PLAER: الگوریتم مسیریابی آگاه از انرژی در شبکه‌های حسگر بی‌سیم به کمک مفهوم جریمه در اتوماتای یادگیر

مرتضی پرویزی عمران، علی معینی و حمید حاج سید جوادی

چکیده: حسگرهای موجود در شبکه‌های حسگر بی‌سیم معمولاً با انرژی باتری و با عمر محدود کار می‌کنند، به همین دلیل کاهش مصرف انرژی در آنها از اهمیت بسزایی برخوردار است. در این مقاله، الگوریتم مسیریابی جدیدی جهت کاهش مصرف انرژی در این شبکه‌ها معرفی می‌شود که از اتوماتای یادگیر به منظور یافتن مسیر مناسب جهت ارسال بسته‌های داده بهره می‌گیرد. رویکرد اصلی این الگوریتم به این صورت است که مصرف انرژی در مسیرهای مختلف را با در نظر گرفتن سطح انرژی و تأخیر گره‌ها متوازن نگه می‌دارد و بدین منظور از شیوه جریمه‌دهی در اتوماتای یادگیر بهره می‌گیرد. برای ارزیابی کارایی الگوریتم پیشنهادی، این پروتکل مسیریابی با نرم‌افزار OMNET++ شبیه‌سازی و نتایج به دست آمده با دو پروتکل LABER و BEAR مقایسه شده است. نتایج شبیه‌سازی نشان می‌دهد که در شبکه با ساختار استاتیک، در مصرف انرژی و ارسال بسته‌های کنترلی و در نتیجه طول عمر شبکه در پروتکل پیشنهادی نسبت به پروتکل‌های مقایسه‌شده بهبود حاصل شده است.

***کلید واژه:* اتوماتای یادگیر، شبکه‌های حسگر بی‌سیم، کاهش مصرف انرژی، مسیریابی آگاه از انرژی.**

۱- مقدمه

استفاده از شبکه‌های حسگر برای مقاصد مختلف نظیر شناسایی محیط، مطالعه شرایط آب و هوایی، کاربردهای نظامی، صنعتی، پزشکی و ... در دهه‌های اخیر بسیار مورد توجه قرار گرفته است. حسگرهایی که برای این منظور طراحی شده‌اند عمدتاً با انرژی باتری کار می‌کنند که عمر محدودی دارند. مسأله انرژی در شبکه‌های حسگر از آنجا دارای اهمیت است که با از کار افتادن حسگری که در شبکه در نقطه حساسی قرار گرفته، ممکن است کل شبکه از کار بیفتد. از آنجایی که تعویض یا شارژ مجدد باتری حسگرها دشوار و در برخی کاربردهای خاص نظیر کاربردهای نظامی غیر ممکن است، مسأله کاهش مصرف انرژی و در نتیجه افزایش طول عمر شبکه مسأله‌ای بسیار مورد توجه است [۱].

اندازه‌گیری‌های تجربی نشان می‌دهد که عموماً ارسال داده در شبکه‌های حسگر بی‌سیم بیشترین میزان مصرف انرژی را دارد. مصرف انرژی ارسال یک بیت اطلاعات در شبکه تقریباً معادل اجرای هزار دستور در سیستم پردازشی گره است و از این رو طراحی و گسترش پروتکل‌های مسیریابی آگاه به مصرف انرژی در این شبکه‌ها از اهمیت بسزایی برخوردار است [۲].

اتوماتای یادگیر مدلی مجرد است که برای حل مسایل یادگیری در محیط‌های تصادفی و ناشناخته مناسب است که عملی را از مجموعه اعمال مجاز خود و بر اساس یک توزیع احتمال انتخاب می‌کند و با دریافت بازخورد عمل انجام‌شده از محیط، توزیع احتمال خود را به روز می‌کند. انواع مختلف اتوماتای یادگیر برای کاربردهای مختلفی ارائه شده که در این مقاله مدل اتوماتای یادگیر با ساختار متغیر و با مدل یادگیری فقط جریمه‌دهنده به کار گرفته شده است [۳].

الگوریتم‌های مختلفی برای مسیریابی در شبکه‌های حسگر بر اساس مفهوم اتوماتای یادگیر ارائه شده که اساس کار غالب این الگوریتم‌ها این است که در هر گره شبکه یک اتوماتای یادگیر به منظور پیداکردن گره مناسب جهت ارسال بسته به آن به کار گرفته می‌شود، در صورتی که مسیر انتخاب‌شده مناسب باشد پاداش می‌گیرد و در غیر این صورت جریمه خواهد شد. در ادامه برخی از این الگوریتم‌ها به اختصار مطرح می‌شوند: الگوریتم‌های TinyLab [۴] و LABER [۵] به منظور پاداش یا جریمه‌دهی به مسیرها از یک بسته کنترلی استفاده می‌کنند که هر گره با دریافت بسته اطلاعات، این بسته کنترلی را که محتوی سطح انرژی گیرنده است، تولید و به فرستنده ارسال می‌کند. فرستنده با دریافت این بسته کنترلی و مقایسه سطح انرژی گیرنده با میانگین سطح انرژی همسایه‌های دیگر، به مسیر انتخاب‌شده پاداش یا جریمه می‌دهد. فاز غرقه‌سازی[1] در این الگوریتم‌ها از گره مقصد یا همان چاهک[2] و به منظور شناسایی مسیرهای اولیه انجام می‌شود.

اساس الگوریتم ICLEAR مشابه دو الگوریتم پیشین است با این تفاوت که قبل از ارسال هر بسته داده‌ای به منظور مسیریابی، فاز غرقه‌سازی شبکه از گره(های) مبدأ انجام می‌شود و از این جهت سربار زیادی دارد [۶].

