooja، امکان شبیه‌سازی شبکه‌های ناهمگن را فراهم کرده و ابزاری ایده‌آل برای شبیه‌سازی شبکه‌های مبتنی بر RPL[1] است.

### ۵-۱ مسیریابی RPL، حملات رتبه‌ای و sinkhole

گره‌ها در اینترنت اشیا با محدودیت منابع از نظر انرژی، حافظه و قدرت پردازش محدود هستند و بنابراین نیاز به یک پشته پروتکل مناسب وجود دارد. بر طبق [۱۲]، پشته پروتکل مورد نظر صنعت که الزامات شبکه‌های IoT محدود با منابع را برآورده می‌کند، در شکل ۸ نشان داده شده است. پروتکل برنامه محدود (CoAP)[2] به‌عنوان یک پروتکل برای لایه

1. Routing Protocol for Low Power and Lossy Networks
2. Constrained Application Protocol

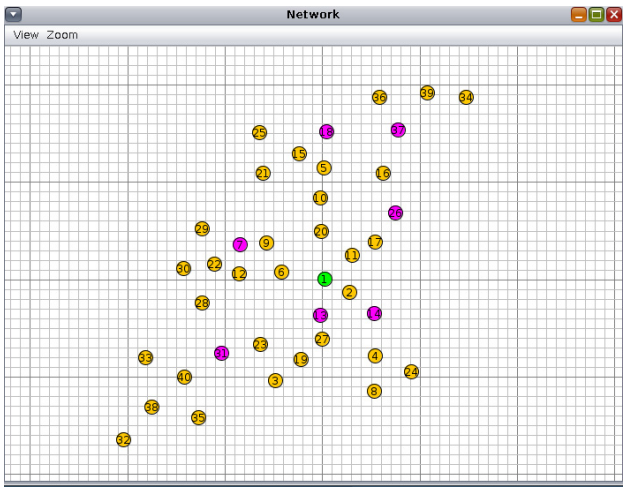



شکل ۸: پشته پروتکلی اینترنت اشیا [۱۲].

برنامه استفاده می‌شود که بر روی UDP[3]، پروتکل لایه انتقال مورد نظر اجرا می‌شود.

RPL یک گراف غیرمدور جهت‌دار مبتنی بر مقصد (DODAG)[4] را بین گره‌ها در شبکه ایجاد می‌کند. هر گره در یک DODAG دارای رتبه‌ای است که موقعیت یک گره را نسبت به گره‌های دیگر و با توجه به ریشه DODAG نشان می‌دهد. رتبه‌ها در جهت بالا به سمت ریشه DODAG کاهش یافته و از ریشه DODAG به سمت گره‌ها افزایش می‌یابند. به‌منظور حفظ توپولوژی مسیریابی و به‌روز نگه‌داشتن اطلاعات مسیریابی، RPL متشکل از چهار نوع پیام کنترلی شامل DIO[5]، DAO[6]، DIS[7] و DAO-ACK[8] است. حمله رتبه‌ای که در آن گره‌های مخرب به‌طور هدفمند بدترین والد موجود را انتخاب می‌کنند و DIO خود را به‌روز نمی‌کنند تا ترافیک عبوری را با تأخیر مواجه کنند. برای دورزدن اعتبارسنجی رتبه توسط والدین، گره‌های مخرب از ارسال پیام‌های DAO خودداری می‌کنند؛ یعنی هیچ مسیر نزولی به گره در معرض خطر و فرزندان آن وجود ندارد. یک گره در حمله Sinkhole، مسیر مسیریابی بهتری را برای همسایگان خود تبلیغ می‌کند تا ترافیک بیشتری را برای استراق سمع جذب کند. این حالت در RPL با تبلیغ رتبه ریشه در پیام‌های DIO به‌دست می‌آید [۱۳]. این حمله به خودی خود لزوماً عملکرد شبکه را مختل نمی‌کند؛ اما هنگامی که با حمله دیگری همراه شود می‌تواند بسیار قدرتمند عمل کند. در این مقاله به‌منظور شبیه‌سازی حمله در کنار حمله Sinkhole از حمله رتبه‌ای نیز استفاده شده است؛ به قسمی که گره مخرب به‌منظور جذب فرزندان، رتبه خودش را کاهش می‌دهد تا بقیه گره‌های فرزند جذب این مسیر شوند. سپس شروع به دورانداختن بسته‌های دریافتی‌اش کرده و به گره والد خود تحویل نمی‌دهد. این امر باعث تأخیر در شبکه و پایین‌آمدن نرخ بسته تحویلی می‌شود. برای اجرای شبکه اینترنت اشیا در بستر شبیه‌سازی از مثال معروف rpl-udp در Cooja استفاده می‌گردد. گره‌های استفاده‌شده در این شبیه‌سازی از نوع Sky mote هستند. جهت اجرای اسکریپت شبیه‌سازی حمله، باید یک سری تغییرات در کدهای سیستمی Cooja اعمال شود که این کدهای سیستمی، مختص مثال کاربردی rpl-udp

