

Winter 2024, 5 (4), 111-122
DOR:

Received: 15 Oct 2024
Accepted: 28 Nov 2024

مقاله پژوهشی

Enhanced RPL Routing Protocol in Underwater Internet of Things using Efficient Network Clustering

Hamed Karimi¹, Keyhan Khamforoosh^{2*}, Vafa Maihami³

1. Ph.D. Student, Department of Computer Engineering, Sanandaj Branch, Islamic Azad University, Sanandaj, Iran.
2. Assistant Professor, Department of Computer Engineering, Sanandaj Branch, Islamic Azad University, Sanandaj, Iran. *Corresponding Author, k.khamforoosh@iausdj.ac.ir
3. Assistant Professor, Department of Computer Engineering, Sanandaj Branch, Islamic Azad University, Sanandaj, Iran

Abstract

Introduction: RPL routing protocol is one of the most important routing protocols in Internet of Things (IoT) and Underwater Internet of Things (UIoT). In this routing protocol, the energy consumption of the nodes and the number of lost packets due to the shutdown of the nodes has created great challenges.

Method: In the new scenario named CB-RPL, by applying a data transfer adjustment mechanism based on the HCEH-UC algorithm and the Bloom filter data structure on each of the created clusters, an attempt is made to improve the RPL routing protocol for the following use. It becomes water. The simulation is done by using the NS3.25 simulator.

Results: In the results, it can be seen that the sending time is reduced, the energy consumption is reduced and the load is balanced. The parameter of destroyed nodes, the average number of access to routing table entries, the number of packet sent to the root, the remaining energy and the lost packets have also improved significantly.

Discussion: The proposed algorithm with an adaptive clustering based on depth by keeping DIO and DAO packets in the RPL algorithm performs routing with a smaller number of packets. To manage cluster head energy like all clustering algorithms in different rounds, the role of cluster heads is distributed. Modes have been added to RPL, including energy harvesting (EH) (sleep mode) and data transmission (operating mode), and a routing mechanism for underwater Internet on this platform is proposed. Also, the Bloom filter structure is used to reduce the memory size of intermediate nodes in routing operations.

Keywords: UIoT, Clustering, Routing protocol, Bloom filter, HCEH-UC.

مسیریابی ارتقایافته RPL در اینترنت اشیاء زیر آب با استفاده از خوشه‌بندی کارآمد شبکه

دوره پنجم، زمستان ۱۴۰۳
شماره چهارم، صص: ۱۱۱-۱۲۲

تاریخ دریافت: ۱۴۰۳/۰۷/۲۴
تاریخ پذیرش: ۱۴۰۳/۰۹/۰۸

حامد کریمی^۱، کیهان خام‌فروش^{۲*}، وفا میهمی^۳

۱- دانشجوی دکترا، گروه مهندسی کامپیوتر، واحد سنندج، دانشگاه آزاد اسلامی، سنندج، ایران. h.karimi@iausdj.ac.ir

۲- استادیار، گروه مهندسی کامپیوتر، واحد سنندج، دانشگاه آزاد اسلامی، سنندج، ایران. (نویسنده مسئول) k.khamforoosh@iausdj.ac.ir

۳- استادیار، گروه مهندسی کامپیوتر، واحد سنندج، دانشگاه آزاد اسلامی، سنندج، ایران. maihami@iausdj.ac.ir

چکیده: پروتکل مسیریابی RPL، یکی از مهم‌ترین پروتکل‌های مسیریابی در اینترنت اشیاء (IoT) و اینترنت اشیاء زیر آب (UIoT) است. در این پروتکل مسیریابی، مصرف انرژی گره‌ها و میزان گم‌شدن بسته‌های ارسالی به دلیل خاموش شدن گره‌ها چالش‌های بزرگی را ایجاد کرده است. برای این منظور در سناریوی جدید که با CB-RPL نام‌گذاری می‌شود با اعمال یک مکانیسم تنظیم انتقال داده براساس الگوریتم HCEH-UC و ساختار داده بلوم فیلتر روی هر کدام از خوشه‌های ایجاد شده، سعی در بهبود پروتکل مسیریابی RPL برای استفاده زیر آب می‌شود. الگوریتم پیشنهادی با یک خوشه‌بندی تطبیقی بر اساس عمق با حفظ بسته‌های DIO و DAO در الگوریتم RPL، مسیریابی با تعداد بسته‌های کمتر را انجام می‌دهد. برای مدیریت انرژی سرخوشه مانند همه الگوریتم‌های خوشه‌بندی در دوره‌های مختلف نقش سرخوشه‌ها به صورت توزیع شده عوض می‌شود. حالت‌هایی به RPL، از جمله برداشت انرژی (EH) (حالت خواب) و انتقال داده (حالت عملیات)، اضافه شده و یک مکانیسم مسیریابی برای اینترنت زیر آب روی این بستر پیشنهاد شده است. همچنین ساختار بلوم فیلتر برای کاهش حجم حافظه نودهای میانی در عملیات مسیریابی به کار گرفته می‌شود. در نتایج شبیه‌سازی که با استفاده از شبیه‌ساز NS3.25 صورت گرفته، کاهش زمان ارسال، کاهش مصرف انرژی و ایجاد تعادل بار مشاهده می‌شود. پارامتر نودهای از بین رفته، متوسط تعداد دسترسی‌ها به ورودی‌های جدول مسیریابی، تعداد بسته‌های ارسالی به ریشه، انرژی باقیمانده و بسته‌های گم‌شده هم بهبود قابل ملاحظه‌ای داشته است.

واژه‌های کلیدی: شبکه زیر آب، خوشه‌بندی، پروتکل‌های مسیریابی، بلوم فیلتر، HCEH-UC.

۱. مقدمه

از سال ۲۰۱۵ به بعد شبکه‌های زیر آب کاربردهای متعددی پیدا کرده و تاکنون نیز تحقیقات گسترده‌ای در این حوزه مانند نظارت‌های دریایی و نظارت بر کیفیت آب مطرح شده است. در سناریوهای جدید اینترنت اشیا زیر آب گره‌های مختلف از طریق سیگنال صوتی با یکدیگر ارتباط برقرار می‌کنند. در برخی از این سناریوها فقط یک گره سینک در سطح آب قرار دارد و در برخی دیگر چندین گره سینک در سطح آب مسئول رساندن پیام به گره‌های عادی واقع در زیر آب هستند. گره سینک، نقش ناظر بین گره‌های حسگر عادی را دارند، به این معنی که یک گره حسگر، زمانی که پیامی را ارسال می‌کند می‌بایست یک بازخورد نیز به سمت گره سینک بفرستد. علاوه بر آن در سناریوهای دیگر، گره‌های سینک در زیر آب نیز قرار داشته و بعد از دسته‌بندی پیام‌ها آن‌ها به سمت نزدیک‌ترین ایستگاه کاری می‌فرستد. مسأله و چالش اساسی در اینترنت اشیا زیر آب این است که نودهای زیر آب بسیار به هم نزدیکند و چگالی بالایی دارند. عملاً چگالی بالا و اینکه هرچه مسیریابی از سطح آب به طرف عمق پیش می‌رود پارامترهایی همچون فشار، افزایش و دمای محیط کاهش می‌یابد یک چالش اساسی در مسیریابی است. استفاده از آکوستیک نفوذپذیری و برد بیشتر را در آب اعمال می‌کند اما چالش‌های تأخیر و تداخل و نویز در آب در لایه فیزیکی همچنان باقی است. بنابراین نمود این چالش‌ها در لایه سوم مصرف بالای انرژی گره‌ها، میزان بالای گم شدن و دراپ شدن بسته‌های مسیریابی و عدم توازن بار در لینک‌های شبکه است که در محیط آب اهمیت بیشتری پیدامی‌کند. استفاده از امواجی که از سطح آب به سمت عمق منتقل می‌شود نیز نمود دیگر چالش‌های لایه اول و دوم به مسیریابی لایه سوم است که در عمق مسیر ویژگی‌های آن نیز عوض می‌شود. چون سیگنال‌های استفاده شده در این شبکه‌ها به جای الکترومغناطیسی عمدتاً آکوستیک هستند و گره‌ها نسبت به هم فاصله نزدیکی دارند به همین دلیل باید از سیگنال‌هایی استفاده کرد که دارای فرکانس پایین‌تری باشند [1-3].

روش‌های مسیریابی معمول روی آب رویکرد متفاوتی با روش‌های زیر آب دارند [4]. ترافیک شبکه از سطح به عمق آب ایجاد می‌شود. شروع مسیریابی از سطح به طرف عمق در این الگوریتم‌ها یک الزام است چرا که مسأله تشخیص گره‌های همسایه در این فاز انجام می‌شود. در فاز ایجاد مسیر، نودهایی که حاوی اطلاعات سینک هستند مسیری را رو به سمت سطح دریا بازسازی می‌کنند. برای ایجاد این مسیر، نودها نیازمند نگهداشتن اطلاعاتی در مورد نودهای همسایه توسط گره‌های میانی هستند تا در مسیر برگشت اطلاعات به نودهای سطح آب، مسیرها همچنان مشخص باشد. لذا این موضوع بسیار هزینه‌بر و چالشی برای این شبکه‌ها است. اگر بتوان با توجه به وابستگی زمانی و مکانی اطلاعاتی که از نودهای مجاور هم جمع‌آوری می‌شود تنها تعدادی از نودهای مجاور را در مسیریابی شرکت داد در کاهش هزینه‌های مسیریابی بهبود زیادی ایجاد می‌شود.