اما الگوریتم BEAR به گونه‌ای طراحی‌شده که بسته کنترلی مجزا ندارد و از همان بسته‌های اطلاعاتی جهت پاداش‌دهی و یا جریمه‌دادن استفاده می‌کند. ایده دیگری که در این الگوریتم مطرح شده، ارسال سطح انرژی گره‌ها به همراه بسته‌های داده‌ای است. سربار اضافی که در این الگوریتم وجود دارد ارسال بسته داده‌ای به همه همسایه‌هاست [۷].

در الگوریتم DEEB از اتوماتای یادگیر به منظور شکل‌گیری یک ستون فقرات[3] جهت مسیریابی در شبکه حسگر استفاده شده است [۸]. در حین اجرای این الگوریتم لیست گره‌های انتخاب‌شده برای شکل‌گیری ستون فقرات به همه گره‌ها ارسال می‌شود، به همین دلیل از جمله مشکلات DEEB بالارفتن حجم اطلاعات کنترلی ارسالی و در نتیجه آن افزایش مصرف انرژی با افزایش تعداد گره‌های شبکه می‌باشد.

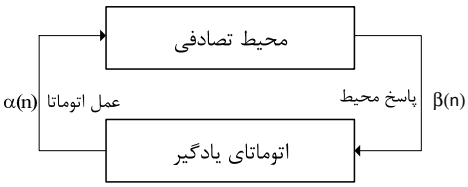
این مقاله در تاریخ ۶ خرداد ماه ۱۳۹۲ دریافت و در تاریخ ۱ دی ماه ۱۳۹۲ بازنگری شد.

مرتضی پرویزی عمران، دانشکده علوم مهندسی، پردیس دانشکده های فنی، دانشگاه تهران، تهران (email: m.parvizi@ut.ac.ir).

علی معینی، دانشکده علوم مهندسی، پردیس دانشکده های فنی، دانشگاه تهران، تهران (email: moeini@ut.ac.ir) .

حمید حاج سید جوادی، گروه ریاضی و علوم کامپیوتر، دانشگاه شاهد، تهران (email: h.s.javadi@shahed.ac.ir).

1. Flooding
2. Sink
3. Back Bone



شکل ۱: تعامل بین اتوماتای یادگیر و محیط.

الگوریتم پیشنهادی با عنوان PLAER[1] که در این مقاله ارائه شده به منظور کاهش تعداد بسته‌های کنترلی، فقط از بسته‌های کنترلی جریمه[2] استفاده می‌کند. همچنین در بسته‌های اطلاعاتی ارسالی میانگین سطح انرژی همسایه‌های گره فرستنده نیز به منظور مقایسه با سطح انرژی گره گیرنده ارسال می‌شود. مقایسه سطح انرژی در این الگوریتم در گره گیرنده انجام می‌شود و در صورت نیاز بسته‌های جریمه تولید و به فرستنده ارسال می‌شود. این الگوریتم از جنبه‌هایی تلفیقی از LABER و BEAR به شمار می‌آید و به همین دلیل به همراه دو الگوریتم مذکور در شبیه‌ساز OMNET++ پیاده‌سازی شده و نتایج شبیه‌سازی، بهبود در مصرف انرژی و طول عمر شبکه را نسبت به دو الگوریتم دیگر نشان می‌دهد.

ادامه این مقاله بدین صورت سازماندهی شده است: در بخش ۲ مفهوم و مدل اتوماتای یادگیر به اختصار آمده است. در بخش ۳ الگوریتم‌های LABER و BEAR و سپس الگوریتم پیشنهادی شرح داده شده‌اند. بخش ۴ مربوط به شبیه‌سازی و تحلیل نتایج بوده و نهایتاً نتیجه‌گیری در بخش ۵ آمده است.

۲- اتوماتای یادگیر

اتوماتای یادگیر مدلی مجرد است که تعداد معدودی عمل را می‌تواند انتخاب کند. هر عمل انتخاب‌شده، توسط محیط تصادفی ارزیابی شده و پاسخی تحت عنوان سیگنال تقویتی به اتوماتای یادگیر ارسال می‌شود. وضعیت داخلی اتوماتا با دریافت این سیگنال تقویتی و با توجه به عملی که انجام داده بود، به روز گردیده و عمل بعدی توسط اتوماتا انتخاب می‌شود. با تکرار این فرایند اتوماتا یاد می‌گیرد که عمل بهینه را انتخاب نماید. شکل ۱ ارتباط بین اتوماتای یادگیر و محیط را نشان می‌دهد.

اتوماتای یادگیر با ساختار ثابت به صورت پنج‌تایی $\{\alpha,\beta,F,G,\phi\}$ تعریف می‌شود که $\alpha=\{\alpha_1,\alpha_2,\ldots,\alpha_r\}$ مجموعه عمل‌های اتوماتا، $\beta=\{\beta_1,\beta_2,\ldots,\beta_m\}$ مجموعه ورودی‌های اتوماتا، $F=\phi\times\beta\to\phi$ تابع تولید وضعیت جدید، $G=\phi\to\alpha$ تابع خروجی که وضعیت فعلی را به خروجی بعدی نگاشت می‌کند و $\phi(n)=\{\phi_1,\phi_2,\ldots,\phi_k\}$ مجموعه وضعیت‌های داخلی اتوماتا در لحظه n می‌باشد.