3. User Datagram Protocol
4. Destination-Oriented Directed Acyclic Graph
5. DODAG Information Objects
6. Destination Advertisement Object
7. DODAG Information Solicitation
8. DAO Acknowledgment




شکل ۹: مثالی از شبکه اینترنت اشیا با ۴۰ گره.

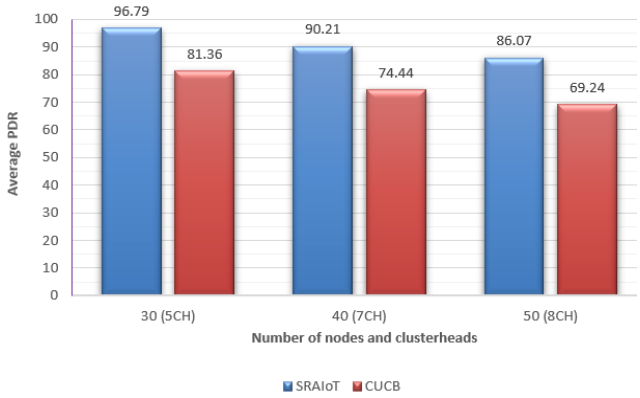
هستند. کدهای سیستمی در Cooja با استفاده از مجموعه کامپایلرها و کتابخانه‌های GCC قابل تغییر بوده و اجرا گرفته می‌شود. برای اعمال برخی از تغییرات از توضیحات [۱۳] استفاده شده است. به‌طور ساده، هر مثال rpl-udp از یک سری گره‌های udp-client و گره‌های udp-server تشکیل می‌شود که هدف گره‌های udp-client، ارسال داده از طریق والد خود به ریشه یعنی udp-client است.

## ۵-۲ آزمایش شبکه اینترنت اشیا

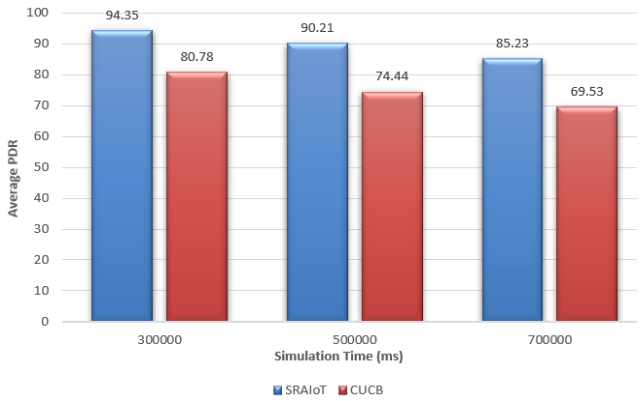
در این بخش در سه آزمایش، نمونه‌هایی از شبکه اینترنت اشیا را مورد بررسی قرار خواهیم داد. هدف از شبیه‌سازی حمله، تحت تأثیر قراردادن نرخ تحویل بسته‌ها و مقایسه الگوریتم پیشنهادی ما با الگوریتم CUCB است. طی اجرای این دو الگوریتم و پس از آنکه شبکه به یک سری خوشه تقسیم شد، باید الگوریتم برای تخصیص منبع امنیتی بر روی سرخوشه‌ها تصمیم بگیرد. هر کدام از الگوریتم‌ها، تعدادی از سرخوشه‌ها را برای محافظت انتخاب می‌کنند و بقیه سرخوشه‌ها و به تبع، اعضای خوشه‌های مد نظر بدون محافظت رها می‌شوند. تمام این آزمایش‌ها از طریق ماشین مجازی و بر روی اوبونتو ۲۰/۰۴ اجرا شده است. شکل ۹ مثالی از نمای گره‌ها را در شبیه‌ساز نشان می‌دهد. در این مثال، ۴۰ گره اینترنت اشیا با هفت سرخوشه حضور دارند.