در میان پروتکل‌های مسیریابی مختلف، یکی از مهم‌ترین پروتکل‌های مسیریابی در IoT، پروتکل مسیریابی RPL^۱ است. این پروتکل می‌تواند یک پروتکل مسیریابی امیدوارکننده برای اینترنت زیر آب باشد که می‌توان با تغییراتی آن را برای مسیریابی زیر آب بهینه کرد. این پروتکل با آر سال بسته‌های DIO از سینک روی آب به طرف نودهای زیر آب، فاز معرفی همسایه‌ها را به خوبی انجام می‌دهد و با آر سال بسته‌های DAO مسیر رو به بالا را بازسازی می‌کند. مشکل این الگوریتم آن است که در اعماق آب نودهای ارسال‌کننده اطلاعات با دمای کمتر و فشار بیشتر مواجهند، لذا نرخ خطا بالا رفته و انرژی بیشتری از نودها مصرف می‌شود. این باعث می‌شود که داده‌های غلط ایجاد و خیلی سریع پوشش نودهای اعماق آب از دست برود. علاوه بر آن نودها مجبور به نگهداری جداول مسیریابی بزرگی خواهند بود که مسیریابی را به شدت کند می‌کند. چسبجی جداول بزرگ مسیریابی یکی دیگر از چالش‌هایی است که در این مقاله سعی در حل آن است. همان‌طور که مطرح شد داده‌ها توسط گره‌های حسگر به سمت ایستگاه کاری ارسال می‌شوند. هر چقدر فاصله گره حسگر از ایستگاه کاری بیشتر باشد مصرف انرژی گره نیز افزایش می‌یابد و هم چنین احتمال گم شدن بسته‌های مسیریابی نیز افزایش می‌یابد. در صورتی که بسته گره گم شود، گره مجدداً آن را ارسال می‌کند و احتمال کاهش کیفیت گره سینک و تأخیر در صف به دلیل پهنای باند ناکافی وجود دارد. در صورتی که انرژی گره حسگر به پایان برسد باعث ایجاد حفره پوششی می‌شود. در این مقاله برای غلبه بر چالش‌های ذکر شده از الگوریتم‌های خوشه‌بندی استفاده شده است. اطلاعات هر گره یا ترمینال برای اینترنت اشیا ضروری است. مرگ گره‌ها از یک منطقه ممکن است منجر به از دست دادن کنترل کامل یک نوع تجهیزات شود، بنابراین، شبکه تطبیقی که اندازه خوشه بر اساس عمق تغییر می‌کند می‌تواند برای عملیات در اینترنت اشیا زیر آب مناسب‌تر باشد. در این مقاله، یک پروتکل مسیریابی مبتنی بر سلسله مراتب خوشه‌بندی تطبیقی^۲ برای اینترنت اشیا زیر آب با برداشت انرژی پیشنهاد شده است. HCEH-UC پیشنهادی از سرخوشه‌های (CH) توزیع شده جهت توازن مصرف انرژی بین گره‌ها استفاده می‌کند. همچنین برای بهبود حافظه که در نهایت به بهبود مصرف انرژی گره‌ها نیز می‌انجامد در پروتکل مسیریابی RPL از ساختار داده‌ای به نام بلوم فیلتر استفاده می‌کند که در شبیه‌ساز NS-3 پیاده‌سازی شده است. یکی از راه حل‌های تأثیرگذار برای بهبود حافظه گره‌ها از طریق کم کردن تعداد دسترسی به مدخل‌های جدول مسیریابی، استفاده از ساختار داده بلوم فیلتر است. این روش قبل از اینکه گره مورد نظر را در جدول مسیریابی جستجو کند، لینک مقصد را به بلوم فیلتر داده و در زمان $O(k)$ تعداد کلیدهای جستجو در آرایه بلوم فیلتر است) مشخص کرده که آیا گره مورد نظر در جدول مسیریابی به عنوان گره مقصد وجود دارد یا خیر. این راه حل می‌تواند به گره‌های موجود در عمق آب کمک کند که

جداول مسیر یابی کوچتری راجستجو کنند و در نتیجه عمر طولانی تری داشته باشند.

نوآوری های زیر در این مقاله ارائه شده است:

۱. این الگوریتم با یک خوشه بندی تطبیقی بر اساس عمق با حفظ بسته های DIO و DAO در الگوریتم RPL، مسیریابی با تعداد بسته های کمتری را انجام می دهد. همچنین تعداد نودهای کمتری در مسیر یابی شرکت خواهند کرد. برای مدیریت انرژی سرخوشه مانند همه الگوریتم های خوشه بندی در دورهای مختلف نقش سرخوشه ها به صورت توزیع شده عوض می شود [20-17]. مزیت خوشه بندی این است که با اعمال شرایط متفاوت مسیریابی در هر خوشه، عملاً برای هر خوشه بر اساس شرایط مسیریابی انجام می شود.

۲. استفاده از بلوم فیلتر باعث بهبود جستجوی حافظه و تسریع در جریان مسیریابی به خصوص در هنگام ارسال بسته های DAO در پروتکل مسیریابی RPL می شود. (البته بلوم فیلتر روی بسته های DIO و DIS نیز اعمال شده است). این مسأله باعث کاهش زمان جستجو در نودهای موجود در خوشه های اعماق آب می شود.

۳. یک مکانیسم تنظیم انتقال داده برای RPL پیشنهاد شده است که بسیار شبیه پروتکل مسیریابی (HCEH-UC) است. حالت هایی در RPL، از جمله برداشت انرژی (EH) (حالت خواب) و انتقال داده (حالت عملیات)، اضافه شده است. بنابراین، یک مکانیسم مسیریابی متناظر برای اینترنت زیر آب برای حالت خواب و حالت عملیات پیشنهاد شده است که در اعماق درصد بیشتری را از نودها را درگیر فرایند خواب و بیداری می کند. در RPL پیشنهادی به بسته های DIS که مسئول انتشار خرابی مسیر هستند یک فیلد جدید اضافه می شود که گره هایی که در حالت خواب و گره هایی که در حالت انتقال داده هستند را از هم جدایی کند و اطلاعات آن را به گره های همسایه و گره والد ترجیحی ارسال می کند.

۲. کارهای مرتبط

محققان در حوزه اینترنت اشیا زیر آب به بررسی و پیاده سازی شهرهای هوشمند می پردازند که این موضوع به یک چارچوب مهم برای تحقیقات تبدیل شده است. هدف اصلی آن ها ایجاد رضایت شهروندان از پروژه های شهرهای هوشمند است. موضوع مسیریابی در اینترنت زیر آب و اجرای شهرهای هوشمند، اعمالی همچون پیاده سازی بنادر هوشمند و کاربردهای زیرآب آن ها را به یک چالش جدید تبدیل کرده است. پروتکل مسیریابی RPL یکی از مهم ترین پروتکل های مورد استفاده در حوزه اینترنت اشیا است و اخیراً برای حوزه زیرآب نیز گسترش پیدا کرده است. تحقیقات در این زمینه می تواند به بهبود اجرای پروژه های شهرهای هوشمند و کاربردهای زیرآب آن ها کمک کند. استفاده از پروتکل های مسیریابی مناسب می تواند به بهینه سازی ارتباطات دستگاه های مختلف در این شبکه ها و افزایش کارایی کلی آن ها کمک کند [5,6].

در تحقیقات، سه سناریو مختلف با ایستگاه کارهای متفاوت برای اینترنت زیر آب مدنظر قرار گرفته است. در شبیه سازی نخست، گره سینک به طور مجزا در سطح آب قرار می گیرد. در شبیه سازی دوم، چندین گره سینک واقع شده اند که هر کدام از آن ها با فاصله یکسان در سطح آب قرار دارند. در محیط سوم، گره های سینک با فاصله مساوی در سطح آب قرار دارند و برخی گره ها نیز در زیر آب واقع شده و فاصله آن ها به صورت تصادفی با یکدیگر ایجاد شده است.

در روش پیشنهادی این مقاله، از هر سه سناریو در شبیه سازی استفاده شده است و خوشه بندی و بلوم فیلتر جهت بهبود مصرف انرژی گره ها در RPL به کار رفته است. این اقدامات نشان می دهد که تحقیقات در این حوزه به منظور بهبود عملکرد و کارایی شبکه های اینترنت زیر آب و استفاده بهینه از انرژی، از اهمیت ویژه ای برخوردار است. این انواع شبیه سازی و استفاده از ابزارهای بهینه سازی می تواند به توسعه و پیشرفت تکنولوژی در حوزه اینترنت زیر آب کمک کند و در نهایت منجر به ارتقاء و بهبود کیفیت زندگی در شهرهای هوشمند شود.

