

بهبود الگوریتم مسیریابی چندمسیری مجزای ناحیه ای ZD-MPDSR

برای شبکه‌های سیار موردی

نستوه طاهری جوان^۱، یسنا قنبری بیرگانی^۲ و مهدی دهقان^۳

^۱دانشگاه صنعتی امیرکبیر، دانشکده مهندسی کامپیوتر و فناوری اطلاعات، آزمایشگاه شبکه های سیار Ad-Hoc, nastoooh@aut.ac.ir,
^۲مدرس دانشگاه آزاد اسلامی، واحد بندر ماهشهر، دانشکده مهندسی کامپیوتر، yasna_gh@yahoo.com,
^۳دانشگاه صنعتی امیرکبیر، دانشکده مهندسی کامپیوتر و فناوری اطلاعات، آزمایشگاه شبکه های سیار Ad-Hoc, dehghan@aut.ac.ir

چکیده - برخی از الگوریتم‌های مسیریابی چندمسیری جهت کاهش تاخیر انتها به انتها و حتی متعادل کردن بار، برای ارسال اطلاعات به سمت مقصد به طور همزمان از چندین مسیر کشف شده استفاده می‌کنند. در این حالت برای افزایش تحمل پذیری خطا انتخاب مسیرهای مجزای گره‌ای یکی از بهترین گزینه‌ها می‌باشد. اما در شبکه‌های بی سیم ارسال اطلاعات از طریق مسیرهای کاملاً مجزا نیز از هم مستقل نیستند و بخاطر مسائل ذاتی شبکه‌های بی سیم و مکانیزمهای دسترسی به کانال مانند ارسال RTS و CTS، استفاده از دو مسیر کاملاً مجزا نیز به یکدیگر وابسته است و در بخش‌های زیادی از زمان، برخی از گره‌های میانی باید ارسال خود را به احترام گره‌هایی از مسیر مجاور متوقف کنند. برای حل این مشکل الگوریتم ZD-MPDSR از مسیرهای مجزای ناحیه‌ای به جای مسیرهای مجزای گره‌ای استفاده می‌کند. در این مقاله بهبودهای قابل ملاحظه‌ای در فرآیند کشف مسیر الگوریتم ZD-MPDSR جهت پیدا کردن مسیرهای مجزای گره‌ای بین مبدا و مقصد صورت گرفته است. کارآیی روش پیشنهادی در سناریوهای مختلف ارزیابی شده و پیشرفت قابل توجهی در کاهش سربار مسیریابی و کاهش تاخیر انتها به انتها نسبت به الگوریتم ZD-MPDSR به دست آمده است.

کلید واژه- شبکه‌های سیار موردی، الگوریتم‌های مسیریابی چندمسیری، مسیرهای مجزای ناحیه‌ای، الگوریتم ZD-MPDSR.

۱- مقدمه

الگوریتم‌ها عمل مسیریابی را به صورت چند مسیری انجام می‌دهند [4]. به این ترتیب که طی فرآیند کشف مسیر، همزمان چندین مسیر را کشف و ثبت می‌کنند. با این کار تعداد دفعات اجرای فرآیند زمانگیر کشف مسیر به طور چشم‌گیری کاهش می‌یابد که این مسئله اولین مزیت مسیریابی چندمسیری می‌باشد. از طرف دیگر با استفاده از مسیریابی چندمسیری تحمل پذیری در برابر خطا به طور موثری افزایش می‌یابد. اکثر الگوریتم‌های مسیریابی چندمسیری، پس از آنکه در فرآیند کشف مسیر چندین مسیر را از مبدا تا مقصد پیدا کردند، یکی از این مسیرها را به عنوان مسیر اصلی انتخاب کرده و ارسال اطلاعات را از طریق همین مسیر آغاز می‌کنند و سایر مسیرها را به عنوان جایگزین نگهداری کرده و در صورت خرابی مسیر اصلی، یکی از مسیرهای جایگزین را برای ارسال اطلاعات استفاده می‌کنند.

با کمی تامل در الگوریتم‌های مسیریابی چندمسیری می‌توان دریافت پس از اینکه در فرآیند کشف مسیر، چندین مسیر بین مبدا و مقصد پیدا شد، می‌توان ارسال اطلاعات به سمت مقصد را به طور همروند و از طریق چندین مسیر آغاز کرد. با استفاده از این مکانیزم‌ها می‌توان با تقسیم بار اطلاعات بین چندین مسیر، ترافیک را در شبکه متعادل کرد که این امر در مبحث کنترل ترافیک و ازدحام مورد توجه

یک شبکه سیار موردی شبکه‌ای است که در آن هیچ زیر ساخت، مسیریاب و ایستگاه ثابتی وجود ندارد [2]. در این شبکه‌ها، کلیه اعمال شبکه از قبیل مسیریابی توسط خود گره‌ها و با همکاری یکدیگر انجام می‌شود. خصوصیتی از قبیل قابلیت تحرک بسیار بالای گره‌ها و در نتیجه توپولوژی پویای شبکه، پهنای باند کم و حتی توان و انرژی محدود موجب پیچیدگی الگوریتم‌های مسیریابی در شبکه‌های موردی می‌شود. همه این خصوصیات باعث می‌شود تا بسیاری از ایده‌های موجود در الگوریتم‌های مسیریابی در شبکه‌های دیگر، برای شبکه‌های موردی کارایی و کاربردی نداشته باشند. با این حال الگوریتم‌های مسیریابی زیادی برای شبکه‌های موردی پیشنهاد شده‌اند [3] و برخی از آنها عمومیت و استقبال فراوانی پیدا کرده‌اند، که از این بین می‌توان به الگوریتم‌های مسیریابی DSR¹ و AODV² اشاره کرد. هر دوی این الگوریتم‌ها از رده برحسب تقاضا^۳ هستند، به این معنی که فرآیند کشف مسیر تنها هنگامی اجرا می‌شود که یک مبدا به یک مسیر تا مقصدی خاص نیاز داشته باشد.