### ۵-۲-۱ بررسی تأثیر تعداد گره‌ها بر PDR

طی این آزمایش، سه منبع امنیتی در اختیار داشته و دفعات آزمایش به ازای ۳۰، ۴۰ و ۵۰ گره تکرار شده است. دو مهاجم به گره‌های شبکه حمله کرده و عملکرد آن را مختل می‌کنند. این شبیه‌سازی به ازای ۵۰۰۰۰۰ میلی ثانیه اجرا می‌شود. از آنجا که الگوریتم پیشنهادی در یک محیط تخاصمی به‌صورت هوشمند عمل می‌کند. در طی زمان، الگوریتم ما با پیش‌بینی رفتار طرف مقابل به نسبت الگوریتم کار مقایسه‌ای، رفتار معقولی دارد و منابع امنیتی را بر روی سرخوشه‌هایی قرار می‌دهد که احتمال حمله آنها بالاتر است. بنابراین هنگامی که مهاجم به سرخوشه مد نظرش، حمله و عملکرد شبکه را مختل می‌کند، چون سرخوشه مد نظر دارای منبع امنیتی است، با احتمال بالایی، حمله تشخیص داده شده و شبکه به کار خودش ادامه می‌دهد. با افزایش تعداد گره، تعداد سرخوشه‌ها نیز بیشتر می‌شوند. با اعمال فرض تعداد ثابت منابع امنیتی برای تخصیص در سرخوشه‌ها، نرخ تحویل بسته سیر نزولی خواهد داشت؛ اما طبق آزمایشی که انجام شد، رفتار الگوریتم پیشنهادی نسبت به الگوریتم پایه، معقول بوده و سرخوشه‌های مناسب‌تری را برای حفاظت انتخاب می‌کند و




شکل ۱۰: تأثیر تعداد گره بر روی PDR.




شکل ۱۱: بررسی تأثیر اندازه دسته زمانی بر PDR.

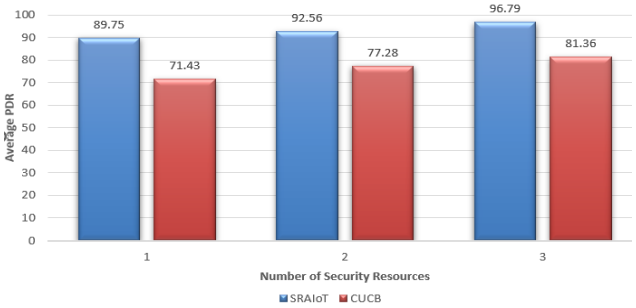
نهایتاً حمله ناموفق می‌شود. به عنوان مثال، مطابق شکل ۱۰، به ازای ۴۰ گره و هفت سرخوشه، PDR الگوریتم مقایسه‌ای نسبت به الگوریتم‌های پیشنهادی کمتر است.