اگر نگاشت‌های F و G قطعی باشند، اتوماتا قطعی نامیده می‌شود و در چنین حالتی با داشتن وضعیت اتوماتا و ورودی، می‌توان وضعیت بعدی و خروجی اتوماتا را به طور قطعی تعیین نمود. در حالتی که نگاشت‌های F و G تصادفی باشند، اتوماتا تصادفی نامیده می‌شود.

اتوماتای یادگیر با ساختار متغیر به صورت چهارتایی $\{\alpha,\beta,P,T\}$ تعریف می‌شود که $\alpha=\{\alpha_1,\alpha_2,\ldots,\alpha_r\}$ مجموعه عمل‌های اتوماتا، $\beta=\{\beta_1,\beta_2,\ldots,\beta_m\}$ مجموعه ورودی‌ها و $P=\{p_1,p_2,\ldots,p_r\}$ بردار احتمال انتخاب هر یک از عمل‌هاست که در آن احتمال انتخاب عمل iام با p_i متناظر است. در تعریف فوق T الگوریتم یادگیری است که بردار احتمال انتخاب‌ها را با توجه به عمل انتخاب‌شده و پاسخ محیط به روز می‌کند. ایده اصلی در الگوریتم‌های یادگیری این است که اگر اتوماتا عمل α_i را در زمان n انتخاب کند و پاسخ مطلوب از محیط دریافت نماید، احتمال $p_i(n)$ افزایش و سایر احتمال‌ها کاهش یابد. برای پاسخ نامطلوب $p_i(n)$ کاهش و سایر احتمال‌ها افزایش می‌یابد. در هر حال، تغییرات باید به گونه‌ای صورت پذیرد که بردار P در همه زمان‌ها بردار توزیع احتمال باقی بماند.

الگوریتم یادگیر خطی برای اتوماتای یادگیر سلولی با ساختار متغیر به صورت زیر تعریف می‌شود:

الف) به روز رسانی احتمال‌ها در حالت پاسخ مطلوب

$$p_i(n+1)=p_i(n)+a[1-p_i(n)]$$
$$p_j(n+1)=(1-a)p_j(n)\quad,\quad\forall j:j\neq i \qquad (1)$$

ب) به روز رسانی احتمال‌ها در حالت پاسخ نامطلوب

$$p_i(n+1)=(1-b)p_i(n)$$
$$p_j(n+1)=\frac{b}{r-1}(1-b)p_j(n)\quad,\quad\forall j:j\neq i \qquad (2)$$

پارامترهای a و b در روابط بالا به ترتیب پارامترهای پاداش و جریمه هستند. با توجه به مقادیر a و b در روابط فوق، چند حالت را می‌توان در نظر گرفت. اگر مقادیر a و b برابر باشند، اتوماتای یادگیر L_{R-P}[3]، زمانی که b مساوی با صفر باشد، اتوماتای یادگیر L_{R-I}[4]، اگر مقدار a به اندازه کافی بزرگ‌تر از b باشد، اتوماتای یادگیر $L_{R\varepsilon P}$[5] و همچنین زمانی که پارامتر پاداش صفر فرض شود، اتوماتای یادگیر L_{I-P}[6] نامیده می‌شود [۳] و [۹].

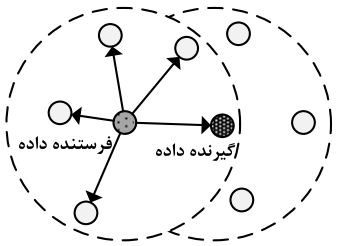
الگوریتم پیشنهادی در این مقاله از مدل اتوماتای یادگیر با ساختار متغیر و الگوریتم یادگیری فقط جریمه‌دهنده یعنی L_{I-P} بهره می‌گیرد.

۳- الگوریتم پیشنهادی

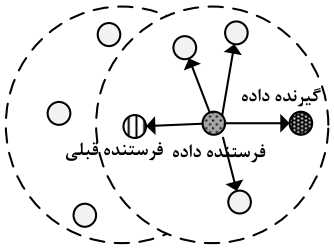
الگوریتم‌های مسیریابی در شبکه‌های حسگر بی‌سیم که از اتوماتای یادگیر بهره می‌برند را از جهت نحوه گزینش مسیر جهت ارسال داده می‌توان به دو دسته کلی تقسیم کرد: دسته اول همواره مسیر با بیشترین احتمال را انتخاب می‌کند و بر اساس معیاری به این مسیر پاداش یا جریمه می‌دهد. در دسته دوم مسیر به صورت تصادفی و بر اساس احتمالات فعلی در جدول مسیرها انتخاب می‌شود و لزوماً همواره مسیر با احتمال بیشتر انتخاب نمی‌شود.

دسته اول: در دسته اول همواره مسیر با بیشترین احتمال انتخاب می‌شود. با هر بار پاداش‌دادن به مسیر با احتمال بیشتر، احتمال انتخاب مسیر فعلی بیشتر و سایر مسیرها کم و کمتر می‌شود. اشکالی که می‌توان به این شیوه وارد دانست مربوط به زمانی است که وضعیتی به وجود آید که دیگر مسیر با بیشترین احتمال، مناسب ارسال داده نباشد (مثلاً سطح انرژی آن مسیر نسبت به میانگین همسایه‌ها کاهش پیدا کند) لذا تعداد بیشتری بسته پنالتی لازم است تا احتمال انتخاب این مسیر کاهش و سایر مسیرها افزایش یابد. ارسال و دریافت همین بسته‌های پنالتی نیز به نوبه خود موجب اتلاف انرژی گره‌ها می‌شود. اکثر الگوریتم‌هایی که در بخش مقدمه معرفی شد در این دسته قرار می‌گیرند.