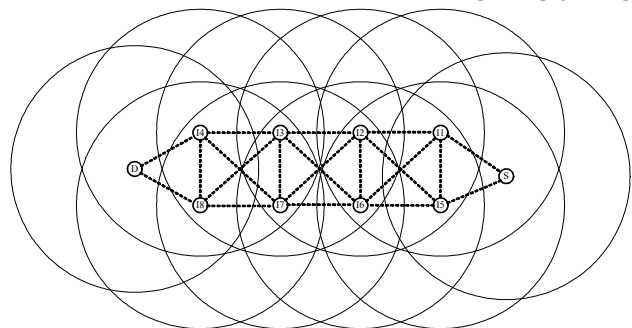
در بین الگوریتم‌های مسیریابی شبکه‌های موردی، برخی از

واقع می‌شود. در نهایت تقسیم و ارسال اطلاعات بین چندین مسیر مختلف، می‌تواند منجر به افزایش پهنای باند انتها به انتها و در نتیجه کاهش قابل توجه تاخیر شود.

در این حالت یکی از مهمترین مسائل، انتخاب مسیرهای مناسب برای ارسال همروند اطلاعات به سمت مقصد می‌باشد. یکی از ایده‌هایی که برای این کار از آن استفاده می‌شود، انتخاب مسیرهای مجزای گره‌ای بین مبدا و مقصد می‌باشد. با این کار تحمل پذیری خطا به طور قابل ملاحظه‌ای افزایش می‌یابد. به طوری که اگر مسیری که برای ارسال انتخاب می‌شوند، هیچ گره مشترکی با هم نداشته باشند، خراب شدن و از بین رفتن یک گره یا یک اتصال نهایتاً به شکستن یک مسیر می‌انجامد و تأثیری بر روی بیش از یک مسیر ندارد [4].

به این ترتیب شاید به نظر برسد به بالاترین حد کارایی و پایین ترین حد تاخیر انتها به انتها رسیده‌ایم، اما همانطور که می‌دانیم در شبکه‌های موردی مشکلات ایستگاه آشکار^۴ و ایستگاه پنهان^۵ وجود دارد که برای برطرف کردن آن‌ها پروتکل CSMA/CA^۶ پیشنهاد شده- است [5]. در استاندارد 802.11 از این پروتکل برای دستیابی به کانال استفاده می‌شود. در این پروتکل به خاطر تبادل پیامهای RTS^۷ و CTS^۸ بین گره‌ها، برخی از گره‌ها مجبور به سکوت و عدم ارسال اطلاعات می‌شوند که این مسأله تاخیر انتها به انتها را افزایش می‌دهد. به عنوان مثال شکل (۱) را در نظر بگیرید. این شکل یک شبکه فرضی را نشان می‌دهد که در آن فقط ده گره نمایش داده شده- است. در این شکل بُرد رادیویی هر گره مشخص شده‌است و خطوط نقطه چین وجود ارتباط مستقیم بین دو گره را نشان می‌دهند، به عبارت دیگر وجود نقطه چین بین دو گره خاص به این معناست که دو گره در بُرد رادیویی یکدیگر قرار دارند.

حال در این شبکه بین دو گره S و D، دو مسیر مجزای گره‌ای S-I1-I2-I3-I4-D و S-I5-I6-I7-I8-D وجود دارند که ارتباط و ارسال داده از یک مسیر، کاملاً مستقل از مسیر دیگر نیست. در این حالت تاخیر انتها به انتها هر یک از مسیرها به ترافیک مسیر دیگر نیز وابسته است که این امر به خاطر تبادل پیام‌های RTS و CTS بین گره‌های شبکه برای اجتناب از تصادم و رفع مشکلات ایستگاه پنهان و ایستگاه آشکار می‌باشد. در نتیجه برخی از ایستگاه‌های یک مسیر، مثلاً به خاطر دریافت CTS از یک گره در مسیرمقابل، باید فعلاً ارسال خود را به تعویق بیندازند.



شکل ۱: مسیرهای مجزای گره‌ای.

برای برطرف کردن این مشکل روش‌هایی پیشنهاد شده‌اند که در آن‌ها از آنتن‌های جهت دار استفاده می‌شود [6,7]. اما آنتن‌های جهت‌دار در بسیاری از موارد برای گره‌های سیار در دسترس نیستند. در [1] یک الگوریتم مسیریابی چندمسیری با عنوان ZD-MPDSR بر اساس الگوریتم مسیریابی [8] DSR پیشنهاد شده‌است. در این الگوریتم با استفاده از راهکاری مسیرهای مجزای ناحیه‌ای بین گره‌های مبدا و مقصد کشف می‌شوند و برای ارسال همروند اطلاعات به سمت مقصد از این مسیرها استفاده می‌شود. این الگوریتم با اینکه به بهبودهایی در زمینه کاهش تاخیر انتها به انتها دست یافته‌است، اما سربار مسیریابی آن بسیار بالاست، از طرفی فرآیند کشف مسیر آن نیز با تاخیر بالایی صورت می‌گیرد، که البته این تاخیر تا حدودی در فاز ارسال اطلاعات جبران می‌شود.

در این مقاله فرآیند کشف مسیر در الگوریتم ZD-MPDSR به گونه‌ی موثری بهبود می‌یابد، به طوری که سربار و تاخیر حاصل از کشف مسیر در آن کاسته می‌شود.

ادامه این مقاله از بخش‌های زیر تشکیل شده است: در بخش دوم به مروری بر کارهای مرتبط با مسیریابی چندمسیری در شبکه‌های موردی از جمله الگوریتم ZD-MPDSR می‌پردازیم. در بخش ۳ جزئیات الگوریتم پیشنهادی را بررسی کرده و در بخش ۴ به بیان نتایج به دست آمده از شبیه‌سازی می‌پردازیم. در نهایت در بخش ۵ یک نتیجه اجمالی از کار صورت گرفته می‌گیریم.