### ۵-۲-۲ بررسی تأثیر تعداد پارامتر اندازه دسته زمانی

هدف از این آزمایش، بررسی و ارزیابی تأثیر پارامتر اندازه دسته زمانی بر نرخ تحویل بسته (PDR) است. مدافع در طول بازه یک دسته زمانی، راهبرد یکسانی را برای تخصیص منابع امنیتی اتخاذ می‌کند. در این آزمایش، چهل گره، هفت سرخوشه، سه منبع امنیتی و دو مهاجم مفروض است. ابرپارامتر مؤثر بر روی اندازه دسته‌های زمانی، پارامتر $\tau$ می‌باشد و زمان اجرای شبیه‌سازی در ۳۰۰۰۰۰، ۵۰۰۰۰۰ و ۷۰۰۰۰۰ میلی‌ثانیه تنظیم شده است. نهایتاً در دسته‌های زمانی مذکور، اسکریپت حمله اجرا می‌شود. در اجرای هر دوی الگوریتم‌ها، یک سری گره‌ها دچار حمله شد و نهایتاً عملکرد شبکه با اختلال روبه‌رو گردید. خروجی آزمایش با سه تنظیم مختلف برای الگوریتم پیشنهادی و الگوریتم مقایسه‌ای در شکل ۱۱ آمده است. هرچه بازه زمانی بیشتر باشد، PDR پایین‌تر بوده و برای بازه زمانی کمتر، PDR بالاتر می‌رود. زمانی که بازه تصمیم به نسبت پایین‌تر است، عامل هوشمند سریع‌تر واکنش نشان داده و به تبع، هزینه جابه‌جایی منابع هم بالا می‌رود. اگرچه در بازه زمانی کمتر، PDR بهتری نصیب شبکه می‌شود، اما برقراری توازن میان پاداش و هزینه جابه‌جایی از اهمیت بالایی برخوردار است. بنابراین این مقدار طبق شکل ۴ بر روی $T^{\frac{1}{3}}$ تنظیم شد تا الگوریتم پیشنهادی با پیش‌بینی رفتار طرف مقابل، رفتار معقول‌تری نسبت به الگوریتم پایه داشته باشد.

### ۵-۲-۳ بررسی تأثیر تعداد منابع امنیتی

هدف از این آزمایش، بررسی تأثیر تعداد منابع امنیتی جهت حفاظت از




شکل ۱۲: تأثیر تعداد منابع امنیتی بر PDR.

سرخوشه‌ها در مقابل حمله مهاجمین است. بدین منظور در یک توپولوژی با سی گره و پنج سرخوشه، آزمایش را ۵۰۰۰۰۰ میلی‌ثانیه انجام دادیم. در ابتدای کار و برای پنج سرخوشه، تنها یک منبع امنیتی حضور داشت. الگوریتم پیشنهادی، گره مناسب‌تری را نسبت به الگوریتم پایه، جهت استقرار منبع امنیتی در نظر گرفته و در نتیجه، نرخ تحویل بسته آن بالاتر است. با افزایش تعداد منابع امنیتی از یک به سه عدد برای پنج جایگاه، سرخوشه‌های بیشتری مورد حفاظت قرار گرفته و نرخ تحویل بسته (PDR) بالاتر می‌رود. نهایتاً با تخصیص سه منبع امنیتی بر روی پنج سرخوشه، مطابق شکل ۱۲ نرخ تحویل الگوریتم پیشنهادی ۹۶/۷۹ است. تعداد مهاجمین به شبکه در این آزمایش، دو عدد می‌باشد.