1. Penalty Based Learning Automata for Energy Aware Routing
2. Penalty
3. Reward Penalty
4. Reward Inaction
5. Reward Epsilon Penalty
6. Inaction Penalty



شکل ۲: ارسال داده و به روز کردن سطح انرژی گره فرستنده در همسایه‌ها.



شکل ۳: پاداش یا جریمه‌دهی به گره فرستنده قبلی.

دسته دوم: در دسته دوم مسیرها به صورت تصادفی و بر اساس احتمال انتخاب آنها در جدول مسیرها گزینش می‌شوند. اشکالی که بر این روش می‌توان وارد نمود این است که ممکن است مسیرهای طولانی‌تری جهت ارسال داده‌ها انتخاب شوند و این مسأله خود موجب افزایش تأخیر در ارسال داده و همچنین اتلاف انرژی گره‌ها می‌شود. الگوریتم‌های LABER و DEEB در این دسته جای دارند.

الگوریتم پیشنهادی با هدف بهبود کارایی الگوریتم‌های دو دسته پیشین طراحی شده و در آن همواره مسیر با بیشترین احتمال انتخاب می‌شود و در صورت لزوم بسته‌های پنالتی به گره‌های فرستنده ارسال می‌شود تا بسته‌های داده‌ای از مسیرهای بهینه دیگر ارسال شود. الگوریتم پیشنهادی بر پایه دو الگوریتم BEAR از دسته اول و LABER از دسته دوم و با هدف بهبود آنها طراحی شده و از این رو در ادامه این دو الگوریتم اندکی دقیق‌تر بررسی می‌شوند.

۳-۱ الگوریتم LABER

الگوریتم LABER از مفهوم اتوماتای یادگیر برای مسیریابی در شبکه‌های حسگر با در نظر گرفتن تعادل بار ترافیک شبکه استفاده می‌کند. این الگوریتم از دو فاز اصلی تشکیل شده است: ۱) فاز ساخت جداول مسیریابی و ۲) فاز مسیریابی و یادگیری.

فاز ساخت جداول مسیریابی: در این فاز گره مقصد عمل غرقه‌سازی را انجام می‌دهد تا جداول مسیرها در همه گره‌ها ساخته شود. گره‌های دیگر که بسته مربوط به غرقه‌سازی را دریافت می‌کنند، اطلاعات این بسته را در جداول مسیرهای خود وارد می‌کنند و همچنین احتمالات اولیه برای انتخاب مسیرها از رابطه زیر محاسبه می‌شود

$$\forall_i : i \le m$$

$$p_i = h\frac{\frac{1}{numhop_i}}{\sum_{i=0}^{m}\frac{1}{numhop_i}} + (1-h)\frac{enevgylevel_i}{\sum_{i=0}^{m} enevgylevel_i} \qquad (3)$$

در این رابطه i تعداد بسته‌های دریافتی، $numhop_i$ تعداد گره‌های میانی بسته iام، $enevgylevel_i$ سطح انرژی گره ارسال‌کننده بسته iام و h یک عدد ثابت در بازه [۰,۱] می‌باشد.

فاز مسیریابی و یادگیری: در فاز دوم گره مبدأ یکی از مسیرهای موجود در جدول مسیرهای خود را بر اساس احتمال انتخاب مسیرها برمی‌گزیند و بسته داده را از طریق آن ارسال می‌کند. رویه انتخاب به این صورت است که احتمال انتخاب مسیرهای با احتمال بالاتر، بیشتر است یعنی لزوماً همیشه مسیر با بیشترین احتمال انتخاب نخواهد شد. هر گره‌ای که بسته داده‌ای دریافت می‌کند، بسته تأیید[1] آن که حاوی سطح انرژی خود است را به گره ارسال‌کننده می‌فرستد. گره ارسال‌کننده با دریافت بسته تأیید و مقایسه سطح انرژی آن با میانگین انرژی همسایه‌ها، به مسیر انتخاب‌شده پاداش می‌دهد و یا آن را جریمه می‌کند، بدین ترتیب احتمالات انتخاب عمل‌ها و یا همان مسیرها در هر گره در اتوماتای یادگیر به روز شده و مسیرهای مناسب با احتمال بیشتری انتخاب می‌شوند.

به منظور کاهش تعداد بسته‌های تأیید ارسالی، پارامتر P در این الگوریتم معرفی شده و هر گره با انتخاب هر مسیر، P بسته داده‌ای روی آن ارسال می‌کند. همچنین هر گره میانی با دریافت P بسته داده‌ای تنها یک بسته تأیید به گره فرستنده ارسال می‌کند.

۳-۲ الگوریتم BEAR

در الگوریتم BEAR فاز غرقه‌سازی مشابه الگوریتم LABER داریم، با این تفاوت که پارامتر h در (۳) برابر مقدار ثابت ۰/۵ در نظر گرفته شده است. فاز مسیریابی به این صورت است که گره مبدأ یا هر گره میانی برای ارسال بسته‌ای که دریافت کرده در جدول همسایه‌ها جستجو می‌کند و همسایه با بیشترین احتمال را انتخاب می‌کند. همچنین تفاوتی که این الگوریتم با الگوریتم‌های پیشین دارد این است که سطح انرژی هر گره در بسته داده قرار می‌گیرد و ارسال می‌شود و از این جهت که بسته‌ای مجزا برای این منظور در نظر گرفته نشده، سربار کمتری دارد.