۲- کارهای مرتبط

در راستای مسیریابی چند مسیری در شبکه‌های موردی کارها و تحقیقات بسیاری صورت گرفته است. در اکثر این کارها سعی بر این بوده که الگوریتم‌های تک مسیری موجود در شبکه‌های موردی -که کارایی مناسبی داشته‌اند- را با ایجاد تغییراتی به الگوریتم‌های چندمسیری تبدیل کنند. در [9] الگوریتمی تحت عنوان SMR^۹ پیشنهاد شده که در حقیقت بر اساس الگوریتم مسیریابی DSR بنا نهاده شده است. این الگوریتم سعی دارد در فرآیند کشف مسیر، مسیرهایی را پیدا کند که حداقل اشتراک را با هم دارند. برای این منظور مبدا یک بسته درخواست مسیر را برای همه همسایه‌های خود به صورت فراگیر ارسال می‌کند. در SMR بر خلاف DSR، گره‌های میانی همه بسته‌های درخواست مسیر تکراری را حذف نمی‌کنند، بلکه اگر یک بسته درخواست مسیر به طور تکراری ولی از یک اتصال دیگر به این گره رسیده باشد و همچنین تعداد گام آن از تعداد گام اولین بسته درخواست مسیر دریافت شده بزرگتر نباشد، این بسته را دوباره پخش می‌کنند. به این ترتیب بسته‌های درخواست مسیر بیشتری به مقصد می‌رسد. در این حالت مقصد در جواب اولین درخواست مسیری که دریافت کرد، یک بسته پاسخ مسیر به سوی مبدا بر می‌گرداند، زیرا این بسته قاعداً از طریق کوتاه‌ترین مسیر رسیده است. بعد از این مرحله، مقصد پس از دریافت بسته‌های درخواست مسیر دیگر، از بین آنها مسیرهای مجزای گره‌ای را انتخاب کرده و بسته پاسخ مسیر را از

طریق آن‌ها به سمت مبدا روانه می‌کند.

به عنوان یک مثال دیگر از الگوریتم‌های مسیریابی چندمسیری، در [10] روشی تحت عنوان AOMDV¹⁰ براساس الگوریتم مسیریابی AODV طراحی شده است. در الگوریتم AOMDV، طی فرآیند کشف مسیر، چندین مسیر مجزای اتصالی کشف می‌شوند. در این الگوریتم، هر گره به ازای هر مقصد، دو گام بعدی را نگهداری می‌کند. اگر در طول برقراری ارتباط، یکی از اتصال‌ها از بین رفت، گره میانی از گام بعدی جایگزین استفاده می‌کند. نقطه ضعف این الگوریتم این است که درحین ارتباط، اتصال با گام بعدی جایگزین نیز ممکن است از بین رود و هنگامی که به این گام بعدی نیاز است، اتصال با آن نیز از بین رفته باشد.

اما در راستای مسیریابی چندمسیری با استفاده از مفهوم مسیرهای مجزای ناحیه‌ای، با استفاده از آنتن‌های جهت‌دار یک روش در [6] پیشنهاد شده است. در این روش هر گره باید به‌طور متناوب اطلاعات مربوط به همسایه‌های خود را در یک جدول موسوم به AST درج کند. این اطلاعات باید شامل قدرت سیگنال رادیویی یک گره تا یک همسایه و در زاویه‌ای خاص باشد. دقت کنید برای این کار حتماً به آنتن‌های رادیویی جهت‌دار نیاز داریم. پس از این مرحله، هنگامی که مبدا به یک مسیر تا مقصد نیاز داشت، ابتدا سعی می‌کند تمام مسیرهای مجزای گره‌ای تا مقصد را شناسایی کند، سپس از بین این مسیرها، مسیرهای مجزای ناحیه‌ای را انتخاب می‌کند. در این حالت مبدا می‌تواند محاسبات مربوط به شناسایی مسیرهای مجزای ناحیه‌ای را به‌طور متناوب تکرار کرده و تصمیم مسیریابی خود را به‌روز کند. لازم به ذکر است در [7] نیز درباره مزایا و معایب الگوریتم‌های چندمسیری با استفاده از آنتن‌های جهت‌دار بحث شده است.

الگوریتم ZD-MPDSR بر اساس الگوریتم مسیریابی DSR بنا شده است [1]. به طور کلی می‌توان گفت در الگوریتم ZD-MPDSR، مقصد سعی می‌کند از بین RREQ های دریافتی مسیرهای مجزای ناحیه‌ای را انتخاب کرده و بسته پاسخ مسیر را از طریق این مسیرها به سمت مبدا ارسال کند. برای کشف مسیرهای مجزای ناحیه‌ای بین مبدا و مقصد، یک فیلد جدید با عنوان ActiveNeighborCount و با مقدار اولیه صفر در سرآیند بسته‌های RREQ ایجاد می‌شود. این فیلد در واقع تعداد همسایه‌های فعال برای گره‌های موجود در یک مسیر را نشان می‌دهد. در اینجا منظور از همسایه‌های فعال گره‌هایی هستند که قبلاً همین RREQ را دریافت کرده‌اند و این احتمال وجود دارد که مبدا و مقصد برای تبادل اطلاعات بین خود، مسیری دیگر را که از آن گره می‌گذرد، نیز انتخاب کرده باشند، که در این صورت ارسال اطلاعات از طریق این دو مسیر به هم وابسته است. به علاوه برای پیاده سازی الگوریتم پیشنهادی، همه گره‌ها باید یک جدول با عنوان RREQ Seen نگهداری کنند، که در این جدول مشخصات RREQ های دریافت شده توسط هر گره ثبت می‌شود.

در الگوریتم ZD-MPDSR هر یک از گره‌های میانی که یک

RREQ را دریافت کرد، ابتدا مشخصات آن را در جدول RREQ_Seen خود درج می‌کند، اما قبل از ارسال این بسته فقط از همسایه‌های خود پرس و جو می‌کند که "آیا شما قبلاً این RREQ را با این مشخصات دیده‌اید؟" و برای این منظور، یک بسته با عنوان RREQ_Query برای همسایه‌های خود ارسال می‌کند و مدت زمان مشخصی را با تنظیم کردن یک تایمر، منتظر بازگشت پاسخ همسایه‌ها می‌ماند. در این حالت همسایه‌ها پس از دریافت این پرسش، موظف هستند با جستجو در جدول RREQ_Seen پاسخ این سوال را برگردانند. پس از منقضی شدن زمان تایمر، این گره به تعداد همسایه‌هایی که پاسخ مثبت می‌دهند، محتوای فیلد ActiveNeighborCount را در بسته RREQ افزایش می‌دهد و آنگاه آن را برای همه ارسال می‌کند.