در شبیه سازی صورت گرفته، گره های ایستگاه کاری (گره های چاهک) به صورت تصادفی در محیط پخش شده به طوری که سرخوشه های هر گروه با یکدیگر ارتباط برقرار می کنند. انتخاب CH تقریباً در تمام پروتکل های مسیریابی مبتنی بر خوشه بندی یک کار پرچالش است زیرا مستقیماً بر عملکرد شبکه تأثیر می گذارد. ایده آل ترین گره باید به عنوان CH انتخاب شود تا طول عمر شبکه افزایش یابد.

یکی از مهم ترین معیارهای انتخاب CH موضوع انرژی باقی مانده گره ها است که برای این کار از فرمول خاصی در الگوریتم های خوشه بندی استفاده می شود [7]. یکی از راهکارهای جلوگیری از مصرف بیش از حد انرژی گره ها این است که تعداد گره های در هر بازه زمانی در هر خوشه متغیر باشد و سرخوشه هایی انتخاب شوند که کمترین انرژی مصرفی را داشته باشند. برای این موضوع در این مقاله از الگوریتم HCEH-UC استفاده می شود. یکی از چالش های مهمی که انتخاب سرخوشه به صورت پویا دارد مسأله سازگاری سرخوشه با سایر گره ها است. برای حل این موضوع قبل از انتخاب گره به عنوان سرخوشه باید فیلد مقایسه جدیدی برای مقایسه سازگاری بین گره ها ایجاد کرد. در سیستم خوشه بندی پیشنهادی، گره ها به دو دسته حالت خواب و حالت انتقال داده تقسیم می شوند. گره ای که در حالت خواب قرار دارد گره ای است که باقی مانده انرژی اش از حد آستانه کمتر بوده و نیاز به شارژ مجدد دارد در حالی که گره در حالت انتقال داده انرژی باقی مانده اش از حد آستانه بیشتر است و می تواند هم به عنوان گره سرخوشه و هم به عنوان گره عادی به انتقال داده بپردازد [8,9].

در پروتکل مسیریابی RPL، ساختاری به نام DODAG توسط گره سینک برای مسیرهای رو به بالا و رو به پایین ایجاد می شود. برای انجام این کار، ابتدا گره سینک پیام DODAG (DIO) را برای گره های عادی پخش می کند. یک گره میانی به طور معمول

چندین پیام DIO را دریافت می‌کند که هر کدام از فرستنده‌های پیام DIO می‌تواند عنوان گره والد ترجیحی شناخته شود. در این حالت زمانی که بسته داده‌ای به سمت سینک ارسال شود (مسیریابی رو بالا) بسته مورد نظر باید از طریق گره والد ترجیحی ارسال شود. برای انتخاب گره والد ترجیحی فاصله گره نسبت به سینک مدنظر قرار می‌گیرد. در صورتی که گره والد ترجیحی از بین برود، باید گره میانی یک گره والد ترجیحی جدید را شناسایی کند.

در پروتکل مسیریابی RPL مسیریابی روبه بالا با استفاده از پیام DAO^۱ (انجام می‌گیرد. پیام DAO بعد از دریافت پیام DIO توسط گره میانی شکل می‌گیرد و این پیام به سمت گره سینک در مسیریابی رو به بالا ارسال می‌شود. گره والد ترجیحی، زمانی که پیام DAO را دریافت می‌کند پیام DAO را به سمت همسایگان خود می‌فرستد به طوری که آدرس خود را نیز در پیام منتشر شده قرار می‌دهد. بنابراین گره سینک جدول مسیریابی خود را براساس آدرس‌های DAO ثبت شده به‌روزرسانی می‌کند.

در نهایت گره سینک از همین طریق بسته‌های داده خود را به سمت پایین هدایت می‌کند. روال کار ارسال بسته توسط سینک به این صورت است که آدرس گره‌های میانی در هدر بسته ارسال وجود دارد و هر گره عادی که بسته مورد نظر را دریافت می‌کند از طریق آن هدر مسیر بعدی خود را پیدا کرده و بسته مورد نظر را ارسال می‌کند.

در صورتی یکی از گره‌های میانی از بین بروند بسته مورد نظر گم خواهد شد زیرا گره‌های عادی در RPL استاندارد جدول مسیریابی ندارند. برای حل این مشکل از روشی بنام تعمیر مسیر استفاده می‌شود. روش تعمیر مسیر در صورتی خرابی مسیر را تشخیص دهد با ارسال پیامی به نام DIS^۲ به گره‌های همسایه و گره سینک آن‌ها از این موضوع باخبر می‌کند. وقتی که گره سینک پیام DIS را دریافت می‌کند ابتدا جدول مسیریابی خود را به‌روزرسانی کرده پیام‌های بعدی را با توجه به هدر جدید به‌دست‌آمده ارسال می‌کند. در نهایت گره سینک خرابی مسیر را با ارسال پیام DIO به کل شبکه اطلاع‌رسانی می‌کند [10,11].

با این حال، اگر گره‌های بعدی تعیین‌شده از بین برود، گره میانی نمی‌تواند به‌تنهایی پرش بعدی جدید را تعیین کند زیرا گره میانی هیچ جدول مسیریابی ندارد (یعنی بسته مورد نظر در شبکه گم خواهد شد). از این رو، استاندارد RPL یک روش تعمیر مسیر را تعریف می‌کند. در این روش تعمیر مسیر، گره میانی که از دست‌دادن مسیر را تشخیص می‌دهد، پیام DIS را به سینک یا همسایگان آن ارسال می‌کند. هنگامی که یک سینک پیام DIS را دریافت می‌کند، سینک می‌تواند پیام DIO جدید را به کل شبکه ارسال کند تا مسیرهای رو به بالا و پایین را تجدید کند [12,13].

[14] یک پروتکل جدید مسیریابی جغرافیایی مبتنی بر شبکه چندمسیره (EMGGR) را برای UWSN پیشنهاد کرده‌است. کار توسعه‌یافته، EMGGR، شبکه را به عنوان شبکه‌های سه‌بعدی منطقی

می‌بیند. مسیریابی به صورت شبکه به شبکه از طریق دروازه‌هایی انجام می‌شود که از مسیرهای غیرمجاز برای انتقال بسته‌های داده به گره سینک استفاده می‌کنند. الگوریتم از سه جزء اصلی تشکیل شده‌است: (۱) الگوریتم انتخاب دروازه که مسئول انتخاب دروازه‌ها بر اساس مکان و سطح انرژی باقی‌مانده است (۲) مکانیزمی برای به‌روزرسانی اطلاعات دروازه‌های همسایه که اجازه‌دادن به گره‌های حسگر برای به خاطر سپردن دروازه‌ها در سلول‌های محلی و مجاور است و (۳) مکانیزم ارسال بسته. که مسئول ساخت مسیرهای مجزا از سلول‌های مبدا به سلول‌های مقصد، ارسال بسته‌ها به مقصد و برخورد با حفره‌ها (یعنی سلول‌های بدون دروازه) در شبکه است. اما با توجه به اینکه این الگوریتم با ویژگی‌های اینترنت اشیا سازگار نیست و همچنین در فضای سه‌بعدی عمل می‌کند برای مقایسه با الگوریتم پیشنهادی مناسب نیست.

در [15]، پروتکل RPLUW/M برای شبکه‌های اینترنت اشیا زیرآبی معرفی شده‌است که از ساختار درخت RPL برای کنترل توپولوژی و مسیریابی در محیط‌های زیرآب بهره می‌برد. این پروتکل با توجه به چالش‌های ارتباطات زیرآب از جمله تحرک گره‌ها، تأخیر طولانی، پهنای باند پایین، منابع محدود و نرخ خطای بالا طراحی شده‌است.

روش پیشنهادی در [16] با استفاده از رویکرد تصمیم‌گیری چندمعیاره و وزن‌دهی نامطمئن، گره‌های والد بهینه را برای هر سنسور انتخاب می‌کند. این انتخاب بهینه منجر به کاهش تعداد پرش‌ها، کاهش مصرف انرژی و افزایش کارایی کلی شبکه می‌شود. روش SWARA فازی، با در نظر گرفتن شرایط نامطمئن، امکان ارزیابی دقیق‌تر و بهینه‌تر مسیرها را فراهم می‌کند که می‌تواند به کاهش مصرف انرژی کمک کند.