ارسال بسته داده‌ای به این صورت انجام می‌شود که پس از انتخاب همسایه مورد نظر جهت ارسال داده، بسته داده‌ای که سطح انرژی گره ارسال‌کننده را نیز در خود دارد، به همه همسایه‌ها ارسال می‌شود. گره‌های همسایه پس از دریافت بسته داده، فقط سطح انرژی گره ارسال‌کننده را در جدول مسیرها به روز می‌کنند. از میان گره‌های همسایه یکی از گره‌ها فرستنده قبلی بسته است که فرستنده قبلی از بسته داده‌ای به منظور پاداش یا جریمه‌دادن و به روز رسانی احتمال انتخاب مسیرها استفاده می‌کند. شمای کلی ارسال بسته داده‌ای و به روز کردن احتمال مسیرها در شکل‌های ۲ و ۳ آمده است.

در این الگوریتم ملاکِ خوبیِ یک مسیر با (۴) سنجیده می‌شود و از عددی که از این رابطه به دست می‌آید جهت پاداش یا جریمه‌دهی استفاده می‌گردد

$$A = \frac{Energy_i}{Avgenergy} + \frac{HopCount_i}{AvgHopCount} \qquad (4)$$

اگر $A < 2$، جریمه با پارامتر β، اگر $A = 2$، جریمه با پارامتر $\beta/2$، اگر $2 < A < 2.2$، پاداش با پارامتر $\alpha/2$ و اگر $A > 2.2$، پاداش با پارامتر α می‌باشد.

۳-۳ معرفی الگوریتم پیشنهادی (PLAER)

فاز اولیه این الگوریتم از گره چاهک و با غرقه‌سازی شبکه با بسته‌های FLOOD آغاز می‌شود. فیلدهای اطلاعاتی این بسته شامل شماره گره، سطح انرژی و تعداد همسایه تا گره مقصد می‌باشد. بسته FLOOD در

1. Ack

جدول ۱: پارامترهای شبیه‌سازی.

انرژی اولیه هر گره	۰٫۱۸ mAh
ماکزیمم توان ارسال	۱۰۰ mw
فرکانس کاری	۲٫۴۱۲+۹ Hz
محدوده ارسال	۵۰ m
جریان حالت خواب گره‌ها	۰٫۰۲ mA
جریان حالت دریافت	۱۶٫۴ mA
جریان حالت ارسال	۱۷ mA
پارامتر h	۰٫۵
پارامتر جریمه در اتوماتا	۰٫۱
پارامتر φ (درصد)	۵۰
فاصله زمانی ارسال داده در گره‌ها	۱۰ s

هر مرحله به همه همسایه‌ها ارسال می‌شود و ارسال بسته‌ها در هر گره زمانی متوقف می‌شود که از یک همسایه دو بار بسته دریافت شود. هر گره با دریافت این بسته یک عمل به مجموعه اعمال اتوماتای خود اضافه می‌کند و احتمال اولیه برای انتخاب عمل‌ها یا همان مسیرها از (۵) محاسبه می‌شود

$$\forall_i : i \le m$$

$$p_i = h\frac{\frac{1}{\overline{numhop_i^2}}}{\sum_{i=0}^{m}(\frac{1}{numhop_i})^2} + (1-h)\frac{enevgylevel_i}{\sum_{i=0}^{m} enevgylevel_i} \qquad (۵)$$

پارامترهای فرمول بالا و شمای کلی فرمول در بخش ۳-۱ مربوط به شرح الگوریتم LABER معرفی شده است. تفاوتی که در این فرمول نسبت به فرمول مربوط به احتمالات اولیه در LABER مشاهده می‌شود، استفاده از توان دوم پارامتر $numhop$ است. این طراحی با هدف افزایش حساسیت پارامتر $numhop$ که در حقیقت فاصله همسایه‌ها تا گره مقصد می‌باشد، انجام شده است.

با اجرای فاز غرقه‌سازی، جداول مسیریابی در هر گره ساخته می‌شود. هر عنصر جدول متناظر با یک عمل مربوط به اتوماتای یادگیر است که شامل چهار فیلد اطلاعاتی شماره گره بعدی جهت مسیریابی، سطح انرژی مربوط به آن گره، احتمال انتخاب آن و تعداد گره‌های میانی از آن گره تا رسیدن به مقصد می‌باشد.

فاز مسیریابی در این الگوریتم مشابه BEAR و با جستجو در جدول مسیرها و انتخاب مسیر با بیشترین احتمال صورت می‌گیرد. بسته‌های اطلاعاتی از گره‌هایی که رخدادی را مشاهده کنند تولید می‌شود. در هر گره مسیرهای با احتمال بیشتر برای ارسال داده انتخاب می‌شود تا بسته مورد نظر به گره مقصد برسد. همچنین بر خلاف LABER که در آن به ازای ارسال هر (یا چند) بسته داده‌ای یک بسته تأیید به منظور پاداش‌دهی و جریمه داشتیم، در این الگوریتم تنها یک نوع بسته کنترلی یعنی جریمه داریم.

در این الگوریتم هر بسته داده‌ای که ارسال می‌شود، میانگین سطح انرژی همسایه‌ها نیز در آن گنجانده می‌شود. در حقیقت با این طراحی مقایسه سطح انرژی گره گیرنده با میانگین سایر همسایه‌ها در همان گره گیرنده انجام می‌شود و گره فرستنده منتظر دریافت بسته‌ای برای این منظور نیست. این میانگین انرژی با سطح انرژی گیرنده مقایسه می‌شود، در صورتی که سطح انرژی گره گیرنده کمتر از φ درصد میانگین انرژی همسایه‌ها باشد، یک بسته پنالتی تولید می‌شود که حاوی سطح انرژی گره تولیدکننده آن است. بسته پنالتی به همه همسایه‌ها ارسال می‌شود و از میان همسایه‌ها، فرستنده بسته داده با دریافت بسته پنالتی مسیر مربوط به آن را جریمه می‌کند. همچنین همه همسایه‌ها سطح انرژی مربوط به آن گره را در جدول مسیرهای خود به روز می‌کنند.