در این حالت وقتی مقصد RREQ های مختلف را دریافت کرد، شروع به انتخاب مسیرهای مجزای گره‌ای می‌کند، سپس در بین مسیرهای انتخاب شده، محتوای فیلد ActiveNeighborCount را بررسی کرده و مسیرهایی را انتخاب می‌کند که محتوای فیلد ActiveNeighborCount در آنها کمترین باشد. در واقع مقصد با انتخاب مسیرهایی که محتوای این فیلد در آنها کمتر از بقیه است، سعی می‌کند مسیرهای مجزای ناحیه‌ای را انتخاب کند. سپس بسته پاسخ مسیر را از طریق مسیرهای انتخاب شده به سمت مبدا ارسال می‌کند. مبدا نیز به محض دریافت اولین بسته پاسخ مسیر، ارسال اطلاعات را از طریق این مسیر شروع می‌کند و پس از دریافت بسته های پاسخ مسیر بعدی، بر اساس معیارهای موردنظر در زمینه متعادل کردن بار، به تقسیم بار بین مسیرهای موجود می‌پردازد.

۳- الگوریتم پیشنهادی

الگوریتم پیشنهادی در واقع بهبود یافته الگوریتم ZD-MPDSR می‌باشد. همانطور که در بخش کارهای مرتبط مرور شد، روش ZD-MPDSR بر اساس الگوریتم پایه DSR عمل می‌کند. در این الگوریتم جهت ارسال همروند اطلاعات به سمت مقصد چندین مسیر مجزای ناحیه‌ای بین مبدا و مقصد کشف می‌شود. در این قسمت، ابتدا در بخش ۳-۱ اشتراکات بین الگوریتم پیشنهادی و الگوریتم ZD-MPDSR و در بخش ۳-۲ تغییرات مورد نیاز در الگوریتم ZD-MPDSR به اجمال بررسی می‌شوند. در نهایت در بخش ۳-۳ روال کار الگوریتم پیشنهادی به تفصل به همراه مثال و شبه کد عملکرد گره‌ها بیان می‌شود.

۳-۱- اشتراکات الگوریتم پیشنهادی و ZD-MPDSR

الگوریتم پیشنهادی از برخی جهات با الگوریتم ZD-MPDSR اشتراکات و تشابهات بسیاری دارد. در این بخش برخی از این اشتراکات مرور می‌شوند.

همانند الگوریتم ZD-MPDSR، در الگوریتم پیشنهادی گره های میانی نیازی به نگهداری Route Cache ندارند. در واقع در این دو الگوریتم، بر خلاف الگوریتم DSR، از یک طرف اجازه داده می‌شود که

به این ترتیب هنگام پخش شدن بسته‌های RREQ هیچ تاخیری به فرآیند کشف مسیر تحمیل نمی‌شود و در واقع بسته‌های RREQ طبق روال استاندارد خود به سمت مقصد حرکت می‌کنند. در اینجا باید دقت کرد در الگوریتم ZD-MPDSR، هنگامی که یک گره یک RREQ را دریافت می‌کند، قبل از بازپخش کردن آن قدری تأمل می‌کند تا با ارسال بسته‌های RREQ_Query از همسایه‌های خود پرس و جو کند و منتظر بازگشت پاسخ این پرس و جو و بسته‌های RREQ_Query_Reply می‌ماند.

بعد از این مرحله، بسته‌های RREQ به ترتیب به مقصد تحویل می‌شوند و مقصد به ازای هر کدام از RREQ ها، یک بسته RREP به سمت مبدا برمی‌گرداند. در هر کدام از بسته‌های RREP یک فیلد با عنوان ActiveNeighborCount تعبیه شده‌است که با مقدار اولیه صفر پر می‌شود. این بسته‌های RREP براساس اصول الگوریتم DSR از طریق مسیر معکوس RREQ ها به سمت مبدا برمی‌گردند. در واقع بسته‌های RREP مسیر حمل شده توسط RREQ را به طور معکوس در سرآیند خود به همراه دارند تا مسیر بازگشت به سمت مبدا را پیدا کنند.

جهت شمارش تعداد همسایه‌های فعال هر مسیر، بسته‌های RREP در راه بازگشت به مبدا، پس از رسیدن به هر گره میانی، مقدار Counter ذخیره شده در جدول RREQ_Seen آن گره به ازای بسته RREQ معادل خود را با فیلد ActiveNeighborCount خود جمع می‌کنند. به این ترتیب هنگامی که بسته RREP به مبدا رسید، تعداد همسایه‌های فعال مسیر خود را در فیلد ActiveNeighborCount به همراه دارد. در این حالت گره مبدا پس از دریافت اولین RREP با تنظیم یک تایمر، مدت زمان مشخصی را منتظر مابقی RREP ها می‌ماند و پس از منقضی شدن زمان تایمر، RREP های دریافتی را بر اساس مقدار ActiveNeighborCount آنها به صورت صعودی مرتب می‌کند و به تعداد مورد نیاز از ابتدای لیست چند مسیر را انتخاب کرده و ارسال همروند اطلاعات به سمت مقصد را از طریق آنها آغاز می‌کند. در واقع مسیرهایی را برای ارسال همروند انتخاب می‌کند که فیلد ActiveNeighborCount آنها کمترین باشد.

برای درک عملکرد گره‌ها در این الگوریتم، شبه‌کد عملکرد گره مقصد در شکل (۲)، شبه‌کد عملکرد گره مبدا در شکل (۳) و شبه‌کد عملکرد گره‌های میانی در شکل (۴) ارائه شده است.

۱. پس از دریافت بسته‌های درخواست مسیر، بر اساس الگوریتم مسیریابی خود، به ازای هر بسته درخواست مسیر، یک بسته پاسخ مسیر با مقدار اولیه ActiveNeighborCount برابر با صفر به سمت مبدا روانه کن.

شکل ۲: شبه‌کد عملکرد گره مقصد.

بسته‌های RREQ بیشتری به مقصد برسند و از طرف دیگر اجازه داده می‌شود که بسته‌های RREQ مسیر خود را تا انتها و تا گره مقصد طی کنند تا بتوانند تعداد همسایه‌های فعال یک مسیر را بدرستی شمارش کنند.