## ٦- نتیجه‌گیری

ما در این مقاله، مسئله تخصیص منابع امنیتی را برای مقابله با حملات در اینترنت اشیا با استفاده از رویکرد یادگیری برخط مطرح کردیم و با توجه به شرایط محیط از رویکرد چارچوب CMAB تخاصمی با لحاظ هزینه جابه‌جایی استفاده کردیم. برخلاف کارهای پیشین در مدل‌سازی پیشنهادی فرض می‌شود که طرفین (اعم از مدافع و مهاجم) از تابع هدف رقیب و شرایط وی مطلع نیستند و تخصیص منابع توسط مدافع شبکه، صرفاً با انباشت تجربه و یادگیری تدریجی استراتژی تدافعی انجام می‌شود. نوآوری دیگر راهکار پیشنهادی، ، لحاظ‌کردن هزینه جابه‌جایی منابع از یک سرخوشه به سرخوشه دیگر به‌عنوان یک معیار جریمه در کارایی فرایند یادگیری است. از آنجا که نصب و جای‌گذاری منابع امنیتی دارای هزینه است، در این پژوهش برآنیم که با حداقل نصب/ حذف‌ها و هزینه پرسنل مدیریتی از شبکه اینترنت اشیا در مقابل حملات محافظت کنیم. روش پیشنهادی به‌عنوان یک مزیت می‌تواند به‌صورت پارامتریک، موازنه‌ای میان کارآمدی مقابله با حملات و سربار جابه‌جایی‌ها ایجاد کند. شایان ذکر است که مهاجم هوشمند دقیقاً از الگوریتم مدافع استفاده می‌کند. همچنین یک سناریوی واقعی در بستر Cooja شبیه‌سازی شده و کارایی شبکه پس از تخصیص منابع بعد از حمله سنجیده می‌شود. ما نتایج شبیه‌سازی و پیاده‌سازی را در قالب نمودارهایی نشان دادیم. این نمودارها، تأثیر موارد مختلفی اعم از تغییر در تنظیمات شبکه، مهاجم، مدافع، منابع امنیتی و ابرپارامترهای الگوریتم را بر روی کارکرد الگوریتم پیشنهادی در مقابل الگوریتم پایه می‌سنجند. نهایتاً به‌عنوان یک معیار نظری از معیار پشیمانی ضعیف برای سنجش عملکرد و کارایی الگوریتم پیشنهادی خود استفاده کردیم. به‌عنوان نتیجه مشاهده کردیم که با درنظرگرفتن افق زمانی برابر با ۵۰۰۰۰، معیار پشیمانی ضعیف الگوریتم پیشنهادی به سمت صفر همگرا می‌شود. همچنین به‌عنوان کارهای آینده می‌توان به ارائه الگوریتم توزیع‌شده به جای متمرکز، تعمیم روش به حالت چندمنبعی و تعمیم روش به حالت چند نوع حمله اشاره کرد.

## مراجع


[1] A. H. Anwar, C. Kamhoua, and N. Leslie, "Honeypot allocation over attack graphs in cyber deception games," in *Proc. IEEE Int. Conf. on Computing, Networking and Communications, ICNC'20*, pp. 502-506, Big Island, HI, USA, 17-20 Feb. 2020.

[2] L. Chen, Z. Wang, F. Li, Y. Guo, and K. Geng, "A stackelberg security game for adversarial outbreak detection in the Internet of Things," *Sensors*, vol. 20, no. 3, Article ID: 804, Feb. 2020.

[3] A. H. Anwar, C. Kamhoua, and N. Leslie, "A game-theoretic framework for dynamic cyber deception in internet of battlefield things," in *Proc. of the 16th EAI Int. Conf. on Mobile and Ubiquitous Systems: Computing, Networking and Services*, pp. 522-526, Houston, TX, USA, 12-14 Nov. 2019.

[4] A. Rullo, E. Serra, E. Bertino, and J. Lobo, "Optimal placement of security resources for the Internet of Things," *The Internet of Things for Smart Urban Ecosystems*, pp. 95-124, Jan. 2019.

[5] A. Rullo, D. Midi, E. Serra, and E. Bertino, "Pareto optimal security resource allocation for Internet of Things," *ACM Trans. on Privacy and Security*, vol. 20, no. 4, pp. 1-30, Nov. 2017.

[6] M. Zhu, *et al.*, "A survey of defensive deception: approaches using game theory and machine learning," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 4, pp. 2460-2493, Aug. 2021.