به منظور شناسایی گره‌های جدید و یا از کار افتادن گره‌های فعلی شبکه، فاز اولیه الگوریتم یعنی غرقه‌سازی شبکه با فاصله زمانی مشخصی اجرا می‌شود. میزان این فاصله زمانی به نوع و تغییرات توپولوژی شبکه بستگی دارد.

۴- شبیه‌سازی و تحلیل نتایج

الگوریتم‌های LABER، BEAR و الگوریتم PLAER به منظور شبیه‌سازی و مقایسه در نرم‌افزار OMNET++ [۱۰] پیاده‌سازی شده‌اند. شبیه‌سازی در محیط شبکه با تعداد گره‌های ۱۰۰، ۲۲۵ و ۴۰۰ گره و با توپولوژی مش[1] انجام شده و محدوده انتقال هر گره برابر ۵۰ متر در نظر گرفته شده است. برای لایه MAC در گره‌ها از پروتکل LMAC [۱۱] که از زمان‌بندی به خواب رفتن گره‌ها بهره می‌برد، استفاده شده و سایر پارامترهای شبیه‌سازی در جدول ۱ آمده است.

با طراحی دو سناریوی شبیه‌سازی، پروتکل‌ها از جهت میزان انرژی باقیمانده، تأخیر در رسیدن بسته‌ها به مقصد، تعداد بسته‌های کنترلی ارسالی، طول عمر شبکه و تعداد بسته‌های داده دریافتی در مقصد تا پایان عمر شبکه مقایسه شده‌اند.

سناریوی اول: تعداد ۲۰۰ بسته داده‌ای از ۴ گره شبکه به سمت گره مقصد که در مرکز شبکه قرار دارد به صورت تناوبی و با فاصله زمانی ۱۰ ثانیه فرستاده می‌شود. میانگین درصد انرژی باقیمانده گره‌ها و میانگین تأخیر دریافت بسته‌ها در گره مقصد در شکل‌های ۴ و ۵ آمده است.

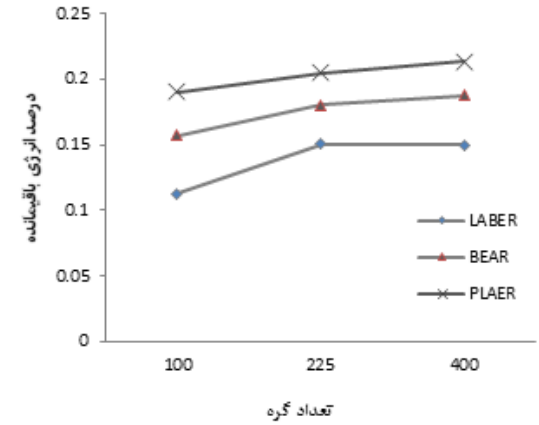
در الگوریتم LABER از آنجایی که به ازای ارسال هر (چند) بسته داده‌ای یک بسته تأیید به فرستنده ارسال می‌شود از نظر درصد انرژی باقیمانده گره‌ها در رتبه آخر قرار دارد. در پروتکل BEAR مفهوم بسته کنترلی حذف شده ولی از آنجایی که در هر مرحله هر بسته داده به همه همسایه ارسال می‌شود- در حقیقت دریافت در همه گره‌های همسایه انجام می‌شود- از نظر مصرف انرژی در جایگاه دوم قرار دارد. اما الگوریتم PLAER از آنجایی که فقط بسته‌های جریمه و در موارد مشخصی تولید می‌کند، از نظر مصرف انرژی نسبت به دو الگوریتم دیگر برتری دارد.

در شکل ۵ میانگین تأخیر دریافت بسته‌ها در مقصد مقایسه شده است. دو الگوریتم BEAR و PLAER تقریباً در یک جایگاه قرار دارند اما نکته‌ای که در مورد الگوریتم LABER لازم به یادآوری است، این که از آنجایی که انتخاب مسیرها یا همان عمل‌های اتوماتای یادگیر به صورت غیر قطعی و بر اساس احتمال انجام می‌گیرد، تحت شرایطی مسیرهای طولانی‌تری برای رسیدن بسته‌ها به مقصد انتخاب می‌شوند و از این جهت میانگین تأخیر دریافت بسته‌ها بیشتر است.

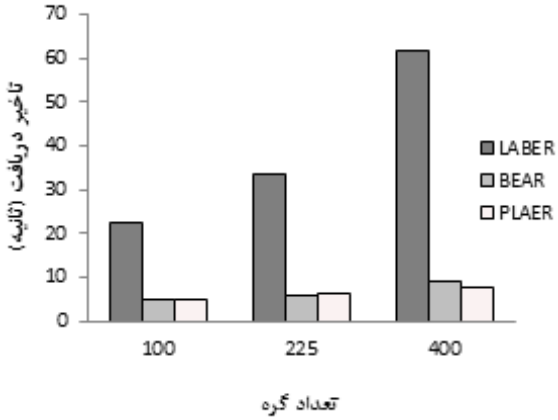
سناریوی دوم: در این سناریو تعداد بسته‌های ارسالی به سمت گره مقصد محدود نشده و تا زمان به اتمام رسیدن انرژی اولین گره شبکه ادامه پیدا می‌کند. لازم به ذکر است در این مقاله طول عمر شبکه برابر مدت زمان تمام‌شدن انرژی اولین گره شبکه در نظر گرفته شده است.