همانند الگوریتم ZD-MPDSR، در الگوریتم پیشنهادی نیز بسته‌های RREP به فیلد ActiveNeighborCount نیاز دارند. همانند الگوریتم ZD-MPDSR، گره‌های میانی کماکان به جداول RREQ_Seen نیاز دارند، البته باید کمی تغییر در جداول RREQ_Seen اعمال شود.

۲-۲ تفاوت‌های الگوریتم پیشنهادی و ZD-MPDSR

در این بخش تعدادی از تغییرات اعمال شده در الگوریتم ZD-MPDSR بررسی می‌شوند.

در الگوریتم پیشنهادی، برخلاف الگوریتم ZD-MPDSR بسته‌های RREQ به فیلد ActiveNeighborCount نیازی ندارند.

روال کشف مسیر در الگوریتم پیشنهادی به گونه‌ای است که برای شمارش تعداد همسایه‌های فعال نیازی به بسته‌های RREQ_Query و RREQ_Query_Reply نیست، که این امر به کاهش سربار مسیریابی و کاهش تاخیر کشف مسیر کمک بسیاری می‌کند.

برخلاف الگوریتم ZD-MPDSR، در الگوریتم پیشنهادی در واقع مبدا به جای مقصد مسیرها را برای ارسال همروند اطلاعات انتخاب می‌کند.

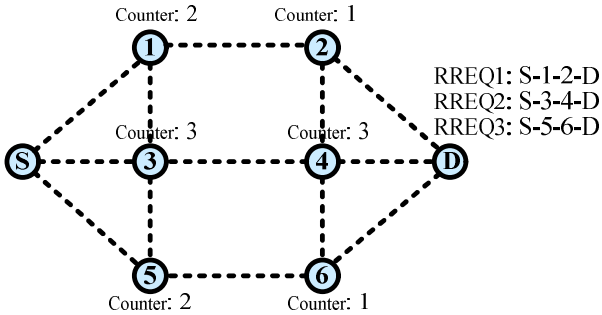
به جداول RREQ_Seen در گره‌های میانی یک فیلد با عنوان Counter جهت شمارش تعداد RREQ های دریافتی افزوده می‌شود.

۳-۳ روال کار الگوریتم

در الگوریتم‌های برحسب تقاضا، هنگامی که مبدا برای مقصدی خاص داده برای ارسال داشت، اما مسیری تا مقصد در اختیار نداشت، فرآیند کشف مسیر را اجرا می‌کند. در این حالت مبدا یک بسته درخواست مسیر (RREQ) را برای همه همسایه‌های خود به طور سیل-آسا ارسال می‌کند. در مسیریابی مبدا هر بسته گره‌هایی را که طی می‌کند، در سرآیند خود ذخیره می‌کند.

در الگوریتم پیشنهادی گره‌های میانی که بسته RREQ را دریافت می‌کنند، مشخصات آن را در جدول RREQ_Seen خود درج می‌کنند. در این جدول به ازای هر RREQ یک فیلد با عنوان Counter وجود دارد. هر گره باید تعداد دفعاتی که این بسته RREQ را دریافت کرده است، در این فیلد درج کند. برای این منظور هر گره هر بار که یک RREQ دریافت کرد، بدون توجه به اینکه این RREQ قابل بازپخش هست یا باید حذف شود، یک واحد به فیلد Count مربوط به این RREQ در جدول RREQ_Seen اضافه می‌کند. در واقع این فیلد به طور ضمنی تعداد همسایه‌های فعال این گره را برای این RREQ نشان می‌دهد.

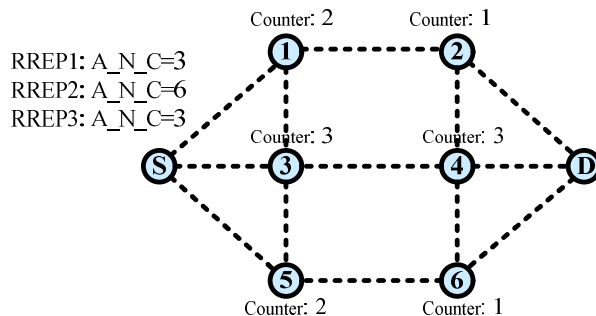
RREQ به طور فراگیر، فرآیند کشف مسیر را آغاز می‌کند. بسته‌های RREQ طبق اصول الگوریتم DSR راه خود را به سمت مقصد طی می‌کنند و در طول این پروسه، گره‌های میانی با دریافت هر RREQ فیلد Counter در جدول RREQ_Seen خود را به روز می‌کنند. شکل ۶ حالت این شبکه وقتی همه RREQ ها به مقصد می‌رسند را نشان می‌دهد. جهت ساده‌تر شدن شکل، از رسم مابقی RREQ ها خودداری شده‌است.



شکل ۶: نحوه دریافت RREQ ها توسط مقصد در یک شبکه فرضی.

همانطور که در شکل ۶ مشاهده می‌شود، در گره شماره ۱ شمارنده مربوط به این RREQ در جدول RREQ_Seen برابر ۲ می‌باشد. زیرا این گره پس از اینکه RREQ را از گره S دریافت کرد، از گره‌های ۲ و ۳ نیز دوباره همین RREQ را دریافت می‌کند، بنابراین مقدار شمارنده این RREQ را دوبار افزایش می‌دهد. اما گره شماره ۲ پس از دریافت RREQ اولین بار از گره ۱، فقط از گره دوباره آن را دریافت می‌کند، بنابراین مقدار شمارنده این RREQ در جدول RREQ_Seen آن برابر ۱ می‌باشد. گره ۳ نیز پس از دریافت RREQ از گره S، دوباره آن را از گره‌های ۱، ۵ و ۴ دریافت می‌کند، بنابراین مقدار شمارنده را برابر ۳ قرار می‌دهد.