۳. روش پیشنهادی

در این مقاله روشی جدید برای ارتقاء الگوریتم RPL ارائه شده‌است. در ابتدای کار، گره‌ها به خوشه‌های مختلف تقسیم می‌شوند، طوری که هر خوشه یک سرخوشه دارد و این سرخوشه بر اساس سانس انرژی باقی‌مانده نسبت به سایر گره‌ها، انتخاب می‌شود و سایر خوشه‌ها در جهت عمق کوچکتر خواهند شد. زیرا انتظار می‌رود بر اساس شرایط دما و فشار، اعماق بیشتر انرژی بیشتری مصرف کنند و شرایط پیچیده‌تری داشته‌باشند. علاوه بر آن از الگوریتم HCEH-UC جهت تنظیم انتقال داده در دو حالت خواب و حالت عملیات استفاده می‌شود که اعماق بیشتر حالت خواب طولانی‌تری را تجربه می‌کند. در شکل ۳ جزئیات روش پیشنهادی به صورت فلوچارت بیان شده طوری که روش پیشنهادی دارای ۴ قدم است. در قدم یک گره‌ها را مقداردهی اولیه می‌کنیم. در قدم ۲ خوشه‌بندی بر اساس عمق و اعمال الگوریتم HCEH-UC روی پروتکل مسیریابی RPL انجام می‌شود. در قدم ۳ به RPL حالت خواب بر اساس عمق الگوریتم HCEH-UC اعمال می‌شود تا توازن مصرف انرژی در گره‌ها انجام شود. در نهایت در قدم ۴ نیز از

آستانه E_{Δ} بیشتر باشد در غیر این صورت خاموش خواهد شد و دیگر در حالت فعال قرار نمی‌گیرد.

$$E_{\Delta} \geq Z * \left(E_{Bs} + \sum_{q=1}^Q E_{qs} \right), q \neq s \quad (3)$$

که در آن E_{Bs} میزان مصرف انرژی در حالتی که گره s سرخوشه است را نشان می‌دهد. E_{qs} به انتقال داده‌ها بستگی دارد و میزان مصرف انرژی را در آن نودها نشان می‌دهد. زمان شارژ مورد نیاز برای E_{Δ} برابر σ_E و زمان لازم برای اجرای کامل اطلاعات دریافتی برای یک خوشه خاص ΔT است. شبکه در صورتی به شکل مستمر به کار خود ادامه می‌دهد که انرژی باقی‌مانده و همچنین تعداد گره‌های حسگر عادی به اندازه کافی در حال تبادل داده باشند. اما با همه این تفاسیر باید افزونگی را مدنظر قرارداد. برای بررسی عملکرد معمول شبکه، شرایط مورد نیاز را باید به صورت (۴) نشان داد:

$$\sigma_E \leq (30\% * Q) * (Z * \Delta T) \quad (4)$$

به‌طور کلی کد زیر بیانگر پیاده‌سازی الگوریتم HCEH-UC روی RPL است.

Algorithm1: HCEH-UC routing algorithm

Input: X_i
 1-Generate initial clusters: $C_i = \{X_i\}, i \in M$
 2-Obtain node clusters adaptively according to T:
 $3-T = \sigma \text{Max} \left\{ \sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2} \right\}, i, j \in M, i \neq j$
 4- for $i = M + a, \dots$ do
 5-Guarantee the uninterrupted monitoring of UIoT:
 6- $E_{\Delta} \geq Z * (E_{Bs} + \sum_{q=1}^Q E_{qs}), q \neq s$
 7- $\sigma_E \leq (30\% * Q) * (Z * \Delta T)$
 8-for Each data transmission do
 9-the energy consumption of the sth cluster head:
 $10-E_{Bs} = E_{elec} * K_{Bs} + E_{DA} * K_{Bs} + E_{Tx} - elec(K_{Bs}) + E_{Tx} - amp(K_{Bs}, d)$
 11-the energy consumption of the qth node:
 $12-E_{qs} = E_{Tx} - elec(K_{qs}) + E_{Tx} - amp(K_{qs}, d), q \neq s$
 13-if $E_{rest} < E_{estimation}(s)$ then
 14-Select new cluster head:
 $15-p(q) = \begin{cases} 1 - \frac{E_{estimation}(q)}{E_{rest}(q)}, & q \in G \\ 0, & otherwise \end{cases}$
 16-Charging the exhausted node

ساختار داده بلوم فیلتر (بر اساس عمق خوشه) جهت تسریع در مسیریابی استفاده می‌شود.^۸ در ادامه به تفصیل هر یک از چهار قدم بیان می‌شود.

قدم ۱: مقداردهی اولیه به گره‌ها، ارسال پیام‌های کنترلی:

در این مرحله گره چاهک پیام کنترلی را به گره‌های دیگر در محدوده خود ارسال کرده و هر گره که این پیام را دریافت کند در فیلد Rank، ارزش خود را قرارداده و به گره‌های در محدوده خود ارسال می‌کند (پیام‌های DIO و DAO) به این ترتیب هر نود، همسایه خود را شناسایی کرده و فاصله از آن را محاسبه می‌کند.

قدم ۲: خوشه بندی

در این مرحله نودهای سرخوشه براساس انرژی باقی‌مانده و عمق با محاسبه میزان حد آستانه، نودهای سرخوشه را انتخاب می‌کند [56-59]. حد آستانه به این صورت تعریف می‌شود که هر گره‌ای که حد آستانه بالاتری دارد (یعنی بالاتر از 0.0015 ژول) در لیستی با عنوان لیست کاندیداهای سرخوشه قرار گرفته و می‌تواند برای دور بعد به عنوان سرخوشه آن گروه پیام‌ها را دریافت و به BS ارسال نماید. فرمول محاسبه حد آستانه از رابطه (۱) محاسبه می‌شود. رابطه زیر میزان حد آستانه را محاسبه می‌کند. همان‌طور که قبلاً اشاره شد هر چقدر میزان حد آستانه بیشتر باشد گره شانس بیشتری برای سرخوشه شدن دارد و در لیست کاندیداهای سرخوشه قرار خواهد گرفت.

$$f(n) = \begin{cases} \left(\frac{\text{Remaining Energy}}{Energy_{max}} \right) + \left(\frac{h_{node}}{h_{max}} \right) & \text{if } n \in G \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

n تعداد نودها و Remaining Energy انرژی باقی‌مانده نودها و h عمق گره است. بهبود نهایی اعمال شده و حد آستانه به شکل در رابطه ۲ تعریف می‌شود که در آن علاوه بر محاسبه باقی‌مانده انرژی گره و عمق، $distance_{ij}$ یعنی متوسط فاصله گره نسبت به سایر گره‌ها نیز در نظر گرفته شده است. هرچه این دو مقدار بیشتر باشد یعنی تعداد نودهای بیشتری شانس سرخوشه شدن دارند و این یعنی در اعماق بیشتر و با تعداد نودهای بیشتر در کنار هم خوشه‌ها کوچکتر خواهند شد. وجود مقادیر $Energy_{max}$ یعنی ماکسیمم انرژی ذخیره شده نودها، h_{max} عمق کف دریا و $max - distance$ بیشترین فاصله نودها از هم در سطح دریا تنها برای نرمالسازی فرمول به کار می‌رود. قضیه فاصله گره با دیگر گره‌ها در زمان ارسال بسته و مصرف انرژی گره تأثیر مستقیم دارد.

$$f(n) = \left(\frac{\text{Remaining Energy}}{Energy_{max}} \right) + \left(\frac{h_{node}}{h_{max}} \right) + \left(\frac{max-distance}{distance_{ij}} \right) \quad (2)$$

در رابطه ۲ در بازه زمانی i مصرف انرژی گره‌ها و فاصله آن‌ها با گره‌های محاور و عمق محاسبه می‌شود.

قدم ۳: اعمال الگوریتم RPL و HCEH-UC برای مسیریابی

در این مقاله، HCEH-UC روی RPL توسط شبیه‌ساز NS-3 برای انجام پوشش بدون وقفه گره‌های شبکه پیاده‌سازی شده است. رابطه ۳ بدین معنی است که مجموع انرژی گره سرخوشه همیشه باید از حد

در حالتی که انرژی باقی‌مانده گره به اتمام برسد باید الگوریتم خوشه‌بندی مجدداً شروع به انتخاب سرخوشه جدید نماید. دو فاز تشکیل خوشه و فاز انتقال داده در روش پیشنهادی اجرا شده است. اجرای الگوریتم خوشه‌بندی زمانی برای محاسبه فاصله بین گره‌ها، زمانی برای اجرای خوشه‌بندی و زمانی برای خاتمه خوشه‌بندی نیاز دارد.

علاوه بر این، در صورت فرسودگی غیرمنتظره گره، HCEH-UC پیشنهادی، خوشه‌بندی مجدداً اعمال خواهد شد و از گره‌های نرمال موجود در آن خوشه به عنوان کاندیدای سرخوشه شدن استفاده می‌شود. بنابراین، محیط پیشنهادی می‌تواند یک حالت تحویل اطلاعات قوی را به روشی توزیع شده با توجه به توزیع گره‌های شبکه ایجاد کند و این الگوریتم توانایی پیاده‌سازی در محیط‌های توزیع شده را نیز دارد [18]. زمان خاتمه خوشه‌بندی در بازه‌های زمانی مشخصی محاسبه می‌شود و هرچقدر تعداد این محاسبات افزایش یابد پیچیدگی زمانی نیز افزایش خواهد یافت. بنابراین متعادل کردن تعداد محاسبات یکی از چالش‌های اصلی الگوریتم HCEH-UC محسوب می‌شود.