در شکل ۶ تعداد بسته‌های کنترلی ارسالی تا پایان عمر شبکه در الگوریتم‌های LABER و PLAER مقایسه شده و از آنجایی که BEAR بسته کنترلی ندارد در مقایسه ظاهر نشده است.

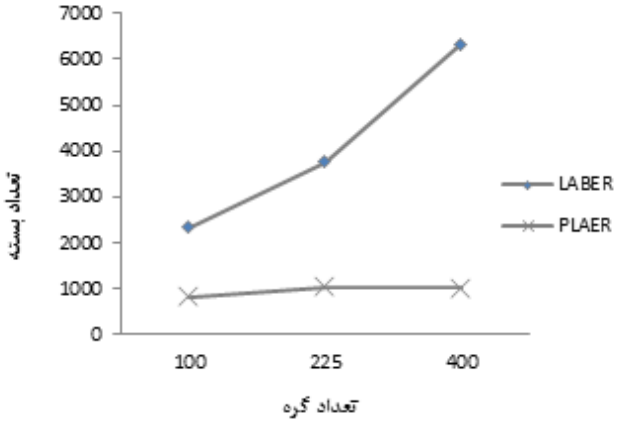
1. Mesh



شکل ۴: نمودار میانگین درصد انرژی باقیمانده گره‌ها.

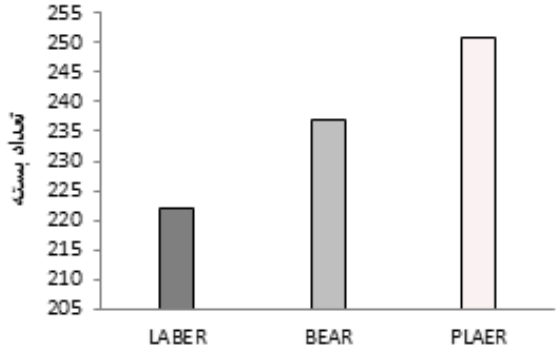


شکل ۵: نمودار میانگین تأخیر دریافت بسته‌ها در مقصد.

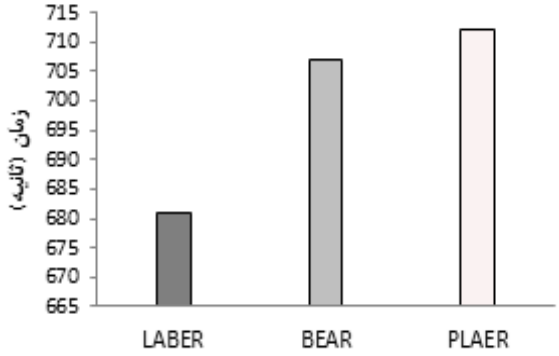


شکل ۶: نمودار تعداد بسته‌های کنترلی ارسالی تا پایان عمر شبکه.

الگوریتم PLAER نسبت به LABER از نظر تولید و ارسال بسته‌های کنترلی به طور محسوسی بهینه‌تر عمل می‌کند. این مسأله از اینجا ناشی می‌شود که در LABER به ازای ارسال هر (یا P) بسته داده‌ای یک بسته کنترلی به منظور پاداش و یا جریمه‌دهی به مسیر انتخابی ارسال می‌شود، ولی در PLAER بسته‌های کنترلی فقط بسته‌هایی از نوع جریمه هستند که این بسته‌ها نیز تنها در موارد مشخص تولید و ارسال می‌شوند. نکته دیگری که در شکل مشهود می‌باشد، این است که تعداد بسته‌های کنترلی در الگوریتم LABER با افزایش تعداد گره‌های شبکه به صورت خطی افزایش می‌یابد. این مسأله نیز به این علت است که در سناریوی طراحی‌شده، بسته‌های داده از گره‌های مرزی به سمت مرکز شبکه ارسال می‌شوند و با افزایش تعداد گره‌های شبکه، مسیر ارسال داده نیز طولانی‌تر



شکل ۷: نمودار تعداد بسته‌های داده دریافتی تا پایان عمر شبکه.



شکل ۸: نمودار طول عمر شبکه.

خواهد شد. با طولانی‌تر شدن مسیر، در الگوریتم LABER تعداد بسته‌های کنترلی نیز به صورت خطی افزایش خواهد یافت و همان گونه که در شکل ۶ نیز مشخص است، این رفتار در الگوریتم PLAER دیده نمی‌شود.

همان طور که در شکل‌های ۷ و ۸ مشاهده می‌شود، طول عمر شبکه و همچنین تعداد بسته‌های داده ارسالی در الگوریتم پیشنهادی نسبت به دو الگوریتم دیگر افزایش پیدا کرده است.

در الگوریتم PLAER بسته‌ها با تأخیر کمتر و با مصرف انرژی پایین‌تر و با ارسال بسته‌های کنترلی کمتر نسبت به دو الگوریتم دیگر ارسال می‌شوند، در نتیجه با ظرفیت انرژی یکسان امکان ارسال تعداد بیشتری بسته داده نسبت به دو الگوریتم دیگر وجود دارد و بدین ترتیب طول عمر شبکه نیز افزایش پیدا می‌کند.