پس از اینکه بسته‌های RREQ به مقصد رسیدند، گره مقصد شروع به ارسال بسته‌های RREP به ازای هر کدام از آنها می‌کند. در ابتدا فیلد ActiveNeighborCount در این بسته‌ها برابر صفر در نظر گرفته می‌شود. این بسته‌ها در راه برگشت به سمت مبدأ، شروع به جمع کردن مقادیر فیلد Counter مربوط به RREQ معادل خود از جداول RREQ_Seen گره‌های میانی می‌کنند. شکل شماره ۷ وضعیت شبکه را هنگامی که این بسته‌های RREP به مبدأ برمی‌گردند را نشان می‌دهد.



شکل ۷: نحوه دریافت RREP ها توسط مقصد در یک شبکه فرضی.

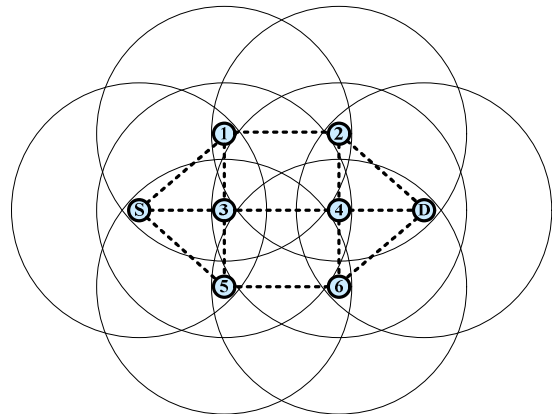
۱. هنگام نیاز به مسیر تا مقصدی خاص، بسته درخواست مسیر را برای همه پخش کن.
۲. منتظر بازگشت بسته‌های پاسخ مسیر از جانب مقصد بمان.
۳. در صورت بازگشت اولین بسته پاسخ مسیر از جانب مقصد با تنظیم یک تایمر، مدت زمان مشخصی را برای رسیدن مابقی بسته‌های پاسخ مسیر منتظر بمان.
۴. پس از انقضای زمان تایمر، بسته‌های پاسخ مسیر دریافت شده را بر اساس فیلد ActiveNeighborCount موجود در آنها، به صورت صعودی مرتب کن.
۵. طبق قرارداد مورد نظر در زمینه متعادل کردن بار، به تعداد مورد نیاز از ابتدای لیست مرتب شده بسته‌های پاسخ مسیر، مسیر انتخاب کن.
۶. ارسال اطلاعات به سمت مقصد را به طور همروند، از طریق مسیرهای انتخاب شده در مرحله ۵ آغاز کن.

شکل ۳: شبه کد عملکرد گره مبدأ.

۱. اگر یک بسته درخواست مسیر دریافت کردی و این بسته، یک بسته جدید و مورد قبول است، مشخصات این بسته درخواست مسیر را در جدول RREQ_Seen درج کن و فیلد Counter متناظر با آن را برابر با صفر قرار بده.
۲. اگر یک بسته درخواست مسیر تکراری دریافت کردی، فیلد Counter مربوط به این درخواست مسیر را در جدول RREQ_Seen خود یک واحد افزایش بده.
۳. در راستای سیاست کشف مسیر، اگر این بسته درخواست مسیر قابل بازپخش است، آن را برای همه پخش کن.

شکل ۴: شبه کد عملکرد گره‌های میانی.

برای روشن‌تر شدن راهکار الگوریتم پیشنهادی، شبکه فرضی شکل ۵ را در نظر بگیرید. در این شکل بُرد رادیویی هر گره مشخص شده‌است و وجود نقطه چین بین دو گره به این معناست که دو گره بُرد رادیویی یکدیگر قرار دارند.



شکل ۵: توپولوژی و بُرد گره‌ها در یک شبکه فرضی.

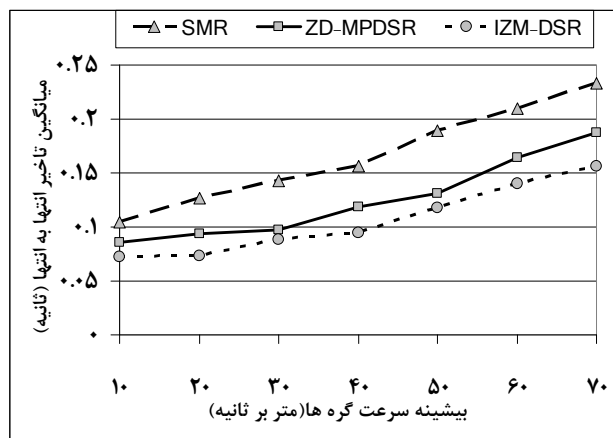
فرض کنید در شکل ۵ گره S قصد ارسال داده به طور همزمان از طریق دو مسیر برای گره D دارد. به همین منظور با ارسال بسته

۴-۲- نتایج شبیه‌سازی

در این قسمت به بررسی نتایج حاصل از شبیه‌سازی می‌پردازیم.

۴-۲-۱- میانگین تاخیر انتها به انتها

در شکل ۸ الگوریتم‌ها از نظر میانگین تاخیر انتها به انتها با هم مقایسه شده‌اند. در این شبیه‌سازی‌ها، زمان توقف یک ثانیه در نظر گرفته شده است. همانطور که در شکل مشاهده می‌شود، با افزایش بیشینه سرعت گره‌ها، در هر سه الگوریتم میانگین تاخیر انتها به انتها افزایش می‌یابد، که این امر به خاطر پویایی توپولوژی شبکه و افزایش نرخ شکسته شدن مسیرها می‌باشد. از طرفی میانگین تاخیر انتها به انتها در الگوریتم‌های ZD-MPDSR و IZM-DSR نسبت به الگوریتم SMR پایین‌تر می‌باشد که این امر به خاطر استفاده از مسیرهای مجزای ناحیه‌ای در طول فاز ارسال اطلاعات می‌باشد. در نهایت به خاطر کوتاه‌تر شدن فاز کشف مسیر در الگوریتم IZM-DSR نسبت به ZD-MPDSR، تاخیر انتها به انتها در الگوریتم پیشنهادی قدری کمتر از الگوریتم ZD-MPDSR می‌باشد.