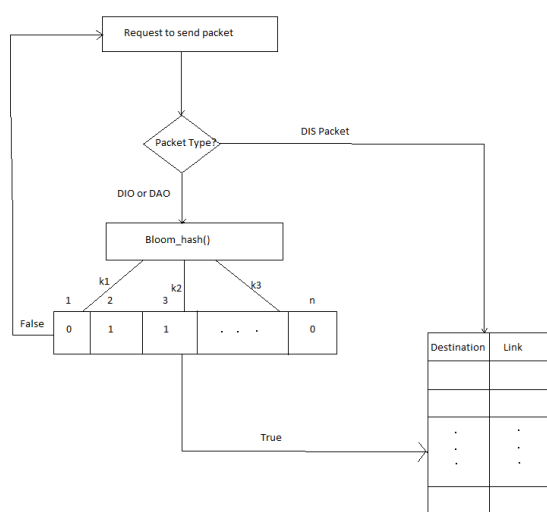
روی پیام‌های DIO، DAO و DIS در پروتکل مسیریابی RPL پیشنهادی تنظیماتی اعمال می‌شود. در حالت ارسال DIO (انتقال از بالا به پایین) گره‌هایی که در حالت خواب قرار دارند موقتاً پیام DIO به آن‌ها ارسال نمی‌شود و فقط گره‌هایی که در حالت انتقال داده قرار دارند پیام را دریافت می‌کنند. در حالت DAO (انتقال از پایین به بالا) گره مورد نظر فقط به گره‌های در حالت انتقال داده پیام ارسال می‌کند و به گره‌های همسایه که در حالت خواب هستند پیام ارسال نمی‌کند. در پیام‌های DIS، که مسئول انتشار خرابی مسیر هستند به عنوان بخشی از نوآوری فی‌لد جدید ایجاد می‌شود که گره‌هایی که در حالت خواب و گره‌هایی که در حالت انتقال داده هستند را از هم جدا کند و اطلاعات آن را به گره‌های همسایه و گره والد ترجیحی ارسال می‌نماید.

قدم ۴: اعمال بلوم فیلتر^۱ روی جدول مسیریابی گره‌ها

جهت بهبود حافظه مصرفی در این مقاله از ساختار داده ای به نام بلوم فیلتر برای خوشه‌های در عمق بیشتر استفاده خواهد شد. در صورتی که گره ای که قصد فرستادن بسته به آن را داریم در جدول مسیریابی وجود داشته باشد با هزینه کمی می‌توان بسته را به سمت مقصد فرستاد. برای این کار از یک آرایه به نام بلوم فیلتر استفاده می‌کنیم و در صورتی که گره مورد نظر در جدول مسیریابی وجود داشته باشد آرایه بلوم فیلتر مقدار True را در زمان $O(k)$ برمی‌گرداند (k در بلوم فیلتر بیانگر تعداد کلیدهایی است که در آرایه بلوم فیلتر اعمال شده است). و در نهایت نیز در زمان $O(n)$ و به صورت جستجوی ترتیبی در جدول مسیریابی پرش بعدی^{۱۱} را پیدا کرده و بسته را به آن ارسال می‌کند. در صورتی که گره مورد نظر که قصد فرستادن بسته به آن را داریم در جدول مسیریابی وجود نداشته باشد، مقدار False را برمی‌گرداند (مقدار False نیز در این حالت در زمان $O(k)$ مشخص می‌شود). نکته قابل توجه استفاده از بلوم فیلتر در این زمان مشخص می‌شود که دیگر نیازی به جستجوی ترتیبی با هزینه $O(n)$

نیست. برای اعمال بلوم فیلتر هر گره قبل از اینکه در جدول مسیریابی به جستجو پردازد، گره مقصد را به آرایه بلوم فیلتر می‌دهد و در صورتی که بلوم فیلتر مقدار True برگرداند جستجو در جدول مسیریابی گره ادامه پیدا می‌کند و در غیر این صورت نیازی به جستجو در جدول مسیریابی نیست.

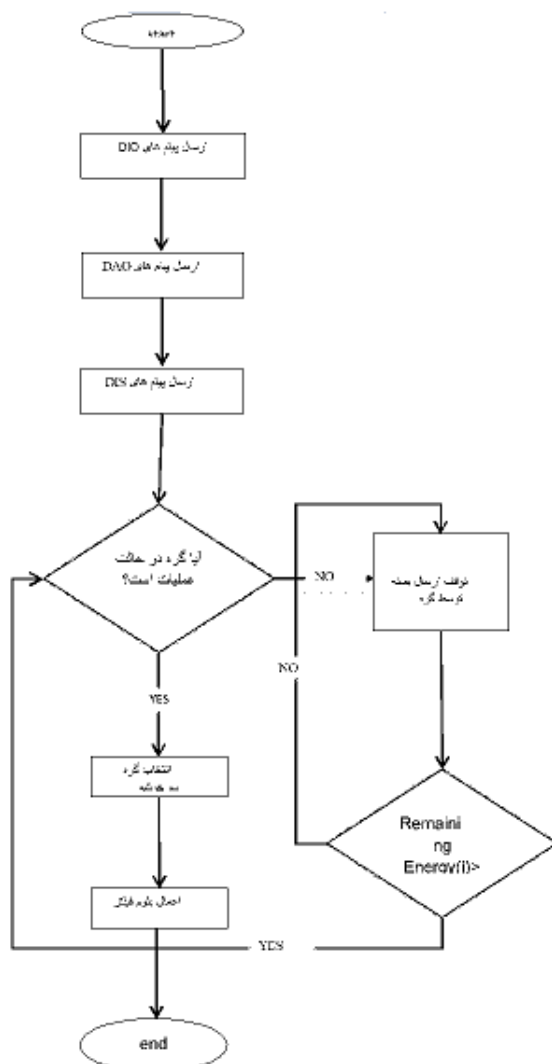
در شکل ۱ ساختار بلوم فیلتر الگوریتم پیشنهادی ارائه شده است. بلوم فیلتر از طریق کم کردن تعداد دسترسی به درایه‌های جدول مسیریابی هزینه جستجو را کاهش می‌دهد که در نهایت روی تعداد بسته‌های مسیریابی، نرخ تحویل بسته‌های مسیریابی و انرژی مصرفی گره‌ها تأثیر مثبت دارد. نحوه جستجو در این مقاله براساس توابع درهم‌سازی (hash) بوده که قبل از فرستادن به آرایه بلوم فیلتر استفاده می‌شود [14-17].



شکل ۱. سیستم بلوم فیلتر پیشنهادی

Hash برای گره i ، لینک گره i را به k تابع hash تبدیل کرده و به آرایه بلوم فیلتر استاندارد می‌دهد. در نهایت در صورتی که خروجی آرایه بلوم فیلتر استاندارد True باشد خروجی عددی Destination Node به جدول مسیریابی جهت یافتن گره مورد نظر و لینک مربوط به آن فرستاده می‌شود. به عنوان مثال قصد داریم از گره ۲ به گره ۴ بسته ارسال کنیم. ابتدا به تابع $Bloom_hash()$ گره مورد نظر را ارسال می‌کنیم. مثلاً گره ۴ دارای هش ۰۱۱۰ است یعنی می‌بایست خانه‌های شماره ۲ و ۳ در آرایه بلوم فیلتر دارای مقدار یک باشد (در صورتی که خانه‌های ۲ و ۳ دارای مقدار یک باشند یعنی گره مورد نظر در جدول مسیریابی وجود دارد در غیر این صورت گره در جدول مسیریابی نیست و آرایه بلوم فیلتر مقدار False را برمی‌گرداند).