[7] A. Rullo, D. Midi, E. Serra, and E. Bertino, "A game of things: strategic allocation of security resources for IoT," in *Proc. IEEE/ACM 2nd Int. Conf. on Internet-of-Things Design and Implementation, IoTDI'17*, pp. 185-190, Pittsburgh, PA, USA, 18-21 Apr. 2017.

[8] M. A. R. Al Amin, S. Shetty, L. Njilla, D. K. Tosh, and C. Kamhoua, "Online cyber deception system using partially observable Monte Carlo planning framework," in *Proc. Int. Conf. on Security and Privacy in Communication Systems*, vol. 2, pp. 205-223, Orlando, FL, USA, 23-25 Oct. 2019.

[9] S. Wang, Q. Pei, J. Wang, G. Tang, Y. Zhang, and X. Liu, "An intelligent deployment policy for deception resources based on reinforcement learning," *IEEE Access*, vol. 8, pp. 35792-35804, 2020.

[10] M. Li, D. Yang, J. Lin, and J. Tang, "Specwatch: a framework for adversarial spectrum monitoring with unknown statistics," *Computer Networks*, vol. 143, pp. 176-190, Oct. 2018.

[11] W. Chen, Y. Wang, and Y. Yuan, "Combinatorial multi-armed bandit: general framework and applications," *Proceedings of Machine Learning Research*, vol. 28, no. 1, pp. 151-159, Feb. 2013.

[12] M. R. Palattella, N. Accettura, X. Vilajosana, T. Watteyne, L. A. Grieco, G. Boggia, and M. Dohler, "Standardized protocol stack for the internet of (important) things," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1389-1406, Dec. 2012.

[13] F. Algahtani, T. Tryfonas, and G. Oikonomou, "A reference implemenation for RPL attacks using contiki-NG and Cooja," in *Proc. 17th Int. Conf. on Distributed Computing in Sensor Systems, DCOSS'21*, pp. 280-286, Pafos, Cyprus, 14-16 Jul. 2021.



**نسیم نوائی** تحصیلات خود را در مقطع کارشناسی مهندسی فناوری اطلاعات در سال ۱۳۹۶ در دانشگاه تبریز به پایان رسانده است. ایشان مدرک کارشناسی ارشد خود را در رشته مهندسی کامپیوتر- شبکه‌های کامپیوتری در سال ۱۴۰۱ از دانشگاه علم و صنعت ایران دریافت نموده است. زمینه‌های تحقیقاتی مورد علاقه ایشان عبارتند از: شبکه‌های کامپیوتری، ارتباطات بی‌سیم، اینترنت اشیا و یادگیری ماشین.

**وصال حکمی** در سال ۱۳۸۳ مدرک کارشناسی مهندسی کامپیوتر- نرم‌افزار خود را از دانشگاه صنعتی امیرکبیر و مدارک کارشناسی ارشد و دکتری خود را در رشته مهندسی فناوری اطلاعات- شبکه‌های کامپیوتری از همان دانشگاه به ترتیب در سال‌های ۱۳۸۷ و ۱۳۹۴ دریافت نموده است. دکتر حکمی از سال ۱۳۹۵ به عنوان عضو هیأت علمی در دانشکده مهندسی کامپیوتر دانشگاه علم و صنعت ایران مشغول فعالیت‌های آموزشی و پژوهشی بوده و ضمنا یکی از اعضای قطب علمی شبکه‌های کامپیوتری وزارت علوم است. نام‌برده قبل از پیوستنش به دانشگاه علم و صنعت ایران طی سال ۱۳۹۴ به عنوان مشاور در حوزه استانداردسازی نسل پنجم شبکه‌های مخابراتی بی‌سیم در پژوهشگاه ارتباطات و فناوری اطلاعات فعالیت داشته است. زمینه‌های تحقیقاتی مورد علاقه ایشان عبارتند از: شبکه‌های کامپیوتری، ارتباطات بی‌سیم، بهینه سازی ریاضی، نظریه بازی‌ها و یادگیری تقویتی.