شکل ۸: مقایسه میانگین تاخیر انتها به انتها در سه الگوریتم SMR، ZD-MPDSR و IZM-DSR

۴-۲-۲- نرخ از بین رفتن بسته‌ها

در شکل ۹ این سه الگوریتم از نظر نرخ تحویل بسته‌ها به مقصد با هم مقایسه شده‌اند. با افزایش بیشینه سرعت گره‌ها، نرخ تحویل بسته‌ها به مقصد در هر سه الگوریتم کاهش می‌یابد، که این امر به خاطر شکسته شدن پی‌در پی مسیرها و از بین رفتن بسته‌ها می‌باشد. همانطور که در شکل ۹ مشاهده می‌شود، نرخ تحویل بسته‌ها به مقصد در الگوریتم پیشنهادی قدری بیشتر از الگوریتم ZD-MPDSR می‌باشد، که این امر به خاطر کاهش تاخیر فاز کشف مسیر از یک طرف و کاهش تصادم‌های لایه MAC از طرف دیگر می‌باشد. دقت کنید در الگوریتم پیشنهادی از بسته‌های RREQ_Query و تصادم‌های لایه MAC کمک می‌کند.

همانطور که در شکل ۷ مشخص است، RREP های شماره ۳ و ۴ که مشخص کننده مسیرهای S-1-2-D و S-5-6-D می‌باشند، با فیلد ActiveNeighborCount برابر ۳ به مبدا تحویل می‌شوند و RREP شماره ۲ با مسیر S-3-4-D با مقدار ActiveNeighborCount برابر ۶ به مبدا تحویل می‌شود. در این حالت مبدا با بررسی RREP های دریافتی، دو مسیر اول و سوم را جهت ارسال همروند اطلاعات به سمت مقصد مناسب تر دیده و ارسال همروند اطلاعات را از طریق این دو مسیر آغاز می‌کند.

۴- شبیه‌سازی

در این قسمت به بیان نتایج حاصل از شبیه‌سازی الگوریتم پیشنهادی می‌پردازیم. جهت مقایسه الگوریتم پیشنهادی، نتایج مربوط به شبیه‌سازی سه الگوریتم بررسی شده‌اند، که این سه الگوریتم عبارتند از:

- الگوریتم مسیریابی SMR، ارائه شده در [9].
 - الگوریتم مسیریابی ZD-MPDSR ارائه شده در [1].
 - الگوریتم مسیریابی پیشنهادی، که در نتایج و نمودارها با عنوان IZM-DSR¹¹ نشان داده شده است.
- در هر سه الگوریتم، مبدا پس از فاز کشف مسیر، ارسال اطلاعات به سمت مقصد را به صورت همروند از طریق سه مسیر آغاز می‌کند.

۴-۱- بستر پیاده‌سازی

ما برای شبیه‌سازی از شبیه‌ساز GloMoSim¹² استفاده کرده‌ایم [11]. برای شبیه‌سازی از ۱۰۰ گره با بُرد رادیویی ۲۵۰ متر استفاده شده‌است که به صورت تصادفی در یک محدوده ۱۰۰۰ در ۱۰۰۰ متری قرار می‌گیرند و به طور تصادفی حرکت می‌کنند. همچنین ما به طور تصادفی ترافیک‌هایی از نوع CBR و FTP را بر شبکه اعمال کرده‌ایم. در این شبیه‌سازی گره‌ها از پروتکل 802.11 لایه MAC استفاده می‌کنند و برای ارسال و دریافت اطلاعات، از مدل رادیویی استاندارد RADIO-ACCNOISE استفاده شده‌است. به علاوه در تمام اجراها مدل Random Waypoint را برای تحرک گره‌های شبکه انتخاب کرده‌ایم. لازم به ذکر است در این مدل، هر گره به طور تصادفی یک نقطه را به عنوان مقصد انتخاب می‌کند، سپس با یک سرعت، مابین سرعت کمینه و بیشینه، به سمت مقصد حرکت می‌کند. پس از اینکه به مقصد رسید، برای مدت زمانی که با عنوان Pause time مشخص شده‌است، در همان نقطه می‌ماند و دوباره همین عمل را تکرار می‌کند.

زمان هر یک از شبیه‌سازی‌ها برابر ۳۰۰ ثانیه در نظر گرفته شده است و هر یک از نتایج ثبت شده، میانگین بیست و پنج بار اجرای شبیه‌سازی می‌باشد.

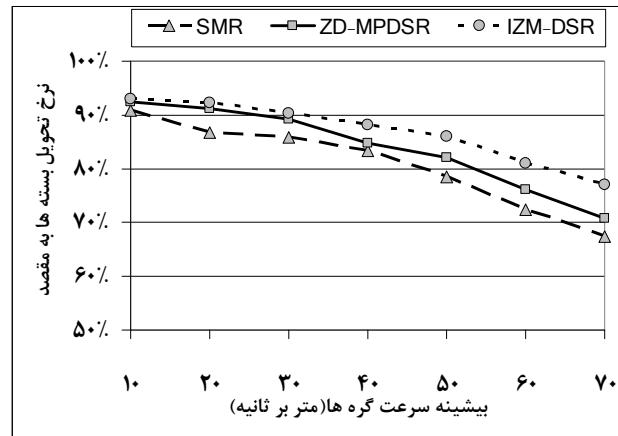
گزینه مطلوبی به حساب می‌آید، اما حتی ارسال اطلاعات از طریق مسیرهای مجزای گره‌ای نیز مستقل از یکدیگر نیستند و به خاطر مسائل ذاتی موجود در شبکه‌های موردی و پروتکل CSMA/CA، ارسال اطلاعات از یک مسیر بر روی مسیر دیگر تأثیر می‌گذارد. برای مقابله با این مساله پیشنهاد می‌شود از مسیرهای مجزای ناحیه‌ای برای ارسال همروند اطلاعات استفاده شود.

یکی از الگوریتم‌هایی که از مسیرهای مجزای ناحیه‌ای جهت ارسال همروند اطلاعات بهره می‌برد، الگوریتم ZD-MPDSR می‌باشد، اما فاز کشف مسیر در این الگوریتم با تأخیر بسیار زیادی صورت می‌گیرد و سربار قابل ملاحظه‌ای به مسیریابی تحمیل می‌کند. در این مقاله یک الگوریتم بر اساس الگوریتم ZD-MPDSR پیشنهاد شد که فاز کشف مسیر را با تأخیر بسیار کمتری نسبت به الگوریتم ZD-MPDSR اجرا می‌کند. الگوریتم مورد نظر در سناریوهای مختلف با الگوریتم‌های ZD-MPDSR و SMR مقایسه شده‌است که نتایج شبیه سازی بهبودهای قابل ملاحظه‌ای در زمینه کاهش تأخیر انتها و کاهش سربار مسیریابی نشان می‌دهند.