۵- نتیجه‌گیری

در این مقاله الگوریتم مسیریابی PLAER جهت کاهش مصرف انرژی در شبکه‌های حسگر بی‌سیم معرفی گردید. ایده اصلی الگوریتم بر پایه بهره‌گیری از اتوماتای یادگیر در هر گره و شیوه جریمه‌دهی بنا شده و بدین منظور از مدل اتوماتای یادگیر با ساختار متغیر و فقط جریمه‌دهنده استفاده شده است. بر خلاف دیگر الگوریتم‌های معرفی‌شده، مقایسه سطح انرژی در این الگوریتم در گره گیرنده انجام می‌شود و در صورت نیاز بسته‌های جریمه تولید و به فرستنده ارسال می‌شود و بدین ترتیب و با به کارگیری این شیوه بسته‌های کنترلی که ارسال می‌شود در مقایسه با روش‌های دیگر به مراتب کمتر است. کاهش تعداد بسته‌های کنترلی و افزایش تعداد بسته‌های داده ارسالی منجر به افزایش طول عمر شبکه در این الگوریتم شده است. الگوریتم PLAER به همراه الگوریتم‌های LABER و BEAR در نرم‌افزار OMNET++ پیاده‌سازی و شبیه‌سازی شد و نتایج شبیه‌سازی از بهبود در مصرف انرژی و افزایش طول عمر

شبکه در این الگوریتم نسبت به دو الگوریتم مطرح‌شده حکایت می‌کند.

در پایان این مقاله به نظر می‌رسد می‌توان از مفهوم جریمه‌دهی در اتوماتای یادگیر و مدل اتوماتای یادگیر فقط جریمه‌دهنده در کاربردهای دیگر نظیر مسیریابی روبات‌ها در محیط با موانع، سامانه‌های مسیریاب در ترافیک، الگوریتم‌های تخصیص کانال در شبکه و ... بهره برد.

مراجع

[1] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292-2330, Apr. 2008.

[2] G. Anastasi, M. Conti, M. Di Francesco, and A. Passarella, "Energy conservation in wireless sensor networks: a survey," *Ad Hoc Networks*, vol. 7, no. 3, pp. 537-568, May 2009.

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

[4] M. Ankit, M. Arpit, T. J Deepak, R. Venkateswarlu, and D. Janakiram, "TinyLap: a scalable learning automata-based energy aware routing protocol for sensor networks," *Communicated to IEEE Wireless and Communications and Networking Conf.*, Las Vegas, NV, USA. 2006.

[5] S. M. Abolhasani, M. R. Meybodi, and M. Esnaashari, "LABER: a learning automata based energy - aware routing protocol for sensor networks," in *Proc. of the Third Information and Knowledge Technology*, *IKT'07*, 9 pp., Mashad, Iran, 27-29 Nov. 2007.

[6] A. H. Fathy Navid and H. Haj Seyed Javadi, "ICLEAR: energy aware routing protocol for WSN using irregular cellular learning automata," in *Proc. IEEE Symp. on Industrial Electronics and Applications, ISIEA'09*, vol. 1, pp. 463-468, 4-6 Oct. 2009.

[7] E. Ahvar and M. Fathy, "BEAR: a balanced energy - aware routing protocol for wireless sensor networks," *Wireless Sensor Network*, vol. 2, no. 10, pp. 793-800, 2010.

[8] J. A. Torkestani, "Backbone formation in wireless sensor networks," *Sensors and Actuators A: Physical*, vol. 185, pp. 117-126, Oct. 2012.

[9] M. A. L. Thathachar and P. S. Sastry, "Varieties of learning automata: an overview," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Trans. on*, vol. 32, no. 6, pp. 711-722, 2002.

[10] A. Varga, "The OMNeT++ discrete event simulation system," in *Proc. of the European Simulation Multiconference, ESM'01*, pp. 319-324, Prague, Czech Republic, Jun. 2001.

[11] L. F. W. van Hoesel and P. Havinga, "A lightweight medium access protocol (LMAC) for wireless sensor networks: reducing preamble transmissions and transceiver state switches," 2004.

مرتضی پرویزی عمران در سال ۱۳۸۸ مدرک کارشناسی مهندسی کامپیوتر (نرم افزار) خود را از دانشکده فنی دانشگاه تهران و در سال ۱۳۹۲ مدرک کارشناسی ارشد مهندسی کامپیوتر (الگوریتم‌ها و محاسبات) خود را از دانشکده علوم مهندسی دانشگاه تهران دریافت نمود. زمینه‌های علمی مورد علاقه ایشان شامل موضوعاتی مانند: الگوریتم‌های موازی، سیستم‌های توزیعی، شبکه‌های کامپیوتری، سیستم‌های یادگیری و پایگاه داده است.

علی معینی کارشناسی ارشد پیوسته برق خود را در سال ۱۳۶۵ از دانشکده فنی دانشگاه تهران اخذ کرده است. دکتری خود را ۱۳۷۵ از دانشگاه ساسکس انگلستان به پایان رسانده است و هم‌اکنون دانشیار دانشکده علوم مهندسی پردیس دانشکده‌های فنی دانشگاه تهران می‌باشد. زمینه‌های تحقیقاتی مورد علاقه ایشان عبارتند از: سیستم‌های دینامیکی و هوشمند، الگوریتم‌های تصادفی، محاسبات برخط و تحلیل رقابتی، روش‌های رسمی و کاربردهای آنها در شبکه‌ها، مهندسی نرم افزار، مدیریت فناوری اطلاعات، و بیوانفورماتیک است.

حمید حاج سید جوادی مدارک کارشناسی، کارشناسی ارشد و دکتری خود را از دانشگاه صنعتی امیرکبیر اخذ کرده و هم‌اکنون دانشیار دانشگاه شاهد در گروه ریاضی و علوم کامپیوتر می‌باشد. زمینه‌های تحقیقاتی مورد علاقه ایشان شامل موضوعاتی نظیر: جبر و امنیت، شبکه های حسگر بی‌سیم و نظریه الگوریتم ها می‌باشد.