در ساختار بلوم فیلتر پیشنهادی ابتدا باید نوع پیام ارسال شده در RPL مشخص شود. در صورتی که پیام DIS یا خرابی مسیر باشد نیازی به استفاده از بلوم فیلتر نیست زیرا باید به همه گره‌های مقصد موجود در جدول مسیریابی پیام مذکور ارسال شود. اما اگر پیام‌های از بالا به پایین و بالعکس باشد (DIO, DAO) چون به دنبال گره یا گره‌های



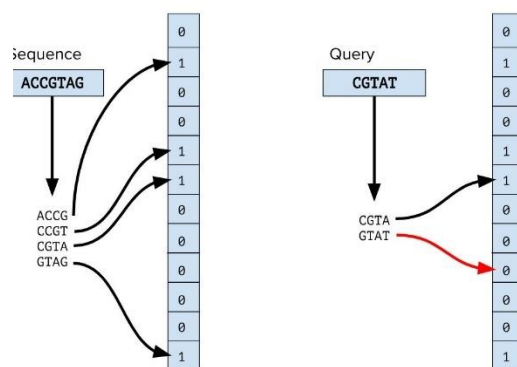
شکل ۳. فلوچارت نحوه فعالیت روش پیشنهادی (CB-RPL)

۴. تحلیل نتایج

برای ارزیابی، پروتکل پیشنهادی که در آن از خوشه‌بندی و بلوم فیلتر استفاده شده است با پروتکل RPL مقایسه شده که این مقایسه با استفاده از نرم‌افزار شبیه‌ساز NS3 ورژن 3.25 شبیه‌سازی و نتایج حاصل ارزیابی شده‌اند. (جدول ۱)

خاصی هستیم برای تسریع در پیدا کردن گره مقصد می‌توان آن را به آرایه بلوم فیلتر داد.

مطابق شکل ۲، طراحی فیلتر بلوم به این صورت است که اگر پاسخ پرس‌وجو از فیلتر بلوم منفی باشد، قطعاً عضو مورد نظر در مجموعه متناظر وجود ندارد (و این را صراحتاً می‌توانیم اعلام کنیم). اما اگر پاسخ پرس‌وجو مثبت باشد، نمی‌توان به‌طور قطع گفت که این عضو در مجموعه مورد نظر وجود دارد یا خیر و باید عملیات محاسباتی دیگری را برای تأیید یا رد ادعای خود به‌کارگیریم.



شکل ۲. نحوه ثبت و پرس‌وجوی عناصر در آرایه بلوم فیلتر

در بلوم فیلتر استاندارد، k تابع جستجوگر hash داریم که برای درج عناصر در آرایه بلوم فیلتر مورد استفاده قرار می‌گیرد. در شبه‌کد ۱ ابتدا تابع bloom_hash مشخص می‌کند که آیا گره مورد نظر در آرایه BF وجود دارد یا خیر. اگر وجود داشته باشد تابع مقدار null را برمی‌گرداند و اگر گره مورد نظر در آرایه BF یافت شود برای جستجو گره مورد نظر، به اندازه مقدار ساینز بسته hash گرفته شده و به آرایه بلوم فیلتر در نهایت ارسال می‌شود. الگوریتم ۲ بلوم هش را نشان می‌دهد. روش bloom Filter که بر اساس هش کار می‌کند بهترین روش کاهش حافظه است. دسترسی به مدخل‌ها $O(1)$ است و تنها مشکل آن این است که مورد در حافظه موجود نباشد که در این صورت ایده‌هایی برای دسترسی به مدخل در کارهای آینده ارائه می‌شود

```
Algorithm2. Bloom_hash()
Bloom_hash(link node, size of packet)
1. while(k!=0)
2.   if(link node is NULL)
3.     return NULL
4.   Else
5.     {
6.       hash function(link node)
7.       k—
8.     }
9. return link node
```

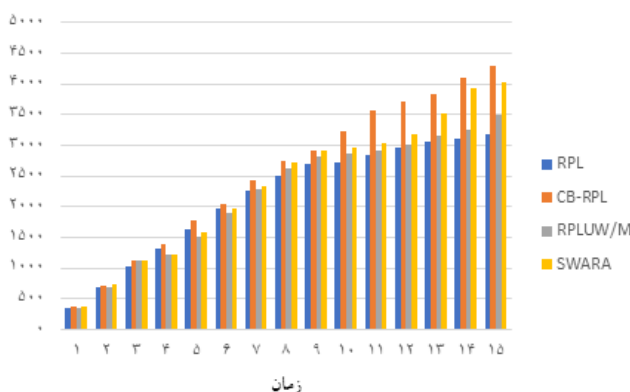
جدول ۱. شرایط شبیه‌سازی روش پیشنهادی

Transmission protocol	UDP
Routing protocol	RPL
Average initial energy of root and non-root nodes	0.5j
Data-Packet Size	4000bit
Packet Header Size	25 bytes
Control Message Size	50 bytes
Transmission protocol	UDP
Queue length	20
Number of normal nodes	100
Bit rate	Fixed
Generated data type	CBR
E_{elec}	50 nJ/bit
E_{fs}	10 pJ/bit/m ²
E_{mp}	0.0013 pJ/bit/m ⁴
E_{DA}	5 nJ/bit/message
Transfer rate	18 Mbps

هدف اصلی که در این مقاله دنبال می‌شود ایجاد توازن انرژی مصرفی در طول اجرای شبیه‌سازی با استفاده از اعمال بهبود یافته الگوریتم HCEH-UC در قالب الگوریتم پیشنهادی، بهبود حافظه گره‌ها از طریق کم کردن تعداد دسترسی به مدخل‌های جدول مسیریابی و بهبود در تعداد بسته‌های مسیریابی منتقل شده است.

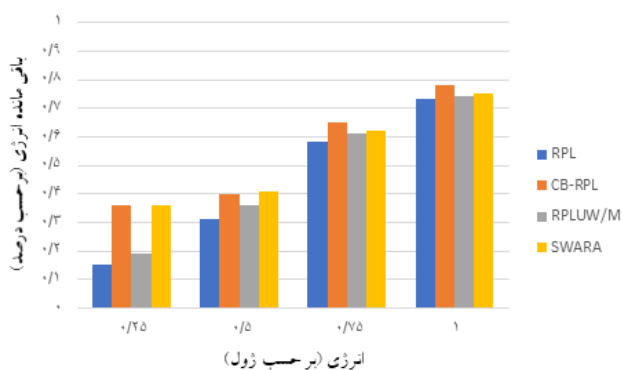
(۱) محاسبه داده منتقل شده در طول زمان

داده منتقل شده براساس RPL مقاله بیس و الگوریتم پیشنهادی نشان داده شده است. مطابق شکل ۴، بعد از هرچه به پایان شبیه‌سازی نزدیک شویم در حالت CB-RPL به دلیل استفاده گره‌ها در حالت خواب و عملیات، تعداد بسته‌های منتقل شده نسبت به RPL بیس بیشتر شده است. علت آن ایجاد خوشه‌های کوچکتر در عمق آب است که باعث می‌شود نودهای بیشتری در انتقال داده شرکت و حجم بیشتری از داده‌ها را منتقل کنند



شکل ۴. داده منتقل شده بر حسب زمان شبیه‌سازی

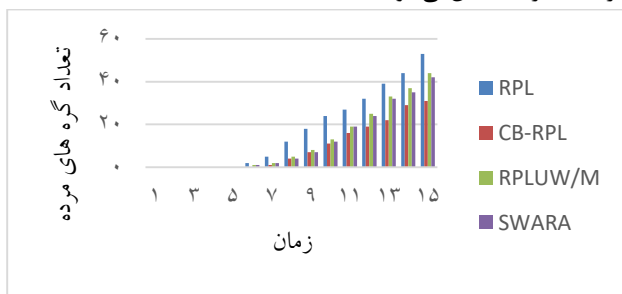
(۲) میانگین انرژی باقیمانده سایر نودها بعد از مرگ اولین نود هر چقدر انرژی اولیه گره‌ها کمتر باشد روش پیشنهادی کارایی بهتری نسبت به RPL مقاله بیس دارد. اما ملاحظه می‌کنید که چنانچه انرژی اولیه هر گره 1 ژول باشد اختلاف زیادی بین این دو روش مطرح نیست. (شکل ۵) علت آن است که در مقدار اولیه انرژی زیاد انرژی باقیمانده در فرمول ۲ همچنان مقادیر انرژی باقیمانده بر ماکسیمم انرژی مقادیر بزرگتری به خود می‌گیرد و در نتیجه مقدار آستانه بالاتر بوده و تعداد سرخوشه‌ها و خوشه‌ها کمتر بوده و نتیجه انرژی مصرفی بیشتر می‌شود. این مورد برای انرژی اولیه کم برعکس بوده و باعث کاهش مصرف انرژی بیشتر الگوریتم پیشنهادی به نسبت الگوریتم اولیه می‌باشد، در نتیجه انرژی باقیمانده در این حالت بیشتر خواهد بود. برای محاسبه این پارامتر، شبیه‌سازی در ۴ مرحله با انرژی‌های اولیه ۰.۲۵، ۰.۵، ۰.۷۵ و ۱ ژول برای گره‌های Root و گره‌های معمولی اعمال شده است که نتایج آن به شکل زیر است:



شکل ۵. میانگین انرژی باقیمانده نودها

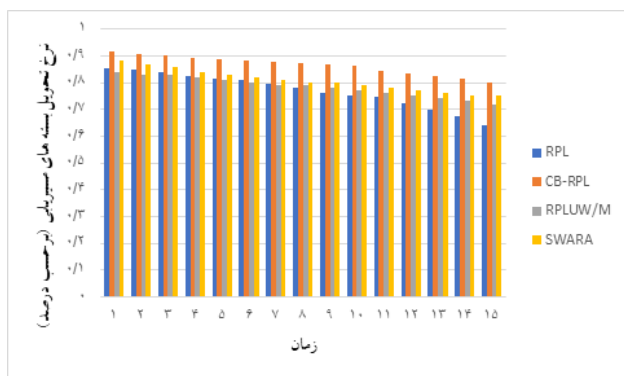
(۳) تعداد نودهای از بین رفته بر حسب زمان

در شکل ۶ تعداد نودهای از بین رفته بر حسب زمان نشان داده شده است. در حالتی که ورودی بسته‌های مسیریابی گره زیاد باشد و در همین حال انرژی گره در سطح پایینی قرار داشته باشد CB-RPL نمی‌تواند گره را به حالت خواب منتقل کند و گره می‌میرد. به این ترتیب با الگوریتم پایه RPL تفاوت چندانی ندارد. همان طور که در شکل زیر مشخص است هرچقدر زمان شبیه‌سازی به انتها نزدیک می‌شود اختلاف معناداری بین CB-RPL و دیگر الگوریتم‌ها در حوزه گره‌های مرده نمایان می‌شود.