مراجع

[۱] نستوه طاهری جوان، مهدی دهقان، "ZD-MPDSR: یک الگوریتم مسیریابی چندمسیری مجزای ناحیه‌ای برای شبکه‌های سیار موردی"، مجموعه مقالات کامپیوتر پانزدهمین کنفرانس مهندسی برق ایران، صفحات ۳۰۹ تا ۳۱۴، مرکز تحقیقات مخابرات ایران، ۱۳۸۶، تهران، ایران.

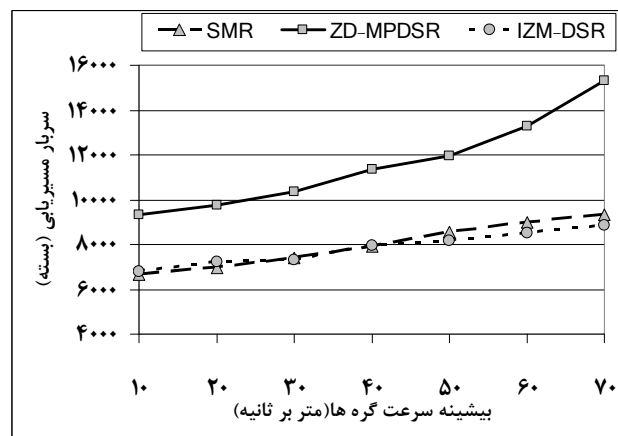
- [2] S. Sesay, Z. Yang, J. He, "A Survey on Mobile Ad Hoc Wireless Network," Information Technology Journal, vol. 2, pp. 168-175, 2004.
- [3] E. Royer, C. Toh, "A Review of Current Routing Protocols for Ad-hoc Mobile Wireless Networks," IEEE Personal Communication Magazine, pp. 46-55, 1999.
- [4] S. Mueller, R. Tsang, D. Ghosal, "Multipath Routing in Mobile Ad Hoc Networks: Issues and Challenges," Lecture Notes in Computer Science (LNCS 2965), pp. 209-234, 2004.
- [5] A. Colvin, "CSMA with Collision Avoidance," Computer Communication, Vol. 6, pp. 227-235, 1983.
- [6] S. Bandyopadhyay, S. Roy, T. Ueda, k. hasuike, "Multipath Routing in Ad hoc Wireless Networks with Directional Antenna," Personal Wireless Communication, vol. 234, pp. 45-52, 2002.
- [7] S. Roy, D. Saha, S. Bandyopadhyay, Tetsuro Ueda, S. Tanaka, "Improving End-to-End Delay through Load Balancing with Multipath Routing in Ad Hoc Wireless Networks using Directional Antenna," in proceedings of IWDC 2003: 5th International Workshop, LNCS, pp. 225-234, 2003.
- [8] B. Johnson, D. A. Maltz. "Dynamic Source Routing in Ad-Hoc Wireless Networks," Mobile Computing, vol.353, pp. 153-81, 1996.
- [9] S. j. lee, M. Gerla, "Split Multipath Routing with Maximally Disjoint Paths in Ad hoc Networks," in proceedings of IEEE



شکل ۹: مقایسه نرخ تحویل بسته‌ها به مقصد در سه الگوریتم SMR، ZD-MPDSR و IZM-DSR

۴-۲-۳- سربار مسیریابی

در شکل ۱۰ الگوریتم‌ها از نظر میزان سربار مسیریابی با هم مقایسه شده‌اند. در هر سه الگوریتم با افزایش بیشینه سرعت گره‌ها و در نتیجه افزایش تعداد دفعات اجرای فرآیند کشف مسیر، سربار مسیریابی به طور نسبی افزایش می‌یابد. همانطور که به وضوح در شکل مشاهده می‌شود، در الگوریتم ZD-MPDSR تعداد بسته‌های مسیریابی به نسبت دو الگوریتم دیگر بسیار بیشتر است که این امر به خاطر استفاده از بسته‌های RREQ_Query_Reply و RREQ_Query در فاز کشف مسیر می‌باشد. اما سربار مسیریابی در الگوریتم پیشنهادی و الگوریتم SMR تقریباً یکسان می‌باشد.



شکل ۱۰: مقایسه سربار مسیریابی در سه الگوریتم SMR، IZM-DSR و ZD-MPDSR

۵- نتیجه‌گیری

برخی الگوریتم‌های مسیریابی چندمسیری در شبکه‌های موردی برای کاهش تأخیر انتها به انتها، در مبدا داده را تقسیم‌بندی کرده و قسمت‌های مختلف را به‌طور همزمان از طریق چندین مسیر به سمت مقصد ارسال می‌کنند. در این راه استفاده از مسیرهای مجزای گره‌ای

International Conference on Communication (ICC), pp. 3201-3205, Helsinki, Finland, 2001.

[10] M. K. Marina, S. R. Das, "**On Demand Multipath Distance Vector Routing in Ad hoc Networks**," in proceedings of IEEE International Conference on Network Protocols (ICNP), pp. 14-23, California, USA, 2001.

[11] L. Bajaj, M. Takai, R. Ahuja, R. Bagrodia, M. Gerla, "**Glomosim: a Scalable Network Simulation Environment**," Technical Report 990027, Computer Science Department, UCLA, 1999.

زیر نویس ها

¹ Dynamic Source Routing.

² Ad hoc On-demand Distance Vector.

³ On-Demand.

⁴ Exposed Terminal Problem.

⁵ Hidden Terminal Problem.

⁶ Carrier Sense Multiple Access with Collision Avoidance.

⁷ Request To Send.

⁸ Clear To Send.

⁹ Split Multipath Routing.

¹⁰ Ad hoc on demand Multipath Distance Vector.

¹¹ Improved Zone-disjoint Multipath DSR.

¹² GLOBal MOBILE SIMulation.