شکل ۶. تعداد نودهای از بین رفته بر اساس زمان

(۴) تعداد دسترسی‌های مدخل جدول مسیریابی



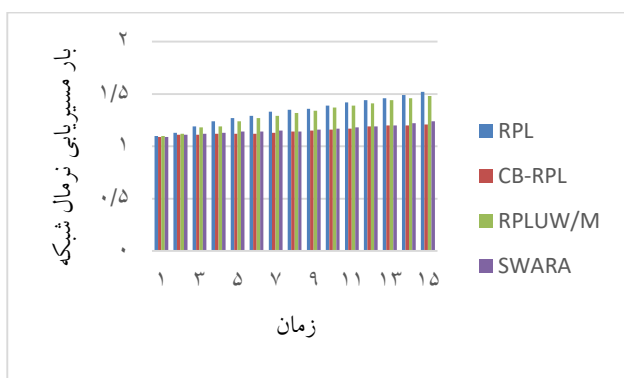
شکل ۸. نرخ تحویل بسته های مسیریابی

(۶) بار مسیریابی نرمال شده

در شکل زیر هر چقدر NRL (Normalized routing load) کمتر

بیشتر باشد به معنای بالا بودن میزان ترافیک در شبکه است و این موضوع می تواند به گم شدن بسته ها و مصرف انرژی بیش از حد منجر شود. یکی از دلایل اصلی بالا رفتن NRL این است که در برخی از موارد حجم عظیمی از بسته ها به سمت یک گره خاص ارسال می شود که این قضیه باعث خاموش شدن گره می شود در حالی که در روش پیشنهادی زمانی که یک گره بیش از حد مجاز انرژی مصرف کند، گره با ارسال پیام به همسایگان اعلام می کند که در حالت به خواب رفتن است و همسایگان از این پس برای گره بسته ارسال نخواهند کرد. بنابراین یکی از دلایل توازن NRL اعمال الگوریتم HCEH-UC روی پروتکل مسیریابی RPL است و SWARA و RPLUW/M چون خوشه بندی ندارد نمی تواند عملکرد مناسب داشته باشد. (شکل ۹) فرمول زیر نحوه محاسبه NRL را نشان می دهد. تعداد بسته های ارسالی، تعداد بسته های فوروارده شده توسط نودهای میانی و تعداد بسته های دریافت شده از جمله متغیرهای این فرمولند.

$$NRL = \frac{\text{Send Packets} + \text{Forward Packets}}{\text{Received Packets}} \quad (6)$$



شکل ۹. بار مسیریابی نرمال شده بر حسب زمان

(۷) تعداد بسته های گم شده

شکل ۱۰ تعداد بسته های گم شده بر حسب زمان را

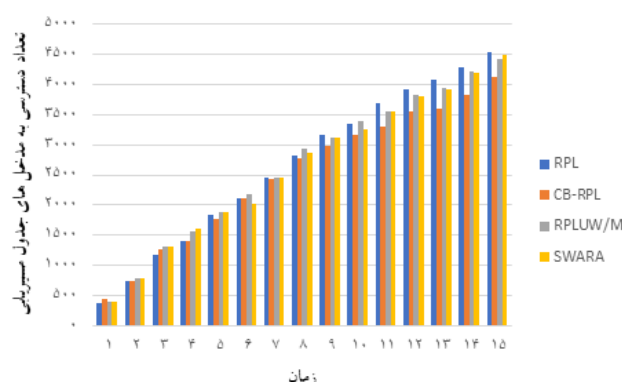
نشان می دهد. گم شدن بسته ها به دو سناریو بستگی دارد. اولاً هر قدر به انتهای شبیه سازی نزدیک تر شویم تعداد گره های مرده افزایش یافته و تعداد بسته های گم شده نیز افزایش می یابد. ثانیاً میزان گم شدن

مطابق شکل ۷، در بعضی زمان ها در CB-RPL دسترسی ها به

مدخل جدول مسیریابی از دسترسی به مدخل RPL پایه مقداری بیشتر است. زیرا در برخی از موارد بلوم فیلتر به اشتباه عنصر مورد نظر را جستجو کرده در حالی که پس از جستجوی عنصر، در جدول مسیریابی یافت نمی شود.

دلیل بیشتر شدن تعداد دسترسی CB-RPL به جدول

مسیریابی در انتهای شبیه سازی بالا رفتن تعداد بسته های ارسالی و دریافتی است که این موضوع نه تنها تأثیر منفی ندارد بلکه استفاده از بلوم فیلتر می تواند در تسریع در ارسال بسته ها و حتی حفظ مصرف انرژی گره ها نیز تأثیر مثبت داشته باشد. RPLUW/M و SWARA چون کلاً کنترلی روی حافظه ندارد، نمی تواند عملکرد مناسبی داشته باشد.



شکل ۷. تعداد دسترسی به مدخل جدول مسیریابی

(۵) نرخ تحویل بسته های مسیریابی

شکل ۸ نرخ پخش بسته را بر حسب زمان نشان می دهد. این

مقدار packet delivery rate (PDR) از طریق فرمول زیر محاسبه می شود. تعداد بسته های ارسالی، تعداد بسته های فوروارده شده توسط نودهای میانی و تعداد بسته های دریافت شده از جمله متغیرهای این فرمولند.

$$PDR = \frac{\text{Received Packets}}{\text{Send Packets} + \text{Forward Packets}} \quad (5)$$

گاهی به دلیل پر شدن بافر ورودی یک گره بسته دراپ

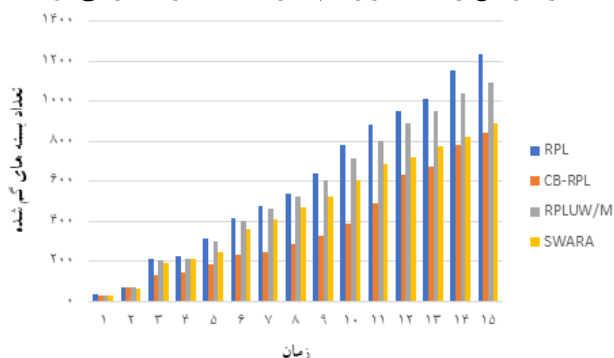
خواهد شد اما در CB-RPL به دلیل استفاده از بلوم فیلتر، دسترسی به حافظه سریع تر صورت گرفته و باعث کمتر شدن بسته های دراپ شده می شود. یکی از چالش ها در پیاده سازی الگوریتم HCEH-UC مسئله گره های خواب است. برای جلوگیری از دراپ شدن بسته ها در CB-RPL بایستی قبل از به خواب رفتن گره، به گره های مجاور پیام ارسال شود تا از این پس بسته، به گره در خواب رفته ارسال نشود.

در این صورت ایده‌هایی برای دسترسی به مدخل در کارهای آینده ارائه می‌شود.

مراجع

- [1] Lin, Shan, Fei Miao, Jingbin Zhang, Gang Zhou, Lin Gu, Tian He, John A. Stankovic, Sang Son, and George J. Pappas. "ATPC: Adaptive transmission power control for wireless sensor networks." *ACM Transactions on Sensor Networks (TOSN)* 12, no. 1 (2016): 1-31. doi: 10.1145/2746342
- [2] Nutov, Zeev, and Michael Segal. "Improved approximation algorithms for maximum lifetime problems in wireless networks." *Theoretical Computer Science* 453 (2012): 88-97. doi: 10.1016/j.tcs.2011.08.001
- [3] Yildiz, Huseyin Ugur, Bulent Tavli, and Halim Yanikomeroglu. "Transmission power control for link-level handshaking in wireless sensor networks." *IEEE Sensors Journal* 16, no. 2 (2015): 561-576. doi: 10.1109/jsen.2015.2486960
- [4] Al Salti, F.; Alzeidi, N.; Arafah, B.R. EMGGR: An energy-efficient multipath grid-based geographic routing protocol for underwater wireless sensor networks. *Wirel. Netw.* 2017, 23, 1301–1314. doi: 10.1007/s11276-016-1224-0
- [5] Zhuo, Xiaoxiao, Meiyun Liu, Yan Wei, Guanding Yu, Fengzhong Qu, and Rui Sun. "AUV-aided energy-efficient data collection in underwater acoustic sensor networks." *IEEE Internet of Things Journal* 7, no. 10 (2020): 10010-10022. doi: 10.1109/iot.2020.2988697
- [6] Wang, Jin, Yu Niu, Jinsung Cho, and Sungyoung Lee. "Analysis of energy consumption in direct transmission and multi-hop transmission for wireless sensor networks." In *2007 Third International IEEE Conference on Signal-Image Technologies and Internet-Based System*, pp. 275-280. IEEE, 2007. doi: 10.1109/sitis.2007.145
- [7] Rana, Sohail, Ali Newaz Bahar, Nazrul Islam, and Johirul Islam. "Fuzzy based energy efficient multiple cluster head selection routing protocol for wireless sensor networks." *International Journal of Computer Network and Information Security* 7, no. 4 (2015): 54-61. doi: 10.5815/ijcnis.2015.04.07
- [8] Borgia, Eleonora. "The Internet of Things vision: Key features, applications and open issues." *Computer Communications* 54 (2014): 1-31. doi: 10.1016/j.comcom.2014.09.008
- [9] Ko, J., Jeong, J., Park, J., Jun, J., Gnawali, O., & Paek, J. (2015). DualMOP-RPL: Supporting multiple of mode downward routing in a single RPL network. *ACM Transaction on Sensor Network (TOSN)*, 11(2), may 2015. doi: 10.1145/2700261
- [10] Jeong, Yonghwan, Sungwon Lee, Eunbae Moon, Hyeoncheol Park, Hongseok Yoo, and Dongkyun Kim. "Hop-interval based decision of operational mode in RPL with multi-instance." In *2016 Eighth International Conference on Ubiquitous and Future Networks (ICUFN)*, pp. 591-593. IEEE, 2016. doi: 10.1109/icufn.2016.7537101
- [11] Han, Bing, Feng Ran, Jiao Li, Limin Yan, Huaming Shen, and Ang Li. "A novel adaptive cluster based routing protocol for energy-harvesting wireless sensor networks." *Sensors* 22, no. 4 (2022): 1564. doi: 10.3390/s22041564
- [12] Bahrambeigy, Bahram, Mahmood Ahmadi, and Mahmood Fazlali. "Towards Accelerating IP Lookups on Commodity PC Routers using Bloom Filter: Proposal of Bloom-Bird." *Information Systems &*

بسته‌ها به محیط شبیه‌سازی نیز بستگی دارد. هر قدر محیط شبیه‌سازی وسیع‌تر باشد میزان گم‌شدن بسته‌ها نیز بیشتر می‌شود.



شکل ۱۰. تعداد بسته‌های گم شده بر اساس زمان

۵. نتیجه‌گیری

اینترنت اشیاء زیر آب طی چند سال گذشته کاربردهای متعددی در زمینه‌های مختلف پیدا کرده است. در روش پیشنهادی که با عنوان CB-RPL شناخته می‌شود ابتدا برای دقت بیشتر، کل نودها به خوشه‌های کوچکتر تقسیم شده و هر سرخوشه مسئول رساندن بسته‌ها به نودهای داخل خوشه است و سعی می‌کند که با اعمال الگوریتم خوشه‌بندی HCEH-UC توسعه یافته توازن مصرف انرژی بین تمام گره‌های موجود در شبکه را برقرار کند. بدون شک خوشه‌بندی‌های انجام شده به کمک ژنتیک و کرم شب‌تاب [19] و دسته‌بندی [20] یا انواع متاهوریسک‌ها می‌تواند خوشه‌بندی بهتری ایجاد کند که پیشنهاد می‌شود در آینده مطالعات برای استفاده در خوشه‌بندی زیر آب انجام شود. همچنین برای بهبود سرعت مسیریابی از بلوم فیلتر استفاده خواهد شد. روش هش بهترین روش کاهش حافظه است. دسترسی به مدخل‌ها $O(1)$ است و تنها مشکل آن این است که موردی که به دنبال آن می‌گردیم در حافظه موجود نباشد که در این صورت ایده‌هایی برای دسترسی به مدخل در کارهای آینده ارائه می‌شود. تصمیم‌گیری برای انتخاب بهترین مسیر به پارامترهای مختلفی بستگی دارد که بسیاری از این پارامترها با استفاده از ساختار داده بلوم فیلتر و انتخاب بهترین مسیر از طریق ساختار DODAG مطرح می‌شود که در نهایت منجر به کاهش دسترسی به مدخل‌های جدول مسیریابی و بهبود در مصرف انرژی گره‌ها از طریق استفاده از بلوم فیلتر می‌شود. نتایج مراحل ارزیابی، بهبود چشمگیر و قابل توجهی در CB-RPL نسبت به الگوریتم RPL نشان می‌دهد. به عنوان پیشنهاد کارهای آتی می‌توان به مسأله توازن انرژی مصرفی با استفاده از الگوریتم HCEH-UC برای سایر پروتکل‌های مسیریابی در اینترنت اشیاء زیر آب اشاره کرد. روش bloom Filter که بر اساس هش کار می‌کند بهترین روش کاهش حافظه است. دسترسی به مدخل‌ها $O(1)$ است و تنها مشکل این است که موردی که مورد نیاز است در حافظه موجود نباشد؛

- Transactions on wireless communications 1, no. 4 (2002): 660-670. doi: 10.1109/twc.2002.804190
- [19] Sah, Dipak Kumar, and Tarachand Amgoth. "A novel efficient clustering protocol for energy harvesting in wireless sensor networks." *Wireless Networks* 26, no. 6 (2020): 4723-4737. doi: 10.1007/s11276-020-02351-x
- [20] Khan, Muhammad K., Muhammad Shiraz, Qaisar Shaheen, Shariq Aziz Butt, Rizwan Akhtar, Muazzam A. Khan, and Wang Changda. "Hierarchical routing protocols for wireless sensor networks: functional and performance analysis." *Journal of Sensors* 2021 (2021): 1-18. doi: 10.15368/theses.2012.175
- [21] Karimi, Hamed, Keyhan Khamforoosh, and Vafa Maihami. "Improvement of DBR routing protocol in underwater wireless sensor networks using fuzzy logic and bloom filter." *Plos one* 17, no. 2 (2022): e0263418. doi: 10.1371/journal.pone.0263418
- [22] Mahsa Afsardeir, Mansuerh Afsar deir, "A New Method fo Data Clustering based on The Combination of Genetic Optimization and Firefly Algorithms" *Intelligent Multimedia Processing and Communication Systems(IMPCS)*, 4(2), 35-43, 2022
- [23] S. Ansari pour, T. J. Gandomani, "Comparison of the classification methods in software development effort estimation", *Intelligent Multimedia Processing and Communication Systems(IMPCS)*, 3(2), 11-18, 2022
- Telecommunication (2017): 25. doi: 10.1109/noms.2014.6838341
- [13] Naous, Jad, Glen Gibb, Sara Bolouki, and Nick McKeown. "NetFPGA: reusable router architecture for experimental research." In *Proceedings of the ACM workshop on Programmable routers for extensible services of tomorrow*, pp. 1-7. 2008. doi: 10.1145/1397718.1397720
- [14] Al Salti, F., Alzeidi, N. & Arafah, B.R. EMGGR: an energy-efficient multipath grid-based geographic routing protocol for underwater wireless sensor networks. *Wireless Netw* 23, 1301–1314 (2017). <https://doi.org/10.1007/s11276-016-1224-0>
- [15] Homaei, Mohammadhossein. "RPLUW/M: Enabling RPL on the Internet of Underwater Things." *arXiv preprint arXiv:2408.08607* (2024).
- [16] Karkehabadi, Ali, Mitra Bakhshi, and Seyed Behnam Razavian. "Optimizing Underwater IoT Routing with Multi-Criteria Decision Making and Uncertainty Weights." *arXiv preprint arXiv:2405.11513* (2024).
- [17] Lucchesi, Alexandre, André C. Drummond, and George Teodoro. "High-performance IP lookup using Intel Xeon Phi: a Bloom filters based approach." *Journal of Internet Services and Applications* 9 (2018): 1-18. doi: 10.1186/s13174-017-0075-y
- [18] Heinzelman, Wendi B., Anantha P. Chandrakasan, and Hari Balakrishnan. "An application-specific protocol architecture for wireless microsensor networks." *IEEE*

پی‌نوشت

- ² IPv6 Routing Protocol For Low Power And Lossy Networks
- ³ hierarchical-clustering-based routing protocol energy harvesting(EH)
- ⁴ Destination Oriented Directed Acyclic Graph